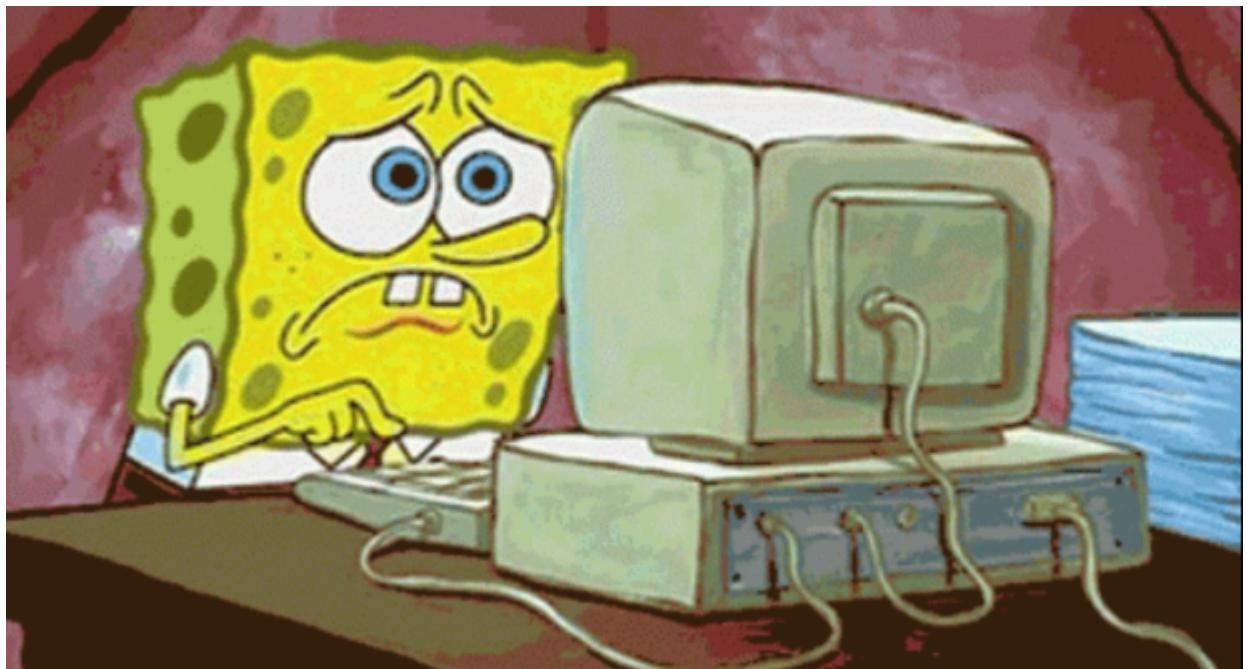


BUNDEL SOAL

TEKNIK INFORMATIKA
SEMESTER 3
TAHUN 2021/2022



“The more i C, the less i see”

DAFTAR ISI

IF2110 Algoritma dan Struktur Data	7
UTS SEMESTER I – 2010/2011	8
UTS SEMESTER I – 2011/2012	12
UTS SEMESTER I – 2014/2015	20
UTS SEMESTER I – 2016/2017	25
<u>UTS SEMESTER I – 2019/2020</u>	<u>31</u>
<u>UTS SEMESTER I – 2020/2021</u>	<u>37</u>
<u>UTS SEMESTER I – 2021/2022</u>	<u>44</u>
UAS SEMESTER I – 2011/2012	48
UAS SEMESTER I – 2012/2013	59
UAS SEMESTER I – 2014/2015	64
UAS SEMESTER I – 2016/2017	70
UAS SEMESTER I – 2017/2018	75
<u>UAS SEMESTER I – 2019/2020</u>	<u>79</u>
<u>UAS SEMESTER I – 2020/2021</u>	<u>83</u>
<u>UAS SEMESTER I – 2021/2022</u>	<u>89</u>
IF2120 Matematika Diskrit	93
UTS SEMESTER I – 2007/2008	94
UTS SEMESTER I – 2008/2009	95
UTS SEMESTER I – 2009/2010	96
UTS SEMESTER I – 2010/2011	98
UTS SEMESTER I – 2011/2012	99
UTS SEMESTER I – 2012/2013	100
UTS SEMESTER I – 2013/2014	101
UTS SEMESTER I – 2014/2015	102
UTS SEMESTER I – 2015/2016	103
UTS SEMESTER I – 2016/2017	104
UTS SEMESTER I – 2017/2018	105
UTS SEMESTER I – 2019/2020	107
<u>UTS SEMESTER I – 2020/2021</u>	<u>109</u>
<u>UTS SEMESTER I – 2021/2022</u>	<u>116</u>
UAS SEMESTER I – 2007/2008	125
UAS SEMESTER I – 2009/2010	128
UAS SEMESTER I – 2011/2012	130
UAS SEMESTER I – 2012/2013	133
UAS SEMESTER I – 2013/2014	135
UAS SEMESTER I – 2014/2015	137
UAS SEMESTER I – 2016/2017	140
UAS SEMESTER I – 2017/2018	142
UAS SEMESTER I – 2019/2020	144
<u>UAS SEMESTER I – 2020/2021</u>	<u>147</u>
<u>UAS SEMESTER I – 2021/2022</u>	<u>157</u>
IF2121 Logika Komputasional	167
UTS SEMESTER II – 2004/2005	168
UTS SEMESTER II – 2005/2006	169

UTS SEMESTER II – 2006/2007	170
UTS SEMESTER II - 2006/2007	171
UTS SEMESTER II – 2009/2010	174
UTS SEMESTER II – 2010/2011	175

UTS SEMESTER II – 2011/2012	176
UTS SEMESTER II – 2012/2013	178
UTS SEMESTER I – 2014/2015	180
UTS SEMESTER I – 2016/2017	181
<u>UTS SEMESTER I – 2017/2018</u>	<u>183</u>
UTS SEMESTER I – 2019/2020	185
<u>UTS SEMESTER I – 2020/2021</u>	<u>187</u>
<u>UTS SEMESTER I – 2021/2022</u>	<u>191</u>
UAS SEMESTER II	203
UAS SEMESTER II – 2005/2006	205
UAS SEMESTER II	208
UAS SEMESTER II – 2010/2011	212
UAS SEMESTER II – 2011/2012	213
UAS SEMESTER II – 2012/2013	214
UAS SEMESTER I – 2014/2015	216
UAS SEMESTER I – 2015/2016	218
UAS SEMESTER I – 2016/2017	220
UAS SEMESTER I – 2017/2018	222
UAS SEMESTER I – 2019/2020	224
IF2123 Aljabar Geometri	226
UTS SEMESTER I – 2007/2008	227
UTS SEMESTER I – 2008/2009	228
UTS SEMESTER I – 2009/2010	229
UTS SEMESTER I – 2010/2011	230
UTS SEMESTER I – 2011/2012	232
UTS SEMESTER II – 2007/2008	233
UTS SEMESTER II – 2008/2009	234
UTS SEMESTER II – 2009/2010	235
UTS SEMESTER II – 2010/2011	236
UTS SEMESTER II – 2011/2012	237
UTS SEMESTER I – 2012/2013	239
UTS SEMESTER I – 2014/2015	240
UTS SEMESTER I – 2015/2016	241
UTS SEMESTER I – 2016/2017	242
UTS SEMESTER I – 2017/2018	243
UTS SEMESTER I – 2019/2020	245
UTS SEMESTER I – 2020/2021	247
<u> UTS SEMESTER I – 2021/2022</u>	<u>251</u>
UAS SEMESTER PENDEK 2006/2007	260
UAS SEMESTER II – 2007/2008	261
UAS SEMESTER II – 2008/2009	262
UAS SEMESTER I – 2013/2014	263
UAS SEMESTER I – 2014/2015	264
UAS SEMESTER I – 2015/2016	265
UAS SEMESTER I – 2016/2017	266
UAS SEMESTER I – 2017/2018	267
UAS SEMESTER I – 2019/2020	269

UAS SEMESTER I – 2020/2021	271
<u>UAS SEMESTER I – 2021/2022</u>	<u>275</u>
IF2130 Organisasi dan Arsitektur Komputer	284
<u>UTS SEMESTER I – 2002/2003</u>	<u>285</u>
UTS SEMESTER II – 2002/2003	286
UTS SEMESTER I – 2004/2005	287
UTS SEMESTER I – 2005/2006	288
UTS SEMESTER I – 2006/2007	289
UTS SEMESTER I – 2007/2008	290
UTS SEMESTER II – 2009/2010	291
UTS SEMESTER II – 2010/2011	292
UTS SEMESTER II – 2011/2012	293
UTS SEMESTER II – 2012/2013	294
UTS SEMESTER I – 2013/2014	304
UTS SEMESTER I – 2014/2015	307
UTS SEMESTER I – 2015/2016	312
UTS SEMESTER I – 2017/2018	316
UTS SEMESTER I – 2019/2020	319
UTS SEMESTER I – 2020/2021	322
<u>UTS SEMESTER I – 2021/2022</u>	<u>326</u>
UAS SEMESTER II – 2010/2011	333
UAS SEMESTER II – 2011/2012	333
UAS SEMESTER II – 2012/2013	334
UAS SEMESTER I – 2013/2014	340
UAS SEMESTER I – 2014/2015	344
UAS SEMESTER I – 2015/2016	349
UAS SEMESTER I – 2017/2018	353
UAS SEMESTER I – 2019/2020	359
UAS SEMESTER I – 2020/2021	362
<u>UAS SEMESTER I – 2020/2021</u>	<u>366</u>
IF2220 Teori Bahasa Formal dan Otomata	376
UTS SEMESTER II – 2006/2007	377
UTS SEMESTER II – 2007/2008	378
UTS SEMESTER II – 2008/2009	379
UTS SEMESTER II	380
UTS SEMESTER II – 2009/2010	382
UTS SEMESTER II – 2010/2011	383
UTS SEMESTER II – 2011/2012	384
UTS SEMESTER I – 2014/2015	385
UTS SEMESTER I – 2016/2017	386
UTS SEMESTER I – 2017/2018	388
UTS SEMESTER I – 2019/2020	390
UTS SEMESTER I – 2020/2021	392
UAS SEMESTER II – 2003/2004	396
UAS SEMESTER II – 2004/2005	397
UAS SEMESTER II – 2006/2007	398
UAS SEMESTER II – 2007/2008	399

UAS SEMESTER II - 2008/2009	400
UAS SEMESTER II – 2009/2010	401
UAS SEMESTER II – 2010/2011	402
UAS SEMESTER II – 2011/2012	403
UAS SEMESTER II – 2012/2013	404
UAS SEMESTER I – 2014/2015	405
UAS SEMESTER I – 2015/2016	406
UAS SEMESTER I – 2016/2017	407
UAS SEMESTER I – 2017/2018	408
UAS SEMESTER I – 2019/2020	409
UAS SEMESTER I – 2020/2021	411
<u>UAS SEMESTER I – 2021/2022</u>	<u>415</u>

IF2110

Algoritma dan Struktur Data

UTS ALGORITMA DAN STRUKTUR DATA

UTS SEMESTER I – 2010/2011

14 Oktober 2010

120 menit

Soal 1. ADT CIRCLE (Notasi Algoritmik) [Total nilai: 30]

Diketahui sebuah ADT CIRCLE yang mendefinisikan type Circle sebagai representasi dari sebuah lingkaran:

```
{*** ADT CIRCLE ***}
{*** Definisi type Circle ***}
type Circle : <TP:Point, d:real>
{<TP,d> adalah sebuah lingkaran, dengan TP adalah titik pusat, d adalah
panjang diameter lingkaran}
(* Definisi dan spesifikasi konstruktur *)
function MakeCircle (VTP: Point, vd: real) []Circle
{membentuk sebuah lingkaran dari point VTP dengan diameter vd}
(* Definisi dan spesifikasi selektor *)
function TitikPusat (L:Circle) []Point
{memberikan titik pusat Circle L}
function Diameter (L:Circle) []real
{memberikan panjang diameter Circle L}
{*** Primitif lain ***}
procedure InputCircle (output C:Circle)
{I.S.: Sembarang}
{F.S.: C adalah lingkaran yang didapatkan dari masukan user}
```

Diberikan juga ADT POINT yang mendefinisikan type Point sebagai representasi dari sebuah titik pada sumbu koordinat kartesian:

```
{*** ADT POINT ***}
{*** Definisi type Point ***}
type Point : <x : real, y : real>
{<x,y> adalah sebuah Point, dengan x adalah absis, y adalah ordinat}
(* Definisi dan spesifikasi konstruktur *) function
MakePoint (a,b : real) []Point
{membentuk sebuah Point dari a dan b dengan a sebagai absis dan b sebagai
ordinat}
(* Definisi dan spesifikasi selektor *)
function Absis (P:Point) []real
{memberikan absis Point P}
function Ordinat (P:Point) []real
{memberikan ordinat Point P}
{*** Definisi dan spesifikasi fungsi lain ***}
function Jarak (P1, P2: Point) []real
{mengembalikan jarak antara titik P1 dan P2 dengan menggunakan rumus jarak}
```

- a. Buatlah realisasi procedure InputCircle yang membaca masukan input sebuah Circle dari user. Dalam realisasinya, perintah input dalam notasi algoritmik dibatasi hanya dapat menerima masukan yang bertipe dasar. Contoh: input(C) dengan C bertipe Circle tidak diperbolehkan. Gunakanlah konstruktur yang diberikan dalam ADT.
- b. Buatlah definisi, spesifikasi, dan realisasi function Check2Circle yang menerima masukan dua Circle C1 dan C2 dan mengembalikan kode hubungan antara dua lingkaran berdasarkan aturan berikut ini:
 - Mengembalikan integer 0 jika C1 dan C2 bersinggungan
 - Mengembalikan integer 1 jika C1 dan C2 beririsan
 - Mengembalikan integer 2 jika C1 dan C2 saling lepas yaitu tidak beririsan dan berpotongan
- c. Buatlah program utama yang membaca masukan dari user 2 buah Circle, lalu menampilkan hasil pengecekan hubungan kedua Circle tersebut.

Semua definisi type dan primitive pada ADT Circle dan ADT Point dapat langsung dipakai tanpa harus direalisasikan. Dalam realisasi sedapat mungkin gunakan semua konstanta dan fungsi-prosedur yang sudah didefinisikan. Tidak diperkenankan membuat type dan konstanta baru. Jika memerlukan fungsi-prosedur tambahan, buatlah definisi, spesifikasi, dan realisasinya.

Soal 2. ADT Array Kontigu (Notasi Algoritmik) [Total Nilai: 30]

Diketahui sebuah ADT array kontigu yang memorinya dialokasikan secara sattik dengan ukuran mengikuti suatu konstanta tertentu (NMax). Indeks awal dimulai dari suatu konstanta IMin. Elemen array bertipe integer. Elemen efektif (yang terdefinisi) dapat dimulai dari indeks berapapun (tidak harus IMin). Beberapa contoh array jika diasumsikan IMin=1 dan NMax=100 adalah sebagai berikut:

Array kosong:

1	2	3	4	5	6	7	8	...	100
---	---	---	---	---	---	---	---	-----	-----

Array dengan elemen efektif dari indeks 1 s.d. 5 (rata kiri):

1	10	-1	2	3	6	7	8	...	100
---	----	----	---	---	---	---	---	-----	-----

Array dengan elemen efektif dari indeks 3 s.d. 8:

1	2	-1	5	6	3	5	8	...	100
---	---	----	---	---	---	---	---	-----	-----

Array dengan elemen efektif dari indeks 96 s.d. 100:

1	2	3	4	...	96	97	98	99	100
---	---	---	---	-----	----	----	----	----	-----

Array penuh elemen efektif dari indeks 1 s.d. 100:

-1	3	75	73	...	2	4	6	-2	2
----	---	----	----	-----	---	---	---	----	---

Dibawah ini adalah kamus umum ADT tersebut. Lengkapilah type TabEl dibawah yang dapat merepresentasikan array diatas. Berdasarkan type TabEl yang Anda buat, buatlah realisasi semua fungsi dan prosedur dibawah ini dalam notasi algoritmik. Dalam realisasi sedapat mungkin gunakan semua konstanta dan fungsi/prosedur yang telah didefinisikan. Dipersilakan menambahkan type dan konsatanta baru jika diperlukan. Jika perlu menambahkan fungsi/prosedur baru, buatlah definisi, spesifikasi, dan realisasinya.

```
(Kamus Umum)
type indeks : integer ≥
0 type eltype : integer
(Konstanta)
constant IMin : indeks = 1 (indeks memori array minimum)
constant NMax : integer = 100 (ukuran memori array
maksimum)
(type array statik)
type TabEl : <TI : array[...] of eltype, {memori tempat menyimpan data}
... {komponen lain jika diperlukan, mungkin lebih dari satu}
    >
(Selektor)
function GetFirstIdx (T: TabEl) ∅ indeks
{menghasilkan indeks pertama efektif pada array T. Prekondisi T tidak kosong.}

function GetLastIdx (T: TabEl) ∅ indeks
{menghasilkan indeks terakhir efektif pada array T. Prekondisi T tidak kosong.}
(predikat pemeriksaan)
function IsEmpty (T : TabEl) ∅ boolean
{menghasilkan true jika T adalah array kosong, tidak ada elemen
terdefinisi} function IsFull (T : TabEl) ∅ boolean
{menghasilkan true jika T adalah array yang penuh, yaitu semua elemen terdefinisi}
(* Konstruktor *)
procedure MakeEmpty (output T : TabEl)
{membentuk array kosong}
{I.S.: Sembarang F.S.: T terbentuk sebagai array
kosong} procedure AddLastElmt (input/output T: TabEl)
{Menambahkan 1 elemen pada akhir array T yang tidak penuh. Jika array terisi secara rata
kanan, maka seluruh isi memori array harus digeser sampai indeks awal mencapai IMin (dengan
urutan tidak berubah) dan elemen baru ditambahkan di akhir array.}
{I.S.: T terdefinisi, mungkin kosong, tapi tidak penuh}
{F.S.: T bertambah satu elemen di akhir.}
```

Soal 3. Binary Search tree dan List (Notasi Fungsional) [Total nilai: 40]

Binary search tree (BST) adalah pohon biner yang memenuhi sifat bahwa semua simpul yang muncul di sub pohon kiri selalu bernilai lebih kecil daripada akar sedangkan semua simpul yang muncul di sub pohon kanan selalu bernilai lebih besar daripada akar. Oleh sebab itu, hasil penelusuran inorder terhadap BST akan selalu memberikan hasil nilai yang terurut membesar. Pada BST, selain menyimpan data nilai, setiap simpul juga memuat jumlah kemunculan nilai tersebut. Berikut ini adalah definisi ADT BST beserta primitifnya:

```

TYPE_BINARY SEARCH TREE
(BST) DEFINISI DAN SPESIFIKASI TYPE
Count : integer >0> {jumlah kemunculan}
type ElmtBST : <L: BSTree, A: ElmtBST, R: BSTree> {notasi infix}
+BSTtree diri dari akar, L dan R yang merupakan subpohon kiri dan kanan}
DEFINISI DAN SPESIFIKASI SELEKTOR
Info : BSTree tidak kosong integer
(Info(P) adalah nilai dari akar P. Jika P adalah //L A R\\, Info(P) adalah A.Info)
Count : BSTree tidak kosong integer>0
(Count(P) adalah jumlah kemunculan dari nilai pada akar P. Jika P adalah //L A R\\, Count(P) adalah A.Count)
Left : BSTree tidak kosong BSTree
(Left(P) adalah sub pohon kiri dari P. Jika P adalah //L A R\\, Left(P) adalah L)
Right : BSTree tidak kosong BSTree
(Right(P) adalah sub pohon kanan dari P. Jika P adalah //L A R\\, Right(P) adalah R)
R) DEFINISI DAN SPESIFIKASI KONSTRUKTOR
(Konstruktur BST dengan basis pohon kosong dituliskan dalam notasi //L A R\\)
(Misalnya:)
// //\\ <a,b> //\\ \\ untuk sebuah pohon yang hanya terdiri dari satu simpul)
// Left(P) <Info(P), Count(P)> Right(P)\\ membentuk pohon
P) DEFINISI DAN SPESIFIKASI PREDIKAT
IsEmpty : BSTree boolean
(IsEmpty(P) true jika P kosong) IsTreeOneElmt : BSTree boolean
(IsTreeOneElmt(P) true jika P hanya mempunyai satu elemen)

```

Buatlah definisi, spesifikasi, dan realisasi dari fungsi InsElmtBST, yang menerima sebuah integer dan sebuah BST dan memasukkan integer tersebut kedalam BST. Apabila elemen sudah ada pada BST, jumlah kemunculan (Count(P)) akan diincrement. Apabila belum ada, maka akan ditambahkan simpul baru sebagai daun pada posisi yang tepat sehingga sifat BST tetap dipenuhi.

Buatlah definisi, spesifikasi, dan realisasi dari fungsi BST2List yang akan mengubah sebuah BST menjadi list yang elemennya dapat berulang serta terurut membesar. Definisi ADT List of Integer beserta primitifnya diberikan sebagai

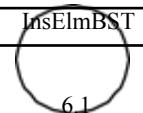
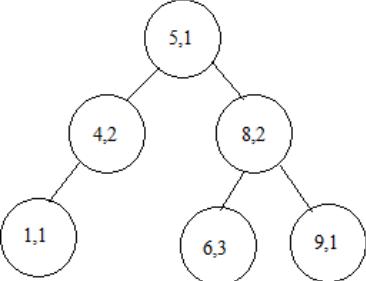
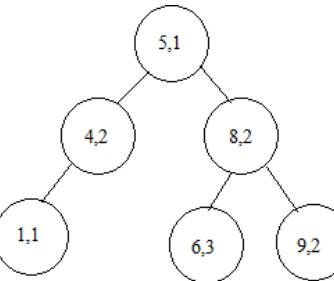
```

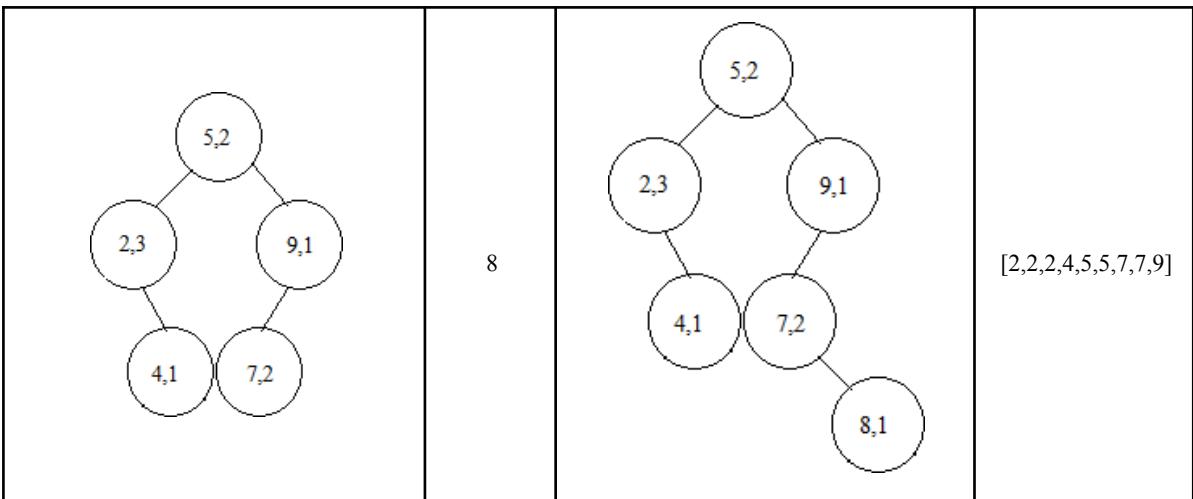
TYPE LIST OF INTEGER
DEFINISI DAN SPESIFIKASI
TYPE
type List of integer
(Definisi list yang elemennya integer) DEFINISI DAN SPESIFIKASI
KONSTRUKTOR
Konso : integer, List of integer o List of integer
(Konso(e,L) menghasilkan sebuah list dari e dan L dengan e sebagai elemen pertama e o L o L') DEFINISI DAN SPESIFIKASI SELEKTOR
FirstElmt: List of integer tidak kosong integer
(FirstElmt(L) menghasilkan elemen pertama list L)
Tail: List of integer tidak kosong o List of integer
(Tail(L) menghasilkan list tanpa elemen pertama list L) DEFINISI DAN SPESIFIKASI PREDIKAT
IsEmpty : List of integer boolean
(IsEmpty(L) true jika L kosong)
IsOneElmt : List of integer boolean
(IsOneElmt(L) true jika L hanya mempunyai satu elemen)

```

berikut:

Contoh masukan dan keluaran fungsi:

Masukan		Keluaran	
BSTree	Elemen	InsElmt	BST2List
Tree kosong	6		[]
	9		[1,4,4,5,6,6,6,8,8,9]



Semua fungsi yang didefinisikan pada type BSTree dan List of integer dapat digunakan tanpa perlu dibuat realisasinya. Jika perlu membuat fungsi baru buatlah definisi, spesifikasi, dan realisasinya.

UTS SEMESTER I – 2011/2012

20 Oktober 2011

100 menit

Soal 1. Jadwal Kereta Api [Prosedural, Notasi Algoritmik] (Total Bobot:25%)

Sebuah sistem pengendali jadwal kereta api terpusat digunakan untuk mengatur jadwal kereta api di seluruh Indonesia. Sebuah jadwal kereta api diketahui nomornya, berikut kota asal berangkatan, kota tujuan, jam keberangkatan, jam kedatangan, dan tarif. Semua jam dituliskan dalam bentuk jam,menit,detik dalam skala 24 jam(jam terkecil 0:0:0 dan terbesar 23:59:59). Tarif adalah harga tiket kereta api per penumpang dalam rupiah yang dinyatakan sebagai bilangan positif atau 0.

Beberapa jadwal kereta api masih aktif beroperasi,namun beberapa yang lain sudah tidak aktif. Jadwal kereta api yang sudah tidak aktif tidak dihapus dari sistem,namun hanya diberi penanda bahwa sedang tidak aktif, karena mungkin saja suatu saat akan diaktifkan lagi. Tarif untuk jadwal kereta api yang tidak aktif adalah 0 rupiah.

Jika Anda menjadi programmer yang diminta untuk membuat program yang mengelola jadwal kereta api tersebut:

- Definisikan ADT sederhana untuk persoalan penjadwalan kereta api di atas, yang terdiri atas deklarasi type bentukan dan definisi,spesifikasi, dan realisasi dari konstruktor-nya.
- Buatlah sebuah program utama yang digunakan untuk membaca dari keyboard data sebuah jadwal kereta api, termasuk apakah jadwal tersebut masih aktif atau tidak. Untuk pemasukkan tariff kereta api, harus dipertimbangkan apakah jadwal tersebut masih aktif atau tidak adan apakah nilainya bilangan bulat positif atau 0. Jika masukan tarif tidak memenuhi persyaratan, pemasukan data harus diulang sampai syarat tersebut dipenuhi. Gunakan sedapat mungkin primitive dari semua ADT yang terdefinisi. Jika dibutuhkan asumsi tambahan (untuk memperjelas spesifikasi program), tuliskan sebagai komentar pada spesifikasi program.

Anda dapat menggunakan ADT JAM berikut, tanpa perlu membuat realisasinya.

```
{ Modul ADT JAM }

{ Definisi TYPE JAM <HH:MM:SS> }

type JAM: <HH : integer[0..23], { bagian jam }
          MM: integer[0..59], { bagian menit }
          SS: integer[0..59], { bagian detik } >

{ *** Konstruktor :Membentuk sebuah jam dari komponen-komponennya *** }

function MakeJAM (h:integer ,m:integer, d:integer) : JAM
{ Membentuk sebuah JAM dengan h sebagai HH, m sebagai MM, dan D sebagai SS. }

{ Prekondisi: h,m,d valid untuk membentuk JAM}

{ *** Primitif-primitif Lain *** }

procedure BacaJAM (output J: JAM)
{ Membaca dan memvalidasi masukan komponen JAM dari keyboard dan membentuk JAM }

{ berdasarkan masukan komponen JAM yang valid }
```

```
{ I.S. : J sembarang }

{ F.S: J terdefinisi dengan komponen-komponen masukan dari keyboard)

procedure CopyJam (input Jin :JAM , output JOOut: JAM)

{ Menyalin Jin ke JOOut }

{ I.S.: Jin terdefinisi; F>S: JOOut berisi salinan dari Jin.}
```

Soal 2. Array of Integer [Prosedural, Notasi Algoritmik] (Total Bobot : 25%)

Diketahui ADT Array of Integer yang diimplementasikan secara statik dan eksplisit sebagai berikut:

```
{ MODUL TABEL INTEGER DENGAN ALOKASI STATIK }
{ Berisi Definisi dan semua primitive pemrosesan table statik dan
eksplisit }
{ Penempatan elemen selalu rapat kiri }
{ Kamus Umum }
constant IdxMax : integer = 100
constant IdxUndef : integer = -999 { indeks tak terdefinisi } constant
ValUndef :integer = -9999 { nilai elemen tidak terdefinisi }
{ Definisi elemen dan koleksi objek }
type IdxType :integer { type indeks }
type ElType: integer { type elemen tabel }
type TabInt: < TI : array [1..IdxMax] of ElType {memori tempat penyimpan
elemen }
                           Neff : integer ≥ 0{ banyaknya elemen efektif } >
{ Definisi:
  Tabel kosong : T.Neff = 0
  Definisi elemen pertama : T.TI[i] dengan i=1
  Definisi elemen terakhir yang terdefinisi: T.TI[i] dengan i=T.Neff }
```

spesifikasi , dan realisasinya.

```
procedure Sorting (input/output T: TabInt)
{ T.TI diurutkan membesar. Tuliskan teknik apa yang digunakan untuk sorting
dalam bentuk komentar.}
{ I.S. : T tidak kosong. }
{ F.S. : T.TI terurut membesar. }

procedure ClosestPair (input T:TabInt; output P1,P2: ElType)
{ Menghasilkan 2 buah nilai elemen T yang memeliki selisih terkecil.}
{ I.S : T terdefinisi, mungkin kosong, P1 dan P2 sembarang.}
{ F.S.: Jika T memeliki lebih dari 1 elemen, P1 dan P2 berisi elemen-
elemen T.TI yang memeliki selisih terkecil. Urutan bebas.}
{Jika ada beberapa pasang elemen yang memiliki selisih terkecil yang
sama, }      {maka diambil yang pertama ditemukan.}
{Jika T memiliki lebih kecil dari 1 elemen, P1 dan P2 berisi elemen tidak
terdefinisi.}
{ Hint: Gunakan procedure sorting. }
```

Contoh:

1	3	Masukan	Keluaran	
0	1	Sorting	Closest Pair	
T: TabInt T.TI	1 3 5 6 10	T: TabInt T.TI	P1= 0 P2=1	
1 3 4 5 6 IidxMax ... 2	1 3 4 5 6 IidxMax ... 2	1 3 4 5 6 IidxMax ... 2	P1= 3 P2=3	
T.Neff=7 T.TI	1 3 5 6 10	T.Neff=7 T.TI	P1=ValUndef P2=ValUndef	
1 3 4 5 6 IidxMax 2	1 3 4 5 6 IidxMax 2	T.Neff=7 Tidak Terdefinisi		

T.Neff=0

Soal 3. ADT Matriks [Prosedural, Notasi Algoritmik] (Total Bobot : 25 %)

Diberikan definisi dan spesifikasi ADT matriks sebagai berikut:

```

type indeks : integer { indeks baris dan kolom }
constant BrsMin : indeks = 1
constant BrsMax : indeks = 100
constant KolMin : indeks = 1
constant KolMax : indeks = 100
type el_type : integer
type MATRIKS :
< Mem: matrix[BrsMin..BrsMax, KolMin..KolMax] of el_type,
  NBrsEff :integer > 0 { banyaknya/ukuran baris yang terdefinisi }
  NKolEff: integer > 0 { banyaknya/ukuran kolom yang terdefinisi }
>
{ Memori matriks yang diakai selalu di "ujung kiri atas" }

{***** DEFINISI PROTOTIPE PRIMITIF *****}
{*** Konstruktor membentuk MATRIKS ***}
procedure CreateMATRIKS (input NB,NK :integer, output M:MATRIKS)
{ Membentuk sebuah matriks "kosong" berukuran NBxNK di "ujung kiri"
memori }
{ I.S. NB dan NK adalah valid untuk memori matriks yang dibuat}
{ F.S. MATRIKS M terdefinisi dengan ukuran NBxNK}

{***Selektor MATRIKS ***}
{*** Selektor Get ***}
function GetNBrsEff (M:MATRIKS) integer
{ Mengirimkan banyaknya baris efektif M }
function GetNKolEff (M:MATRIKS) integer
{ Mengirimkan banyaknya kolom efektif M }
function GetElmt (M:MATRIKS; i,j :indeks) el_type
{ Mengirimkan elemen M dengan nomor baris I dan nomor kolom j }
function GetElmtDiagonal (M:MATRIKS, i:indeks) el_type
{ Mengirimkan elemen M[i,i] }

{*** Untuk sebuah matriks M yang terdefinisi : ***}

```

```
function GetFirstIdxBrs (M:MATRIKS) indeks  
{ Mengirimkan indeks baris terkecil M }
```

```

function GetFirstIdxKol (M:MATRIKS) ① indeks
{ Mengirimkan indeks kolom terkecil M }
function GetLastIdxBrs (M:MATRIKS) ① indeks
{ Mengirimkan indeks baris terbesar M }
function GetLastIdxBrs (M:MATRIKS) ① indeks
{ Mengirimkan indeks kolom terbesar M }

{*** Selektor Set: Operasi mengubah nilai elemen matriks ***}
procedure SetBrsEff (input/output M:MATRIKS, input NB:integer)
{ I.S. M sudah terdefinisi }
{ F.S. Nilai M.NBrsEff diisi dengan NB }
procedure SetKolEff (input/output M:MATRIKS, input NK:integer)
{ I.S. M sudah terdefinisi }
{ F.S. Nilai M.NKolEff diisi dengan NK }
procedure SetElmt (input/output M:MATRIKS, input i,j:integer, input
X :el_type)
{ I.S. M sudah terdefinisi }

```

Dengan menggunakan ADT di atas, selesaikan soal –soal di bawah ini. Spesifikasi prosedur/fungsi tidak perlu ditulis ulang. Jika perlu membuat fungsi/prosedur baru, harus ditulis definisi, spesifikasi, dan realisasinya.

a. Flip Matriks

Suatu matriks dapat di-flip yaitu “dicerminkan” berdasarkan “sumbu” diagonal matriks. Elemen diagonal matriks terletak pada posisi/indeks baris yang sama dengan posisi/indeks kolom. Elemen yang lain adalah elemen non diagonal. Ketika suatu matriks di-flip, elemen-elemen non diagonal dipertukarkan dengan elemen lain pada posisi yang bersesuaian. Lihat ilustrasi.

Buatlah sebuah procedure FlipMatriks yang melakukan flip terhadap matriks dengan definisi dan spesifikasi sebagai berikut:

```

procedure FlipMatriks (input/output M:MATRIKS)
{ Procedure FlipMatriks melakukan pencerminan matriks M berdasarkan sumbu
diagonal }
{ I.S. : M adalah sebuah matriks bujursangkar (jumlah kolom dan barisnya
sama) dan tidak kosong }
b. { F.S. MatriksDiagonal akan hasil “pencerminan” dari M berdasarkan “sumbu” diagonal
matriks M dikatakan sebagai diagonal jika M memiliki jumlah kolom dan baris yang sama, semua elemen non
diagonal bernilai nol, dan elemen diagonalnya ada yang tidak bernilai 0. Sebuah matriks yang memiliki jumlah kolom

```

dan baris yang sama, dan semua elemennya bernilai nol tidak dikatakan sebuah matriks diagonal.

Buatlah sebuah fungsi IsMatriksDiagonal yang menerima sebuah matriks M mengembalikan nilai true jika M adalah sebuah matriks diagonal dengan definisi dan spesifikasi sebagai berikut:

```

function IsMatriksDiagonal (M:Matriks) ① boolean
{ Fungsi IsMatriksDiagonal menerima matriks M yang sudah terdefinisi dan tidak
mungkin kosong dan mengembalikan true jika M adalah matriks diagonal. }

```

Ilustrasi dan contoh:

4	5
2	6

3	4	5
2	2	6
3	1	5

0	0	0
0	1	0
0	0	0

0	0	0
0	0	0
0	0	0

Matriks A

Matriks B

Matriks C

Matriks D

2	6
1	5

1	0	0
0	3	0
0	0	1

6	0	0	0
0	2	0	0
0	0	7	0

1	1
1	1

Matriks E

Matriks F

Matriks G

Matriks H

4	2
5	6

FlipMatriks(A)
FlipMatriks(B)

FlipMatriks(E)

2	1
6	5

Fungsi	Hasil
IsMatriksDiagonal(A)	false
IsMatriksDiagonal(B)	false
IsMatriksDiagonal(C)	true
IsMatriksDiagonal(D)	false
IsMatriksDiagonal(E)	false
IsMatriksDiagonal(F)	True
IsMatriksDiagonal(G)	false

Soal 4. Binary Tree dan List [Notasi Fungsional] (Total Bobot : 25 %)

Berikut ini adalah definisi ADT Pohon Biner (Binary Tree) beserta primitifnya:

```

TYPE BINARY TREE (BinTree)
DEFINISI DAN SPESIFIKASI TYPE
type BinTree : < L:BinTree, A: integer, R : BinTree > {notasi infiks}
    { BinTree terdiri dari Akar bertipe integer , serta L dan R
      (BinTree) yang merupakan subpohon kiri dan subpohon kanan }

DEFINISI DAN SPESIFIKASI SELEKTOR
Akar: BinTree tidak kosong  $\sqsubseteq$  integer
{ Akar (P) adalah nilai dari akar P. Jika P adalah //L A R\\, Akar (P) adalah A }
Left : BinTree tidak kosong  $\sqsubseteq$  BinTree
{ Left (P) adalah subpohon kiri dari P. Jika P adalah //L A R\\, Left (P) adalah L }
Right : BinTree tidak kosong  $\sqsubseteq$  BinTree
{ Right (P) adalah sub pohon kanan dari P. Jika P adalah //L A R\\, Right (P) adalah R }

DEFINISI DAN SPESIFIKASI KONSTRUKTOR
{ Perhatikanlah bahwa konstruktur BinTree dengan basis pohon kosong dituliskan dalam
notasi infiks sebagai //L A R\\ }

{ Misalnya :
{ //34\\ untuk sebuah pohon yang hanya terdiri dari 1 simpul }
{ //Left (P) Akar(P) Right (P) \\ kembali membentuk pohon P }

DEFINISI DAN SPESIFIKASI PREDIKAT
IsEmpty : BinTree  $\sqsubseteq$  boolean
{IsEmpty(P) true jika P kosong ://\\}
IsOneElmt :BinTree  $\sqsubseteq$  Boolean
{IsOneElmt(P) true jika P hanya mempunyai satu elemen, yaitu akar ://A\\ }

```

- Buatlah definisi, spesifikasi, dan realisasi dari fungsi CountNonDaun, yang menerima sebuah *Binary Tree* yang tidak kosong P dan menghitung jumlah simpul pada P yang bukan merupakan simpul daun.
 - Binary Search Tree (BST) adalah pohon biner yang memenuhi sifat bahwa semua simpul yang muncul di sub pohon kiri selalu bernilai lebih kecil dari pada akar, sedangkan semua simpul yang muncul di sub pohon kanan selalu bernilai lebih besar dari pada akar. Oleh sebab itu, hasil penelusuran *inorder* terhadap BST akan selalu memberikan hasil nilai yang terurut membesar.
- Balanced Tree adalah pohon biner yang memiliki sifat bahwa perbedaan tinggi sub pohon kiri dengan sub pohon kanan maksimum 1 dan perbedaan banyaknya simpul sub pohon kiri dengan sub pohon kanan maksimum 1. Balanced Binary Search Tree adalah pohon biner yang memenuhi sifat binary search tree dan balanced tree.

Buatlah realisasi dari fungsi List2BBST berikut ini :

```

List2BBST : List of integer  $\sqsubseteq$  BinTree
{List2BBST(L) menghasilkan sebuah binary tree yg memenuhi syarat balanced binary search
tree. Prekondisi: Jika L tidak kosong, maka L memiliki elemen yang unik dan terurut
membesar.
Hint: Jika list L terdiri dari n elemen, maka sejumlah (n div 2) elemen terdepan akan
membangun sub pohon kiri, elemen berikutnya menjadi akar, dan elemen sisanya membangun
sub pohon kanan.}

```

```

TYPE LIST OF INTEGER
DEFINISI DAN SPESIFIKASI TYPE
type List of integer
{ Definisi: list yang elemennya integer }

DEFINISI DAN SPESIFIKASI KONSTRUKTOR
Konso: integer, List of integer  $\cup$  List of integer
{ Konso(e,L) menghasilkan sebuah list dari e dan L, dengan e sebagai elemen pertama e  $\circ$  L  $\cup$  L' }

DEFINISI DAN SPESIFIKASI SELEKTOR
FirstElmt: List of integer tidak kosong  $\sqcup$  integer
{ FirstElmt (L) menghasilkan elemen pertama list L }

Tail: List of integer tidak kosong  $\sqcup$  List of integer
{ Tail(L) menghasilkan list tanpa elemen pertama list L }

```

Definisi ADT List of Integer beserta primitifnya diberikan sebagai berikut:

```

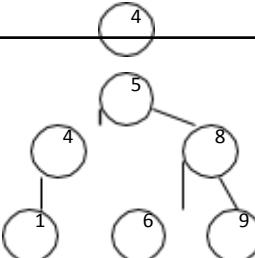
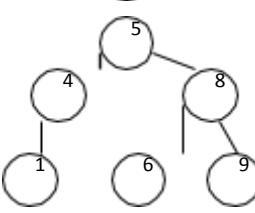
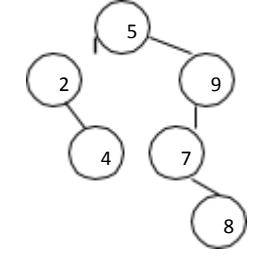
DEFINISI DAN SPESIFIKASI PREDIKAT
IsEmpty : List of integer  $\sqcup$  boolean
{ IsEmpty (L) true jika L kosong }

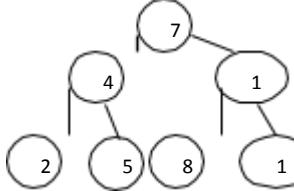
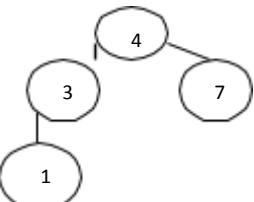
IsOneElmt: List of integer  $\sqcup$  boolean
{ IsOneElmt (L) true jika L hanya mempunyai satu elemen }

DEFINISI DAN SPESIFIKASI PRIMITIF LAIN
NbElmt : List of integer  $\sqcup$  integer  $\geq 0$ 
{ NbEmt(L) mengembalikan jumlah elemen yang terdapat di dalam L }

```

Contoh masukan dan keluaran fungsi adalah sebagai berikut:

Masukan (P)	CountNonDaun(P)
	0
	3
	4

Masukan(L)	List2BBST(L)
[]	Pohon kosong
[5]	
[2,4,5,7,8,10,11]	
[14,36,46,77]	

Semua fungsi yang didefinisikan pada type BinTree dan List of integer dapat digunakan tanpa perlu dibuat realisasinya. Jika perlu membuat fungsi baru (fungsi antara), buatlah definisi, spesifikasi, dan realisasinya.

UTS SEMESTER I – 2014/2015

23 Oktober 2014

100 menit

Soal 1. ADT Stack

Diberikan definisi ADT Stack sebagai berikut:

```
{ Kamus Umum }
constant Nil : integer = 0      { Nil = stack dengan elemen kosong }
constant MaxEl : integer = 10 { Banyaknya elemen maksimum }
type infotype : integer      { elemen stack }
type address : integer       { indeks tabel }

/* Versi I : dengan menyimpan tabel dan alamat top secara eksplisit */
type Stack : < T : array [1..MaxEl] of infotype, { tabel penyimpan elemen stack }
            TOP : address      { alamat TOP: elemen puncak }
            >

{ Deklarasi: S : Stack
  Definisi akses (diimplementasikan sebagai selektor):
  TOP(S) adalah S.TOP
  InfoTOP(S) adalah S.T[S.TOP] }

{ *** Konstruktor/Kreator *** }

procedure CreateEmpty(output S : Stack)
{ I.S. Sembarang
  F.S. Membuat sebuah stack S yang kosong berkapasitas MaxEl
       jadi indeksnya antara 1..MaxEl karena 0 tidak dipakai
       Ciri stack kosong : TOP bernilai Nil }

{ *** Predikat Untuk test keadaan Stack *** }

function IsEmpty (S : Stack)      →
boolean
{ Mengirim true jika Stack kosong: lihat definisi di atas }

function IsFull(S : Stack)      →      boolean
{ Mengirim true jika tabel penampung nilai elemen stack penuh }

{ *** Operasi terhadap Stack *** }

procedure Push (input/output S : Stack, input X : infotype)
{ Menambahkan X sebagai elemen Stack S.
  I.S. S mungkin kosong, tabel penampung elemen stack TIDAK penuh
  F.S. X menjadi TOP yang baru, TOP bertambah 1 }

procedure Pop (input/output S : Stack, output X : infotype)
{ Menghapus X dari Stack S.
  I.S. S tidak mungkin kosong
  F.S. X adalah nilai elemen TOP yang lama, TOP berkurang 1 }
```

Implementasi primitif lainnya hanya dapat memanfaatkan primitif stack yang telah diberikan di atas (akses elemen secara langsung tidak diperbolehkan).

Dengan definisi stack di atas, buatlah definisi, spesifikasi, dan implementasi primitif tambahan stack berikut ini.

- Prosedur Duplicate; yang akan menduplikasi elemen TOP pada stack yang menjadi parameter prosedur, sehingga elemen stack bertambah satu. Jika stack penuh, maka stack tidak berubah. Stack tidak kosong
- Fungsi Peek untuk mengintip elemen TOP; menerima sebuah stack dan mengembalikan elemen TOP, stack tidak berubah. Asumsi stack tidak kosong

- c. Prosedur Swap; yang akan mempertukarkan dua elemen teratas stack. Jika elemen stack hanya satu, maka stack tidak berubah. Stack tidak kosong.
- d. Tuliskan implementasi dari prosedur Rotate berikut ini tanpa harus menulis ulang spesifikasi.

Contoh: MaxEl = 7

S (TOP di kanan)	Duplicate(S)	Peek(S)	Swap(S)	Rotate(S,3,true)	Rotate(S,6,false)
[1 2 3 4 5]	[1 2 3 4 5 5]	5	[1 2 3 5 4]	[1 2 4 5 3]	[5 1 2 3 4]
[1 2 3 4 5 6 7]	[1 2 3 4 5 6 7]	7	[1 2 3 4 5 7 6]	[1 2 3 4 6 7 5]	[1 7 2 3 4 5 6]
[9]	[9 9]	9	[9]	[9]	[9]

Soal 2. ADT Mesin Event dan ADT Queue

Didefinisikan ADT Mesin Event yang mengadopsi mesin kata versi 1 (proses berhenti apabila EndEvent bernilai true), serta ADT Queue yang diimplementasikan dengan tabel alternatif ke-2 (head bergerak jika Del elemen; queue mungkin “penuh semu”).

Mesin Event akan membaca event dari pita karakter. Contoh isi pita adalah sbb:

10-A 15-D 20-D 25-A 30-A 32-D 50-A 59-D.

Sebuah event dituliskan pada pita karakter dalam format <waktu event>-<jenis event>. Misalnya ”10-A” menyatakan pada waktu 10 terjadi event A (Add – penambahan elemen).

Hasil akuisisi event oleh Mesin Event adalah sebuah CEvent terdiri dari dua komponen, yaitu: waktu event dan jenis event (A/D – untuk Add dan Delete). Misalnya, hasil pembacaan pita di atas akan menghasilkan rangkaian event

<10,'A'>, <15,'D'>, <20,'D'>, <25,'A'>, <30,'A'>, <32,'D'>, <50,'A'>, <59,'D'>

Pita karakter diasumsikan selalu dalam format yang benar (tidak ada bentuk-bentuk “kata” lain yang muncul dalam pita, dan waktu event pasti terurut membesar) atau merupakan pita kosong yang hanya terdiri atas ‘.’ (mungkin didahului sejumlah spasi/blank).

Berikut adalah definisi Mesin Event:

```
{ ***** MesinEvent ***** }
use MESINKAR

{ ***** Konstanta ***** }
constant MARK      : character =
'.' constant BLANK : character =
'

constant SEPARATOR : character = '-'

{ ***** Definisi State Mesin ***** }

type TEvent : < ETime : integer>0, {waktu event}
           EType : character['A','D'] {jenis event} >
EndEvent : boolean { penanda akhir akuisisi event }

CEvent : TEvent

{ ***** Primitif-Primitif Mesin Kata ***** }

procedure STARTEVENT
{ I.S. : CC sembarang }
{ F.S. : EndEvent = true, dan CC = MARK; }
{ atau EndEvent = false, CEvent adalah Event yang sudah
  diakuisisi, CC karakter pertama sesudah karakter terakhir
  yang diakuisisi
  Proses akuisisi Event menggunakan prosedur AkuisisiEvent }

procedure ADVEVENT
{ I.S. : EndEvent = false; CC karakter setelah karakter terakhir yang sudah
  diakuisisi }
{ F.S. : Jika CC = MARK, maka EndEvent = true; }
{ atau EndEvent = false, CEvent adalah Event terakhir yang sudah
  diakuisisi, CC karakter pertama sesudah karakter terakhir yang
  diakuisisi
  Proses akuisisi Event menggunakan prosedur AkuisisiEvent }
```

```

procedure AkuisisiEvent
{ Mengakuisisi event, mengkonversinya dan menyimpannya dalam CEvent }
{ I.S. : CC adalah karakter pertama dari event yang akan diakuisisi }
{ F.S. : CEvent berisi event yang sudah diakuisisi; CC = BLANK atau CC = MARK;
{ CC adalah karakter sesudah karakter terakhir yang diakuisisi. }
{ Proses akuisisi harus menggunakan fungsi KarakterToInteger. }

function KarakterToInteger (C : character) → integer
{ Mengirimkan integer [0..9] yang merupakan konversi dari character C yang
merupakan salah satu dari karakter ['0'...'9']. }

```

Berikut adalah definisi ADT Queue yang direpresentasikan dengan array statik dengan alternatif-2:

```

{ *** ADT Queue, model alternatif 2 dengan head bergerak dan bisa penuh "semu"
*** }

{ *** Konstanta *** }

constant Nil : integer = 0
constant MaxEl : integer = 10

{ *** Definisi elemen dan address *** }

type infotype : integer

type address : integer { indeks tabel }

{ *** Definisi Type Queue *** }

type Queue : < T : array [1..MaxEl] of infotype, { tabel penyimpan elemen }
    HEAD : address, { alamat penghapusan }
    TAIL : address { alamat penambahan } >

{ Definisi Queue kosong: Head = Nil; TAIL = Nil. }

{ *** Predikat Pemeriksaan Kondisi Queue *** }

function IsEmpty (Q : Queue) → boolean
{ Mengirim true jika Q kosong }

function IsFull (Q : Queue) → boolean
{ Mengirim true jika tabel penampung elemen Q sudah penuh yaitu mengandung MaxEl
elemen }

function NBElmt (Q : Queue) → integer
{ Mengirimkan banyaknya elemen queue. Mengirimkan 0 jika Q kosong. }

{ *** Konstruktor *** }

procedure CreateEmpty (output Q : Queue)
{ I.S. Sembarang }

{ F.S. Sebuah Q kosong terbentuk dengan HEAD = Nil dan TAIL = Nil }

{ *** Operator-Operator Dasar Queue *** }

procedure Add (input/output Q : Queue, input X : infotype)
{ Proses : Menambahkan X pada Q dengan aturan FIFO }

{ I.S. Q mungkin kosong, tabel penampung elemen Q TIDAK penuh. }

{ F.S. X menjadi TAIL yang baru, TAIL "maju". }

{ Jika TAIL = MaxEl, maka elemen-elemen Queue terlebih dahulu harus digeser. }

procedure Del (input/output Q : Queue, output X : infotype)
{ Proses : Menghapus elemen pertama pada Q dengan aturan FIFO }

{ I.S. Q tidak kosong }

{ F.S. X = nilai elemen HEAD pada I.S., }

```

```
{ Jika Queue masih ada isinya : HEAD "maju". }  
{ Jika Queue menjadi kosong, HEAD = TAIL = Nil. }
```

- a Buatlah realisasi prosedur AkuisisiEvent pada modul Mesin Event sesuai dengan spesifikasi yang diberikan (spesifikasi tidak perlu ditulis ulang). Tidak boleh menambahkan fungsi/prosedur baru.
- b Buatlah realisasi prosedur Add dan Del pada ADT Queue sesuai spesifikasi yang diberikan (spesifikasi tidak perlu ditulis ulang). Tidak boleh menambahkan fungsi/prosedur baru.
- c Buatlah program utama yang mensimulasikan sistem antrian dengan memanfaatkan ADT Mesin Event dan ADT Queue tersebut. Program akan:
- Membaca event dengan menggunakan Mesin Event
 - Melakukan pemrosesan terhadap CEvent yang berhasil dibaca.
 - o Apabila jenis event = 'A' maka elemen dengan nilai waktu event dimasukkan ke dalam antrian.
 - o Apabila jenis event = 'D' maka sebuah elemen dihapus dari antrian. Waktu tunggu untuk elemen tersebut adalah waktu event (elemen dihapus dari antrian) – waktu kedatangan elemen ke antrian (info dari elemen tersebut).
- Perhatikan persyaratan yang harus dipenuhi untuk mengeksekusi Add dan Delete elemen antrian.
- Program akan selesai pada saat mesin event selesai diakuisisi.
 - Pada akhir program akan ditampilkan (1) banyaknya elemen yang sudah selesai dilayani (sudah keluar dari antrian) dan (2) waktu tunggu rata-rata untuk seluruh elemen yang sudah dilayani tersebut, (3) jumlah elemen yang masih tersisa di antrian, serta (4) jumlah event yang tidak berhasil diproses (karena tidak memenuhi persyaratan untuk menjalankan event).

Soal 3. ADT Set: representasi List Linier dengan elemen unik dan terurut

Berikut ini definisi ADT Set dengan menggunakan representasi List Linier yang elemennya unik dan terurut.

```
{ MODUL ADT SET dengan representasi list linier berelemen unik dan terurut }  
{ Berisi definisi dan primitif pemrosesan ADT Set }  
{ Kamus Umum  
} constant Nil :  
...  
type elmttype : integer  
type elmtList : < Elmt : elmttype, Next : address >  
type address : pointer to elmtList  
{ Definisi elemen dan koleksi objek }  
type SET : < First : address>  
{ *** Selektor *** }  
{ Jika S : SET dan P : address, maka penulisan :  
    First(S) menjadi S.First;      Next(P) menjadi P + .Next; Elmt(P) menjadi P  
+ .Elmt }  
{ ***Konstruktor: create set kosong*** }  
procedure CreateEmpty (output S : SET)  
{ I.S. sembarang }  
{ F.S. Terbentuk himpunan kosong S }  
{ ***Primitif Alokasi dan Dealokasi *** }  
procedure Alokasi (output P : address, input X : elmttype)  
{ I.S. X terdefinisi }  
{ F.S. P berisi alamat elemen list dengan Elmt(P) = X dan Next(P)=Nil; P Nil jika alokasi  
tidak berhasil }  
procedure Dealokasi (input/output P : address)  
{ I.S. P terdefinisi }  
{ F.S. P didealokasi dan jadi Nil }
```

```
{ ***Primitif Set*** }

function IsEmpty (S : SET) → boolean
{ Mengirimkan true jika S adalah himpunan kosong. }

function IsMember (e : elmttype, S : SET) → boolean
{ Mengirimkan true jika e adalah elemen/anggota dari set S. }

function IsSubSet (S1, S2 : SET) → boolean
{ Mengirimkan true jika S1 merupakan himpunan bagian (subset) dari S2.
  S1 merupakan himpunan bagian S2 apabila semua elemen S1 juga merupakan elemen
  S2 }

function Union (S1, S2 : SET) → SET
{ mengirimkan set hasil operasi union (penggabungan) antara S1 dan S2. S1 dan S2
tidak
berubah. }

procedure Insert (input e : elmttype, input/output S : SET)
{ I.S. S terdefinisi, mungkin kosong }
{ F.S. elemen S bertambah satu elemen, yaitu e, jika sebelumnya e bukan elemen S.
  jika sebelumnya e sudah merupakan elemen S, S tetap.}
```

Realisasikan fungsi IsMember, fungsi IsSubSet, fungsi Union, dan prosedur Insert tanpa menulis ulang spesifikasi. Tidak diperkenankan menambahkan fungsi/prosedur antara baru.

UTS SEMESTER I – 2016/2017

BACA DULU PETUNJUK DI BAWAH INI:

1. Tuliskan NIM, nama, dan kelas pada lembar soal ini.
2. Lembar soal terdiri atas 6 halaman terdiri atas 3 soal di halaman 1 s.d. 6. Jika Anda belum mendapatkan lembar soal yang lengkap, tukarkan pada pengawas.
3. Tuliskan NIM, Nama, dan kelas di setiap lembar jawaban yang disediakan.
4. Jawaban soal dituliskan pada lembar jawaban yang disediakan. Jangan lupa memberikan nomor soal dengan jelas. Jika nomor soal tidak jelas, penilai berhak tidak menilai pekerjaan Anda.
5. Setelah selesai ujian, masukkan lembar soal ke dalam lembar jawaban. Lembar jawaban dan lembar soal diserahkan pada pengawas.

Soal 1. ADT Matriks [Notasi Algoritmik] (Bobot : 25%)

Diberikan definisi dan spesifikasi ADT matriks sebagai berikut:

```
{ ADT Matriks }

type indeks : integer { indeks baris dan kolom }
constant BrsMin : indeks = 1
constant BrsMax : indeks = 100
constant KolMin : indeks = 1
constant KolMax : indeks = 100
type el_type : integer
type MATRIKS :
  < Mem : matrix[BrsMin..BrsMax,KolMin..KolMax] of el_type,
    NBrsEff : integer > 0, { banyaknya/ukuran baris yg terdefinisi }
    NKolEff : integer > 0 { banyaknya/ukuran kolom yg terdefinisi } >
  { Memori matriks yang dipakai selalu di "pojok kiri atas" }

{***** DEFINISI PROTOTIPE PRIMITIF *****}
{*** Konstruktor membentuk MATRIKS ***}
procedure CreateMATRIKS (input NB, NK : integer, output M : MATRIKS)
{ Membentuk sebuah MATRIKS "kosong" berukuran NBxNK di "pojok kiri atas" memori }
{ I.S. NB dan NK adalah valid untuk memori matriks yang dibuat }
{ F.S. MATRIKS M terdefinisi dengan ukuran NBxNK }

{*** Selektor MATRIKS ***}
{*** Selektor Get ***}
function GetNBrsEff (M : MATRIKS) → integer
{ Mengirimkan banyaknya baris efektif M }
function GetNKolEff (M : MATRIKS) → integer
{ Mengirimkan banyaknya kolom efektif M }
function GetElmt (M : MATRIKS; i, j : indeks) → el_type
{ Mengirimkan elemen M dengan nomor baris i dan nomor kolom j }
function GetFirstIdxBrs (M : MATRIKS) → indeks
{ Mengirimkan indeks baris terkecil M }
function GetFirstIdxKol (M : MATRIKS) → indeks
{ Mengirimkan indeks kolom terkecil M }
function GetLastIdxBrs (M : MATRIKS) → indeks
{ Mengirimkan indeks baris terbesar M }
function GetLastIdxKol (M : MATRIKS) → indeks
{ Mengirimkan indeks kolom terbesar M }

{*** Selektor Set: Operasi mengubah nilai elemen matriks ***}
procedure SetElmt (input/output M : MATRIKS; input i, j : integer; input X : el_type)
{ I.S. M sudah terdefinisi; F.S. M[i,j] bernilai X }
```

Dengan menggunakan ADT di atas, buatlah implementasi prosedur dan fungsi di bawah ini. Spesifikasi prosedur/fungsi yang sudah didefinisikan di atas tidak perlu ditulis ulang. Jika Anda perlu membuat fungsi/prosedur baru, harus ditulis definisi, spesifikasi, dan realisasinya.

a. Procedure Find Element

```
procedure FindElement (input M : MATRIKS, input X : el_type,  

                      output brs, kol : indeks)  

{ Procedure FindElement melakukan proses untuk menemukan indeks di mana elemen X  

pertama kali ditemukan pada matriks M. Jika X ditemukan pada M, maka brs dan kol  

adalah indeks di mana X ditemukan pada M. Jika X tidak ditemukan, brs=-1, kol=-1.  

Jika X terdapat di M lebih dari satu kali, maka indeks yang dihasilkan (brs dan  

kol) adalah indeks "terendah", yaitu baris terendah diperhitungkan lebih dulu  

daripada kolom. }  

{ Misal: X berada di M pada [brs,kol] = [3,4] dan [4,3], maka brs=3, kol=4 }  

{ Misal: X berada di M pada [brs,kol] = [1,4] dan [2,1], maka brs=1, kol=4 }  

{ Misal: X berada di M pada [brs,kol] = [2,4] dan [2,5], maka brs=2, kol=4 }  

{ Misal: X berada di M pada [brs,kol] = [2,1] dan [1,10], maka brs=1, kol=10 }  

{ I.S.: M terdefinisi, X terdefinisi }  

{ F.S.: brs berisi indeks baris terendah dan kol berisi indeks kolom terendah di mana X  

ditemukan di M, atau brs=-1 dan kol=-1 jika X tidak ditemukan di M }
```

b. Function Sub Matriks

```
function SubMatriks (M : MATRIKS, RDel: integer, CDel: integer) → MATRIKS  

{ Function SubMatriks melakukan operasi menghasilkan/mengembalikan matriks yang merupakan  

submatriks dari matriks M dengan cara menghapus baris RDel dan kolom CDel. }  

{ Prekondisi: M adalah sebuah matriks dengan ukuran baris kolom minimal 2x2. RDel dan  

CDel adalah indeks baris dan kolom yang valid pada matriks M. }
```

Ilustrasi dan contoh SubMatriks:

Matriks M	Hasil/Output Fungsi				
<table border="1"> <tr><td>4</td><td>5</td></tr> <tr><td>2</td><td>6</td></tr> </table>	4	5	2	6	=> 6
4	5				
2	6				

RDel = 1
CDel = 1

Matriks M	Hasil/Output Fungsi													
<table border="1"> <tr><td>3</td><td>2</td><td>3</td></tr> <tr><td>4</td><td>2</td><td>1</td></tr> <tr><td>5</td><td>6</td><td>5</td></tr> </table>	3	2	3	4	2	1	5	6	5	<table border="1"> <tr><td>3</td><td>3</td></tr> <tr><td>5</td><td>5</td></tr> </table>	3	3	5	5
3	2	3												
4	2	1												
5	6	5												
3	3													
5	5													

RDel = 2
CDel = 2

Soal 2. Mesin Kata dan ADT Queue [Notasi Algoritmik] (Bobot : 50%)

Sebuah mesin kata model 1 yang telah dimodifikasi menjadi mesin pencetakan, yang digunakan untuk membaca sebuah pita karakter yang terdiri atas kelompok "kata" yang merepresentasikan permintaan pencetakan dokumen yang terdiri dari **id**, **prioritas**, dan **banyaknya lembar** yang harus dicetak. Pita diakhiri dengan '.'. Beberapa contoh isi pita karakter adalah sebagai berikut:

1-5-15 12-3-25 3-4-20.
.
13-4-10 2-3-25 10-5-20 .
2-1-10.

Pita karakter diasumsikan selalu dalam format yang benar, yaitu berisi "kata" yang merepresentasikan data pencetakan yang **tidak terurut**. Format "kata" yang dibaca adalah: <**id**>-<**prioritas**>-<**jumlah lembar**>, dengan <**id**> bertipe integer > 0 dengan digit bisa lebih dari satu; <**prioritas**> bertipe integer > 0 hanya satu digit; <**jumlah lembar**> bertipe integer > 0 dengan digit bisa lebih dari satu. Tidak ada bentuk "kata" lain yang muncul dalam pita. Pita kosong hanya terdiri atas '.' (mungkin didahului sejumlah spasi/blank).

Hasil akuisisi "kata" dalam mesin ini adalah sebuah data permintaan disimpan pada **CData**.

Berikut adalah definisi **Mesin Pencetakan** (hasil modifikasi Mesin Kata model 1):

```
{ ***** Mesin Pencetakan ***** }
use MESINKARAKTER
{ Primitif Mesin karakter adalah: START dan ADV.
  State Mesin karakter adalah: CC (karakter hasil akuisisi) dan
                                EOP (true jika CC = MARK) }
{ ***** Konstanta ***** }
constant MARK : character = '.'
constant BLANK : character = ' '

{ Type dataCetak }
type dataCetak : <id : integer; prioritas : integer; lembar : integer>

{ ***** Definisi State Mesin ***** }
CData : dataCetak
EndData : boolean

{ ***** Primitif-Primitif Mesin Integer ***** }
procedure START-DATA
{ I.S. : CC sembarang }
{ F.S. : EndData = true, dan CC = MARK; }
{     atau EndData = false, CData adalah karakter permintaan yang sudah
       diakuisisi, CC karakter pertama sesudah karakter terakhir data permintaan
       pencetakan}
procedure ADV-DATA
{ I.S. : EndData = false; CC adalah karakter sesudah karakter terakhir yang sudah
       diakuisisi }
{ F.S. : Jika CC = MARK, maka EndData = true; }
{     atau EndData = false, CData adalah karakter terakhir yang sudah diakuisisi;
       CC karakter pertama sesudah karakter terakhir yang diakuisisi }
procedure IgnoreBlank
{ I.S. : CC sembarang }
{ F.S. : CC ≠ BLANK atau CC = MARK }
procedure SalinData
{ Mengakuisisi kata sambil mengkonversinya menjadi data permintaan pencetakan, dan
   menyimpannya dalam CData. }
{ Proses akuisisi tanpa mark, artinya selalu ada data permintaan pencetakan yang
   diakuisisi. }
{ I.S. : CC adalah karakter pertama dari kata. }
{ F.S. : CData berisi data permintaan pencetakan yang sudah diakuisisi; }
{     CC = BLANK atau CC = MARK; }
{     CC adalah karakter sesudah karakter terakhir yang diakuisisi. }
{     Proses akuisisi harus menggunakan fungsi KarakterToInteger. }
function KarakterToInteger (C : character) → integer
{ Mengirimkan integer [0..9] yang merupakan konversi dari character C yang merupakan
   salah satu dari karakter ['0'..'9']. }
```

Diketahui sebuah ADT Priority Queue yang direpresentasikan dengan array statik dengan model implementasi alternatif 1 (HEAD selalu pada indeks 1, sehingga elemen-elemen selalu digeser setiap ada elemen yang dihapus dari antrian) dengan elemen yang selalu terurut membesar berdasarkan prioritasnya, sebagai berikut:

```

{ *** ADT Queue, model implementasi alternatif 1 *** }
use MESIN-PENCETAKAN

{ *** Konstanta *** }
constant Nil : integer = 0
constant MaxEl : integer = ...

{ *** Definisi elemen dan address *** }
type address : integer { indeks tabel }

{ *** Definisi Type Queue *** }
type Queue : < T : array [1..MaxEl] of dataCetak, { tabel penyimpan elemen }
            HEAD : address, { alamat penghapusan }
            TAIL : address { alamat penambahan } >
{ Jika Q adalah Queue, maka: }
{ Definisi Queue kosong: Q.HEAD = Nil; Q.TAIL = Nil. }
{ Selektor: wajib digunakan sebagai selektor get maupun set (pada bagian left-hand-side
ataupun right-hand-side dari ekspresi):
    Head(Q) adalah Q.HEAD
    Tail(Q) adalah Q.TAIL
    InfoHead(Q) adalah Q.TQ.HEAD
    InfoTail(Q) adalah Q.TQ.TAIL
    Elmt(Q,i) adalah Q.Ti     }

{ *** Predikat Pemeriksaan Kondisi Queue *** }
function IsEmpty (Q : Queue) → boolean
{ Mengirim true jika Q kosong }
function IsFull (Q : Queue) → boolean
{ Mengirim true jika tabel penampung elemen Q sudah penuh yaitu mengandung MaxEl elemen }
function NbElmt (Q : Queue) → integer
{ Mengirimkan banyaknya elemen queue. Mengirimkan 0 jika Q kosong. }

{ *** Konstruktor *** }
procedure CreateEmpty (output Q : Queue)
{ I.S. Q sembarang
  F.S. Sebuah Q kosong terbentuk dengan HEAD = Nil dan TAIL = Nil }

{ *** Operator-Operator Dasar Queue *** }
procedure Add (input/output Q : Queue, input X : dataCetak)
{ Proses : Menambahkan X pada Q dengan aturan FIFO terurut membesar berdasarkan
prioritas
  I.S. Q mungkin kosong, tabel penampung elemen Q TIDAK penuh.
  F.S. X menjadi elemen Q; Q tetap terurut membesar menurut prioritasnya. }
procedure Del (input/output Q : Queue, output X : dataCetak)
{ Proses : Menghapus elemen pertama pada Q dengan aturan FIFO
  I.S. Q tidak kosong
  F.S. X = nilai elemen HEAD pada I.S.,
       Jika Queue masih isi : elemen-elemen yang berada di belakang HEAD pada I.S.
       digeser sehingga HEAD tetap 1 dan TAIL "mundur".
       Jika Queue menjadi kosong, HEAD = TAIL = Nil. }

```

- Buatlah realisasi prosedur **ADV-DATA** dan **SalinData** pada mesin pencetakan sesuai dengan spesifikasi yang diberikan (spesifikasi tidak perlu ditulis ulang). Tidak boleh menambahkan fungsi/prosedur baru.
- Buatlah realisasi prosedur **Add** dan **Del** pada ADT Queue sesuai spesifikasi yang diberikan (spesifikasi tidak perlu ditulis ulang). Tidak boleh menambahkan fungsi/prosedur baru.
- Lengkapi **program utama** berikut ini, yang membaca sebuah pita karakter dengan menggunakan mesin pencetakan dan memasukkan isinya ke dalam sebuah Queue. Procedure Cetak tidak perlu dibuat realisasinya. Untuk menuliskan jawaban, tuliskan hanya bagian utama program (spesifikasi program/procedure tidak perlu ditulis ulang).

Program QUEUE_PENCETAKAN

```
{ Input: Pita karakter (yang dibaca menggunakan mesin pencetakan).
Proses: Baca seluruh data permintaan pencetakan dari pita karakter dengan
        menggunakan mesin pencetakan. Setiap data yang berhasil dibaca dari pita
        dimasukkan ke dalam queue sehingga datanya terurut membesar menurut
        prioritasnya. Selanjutnya, lakukan proses pencetakan dengan pemanggil prosedur
        Cetak, untuk seluruh data permintaan pencetakan pada queue. Di akhir proses,
        tampilkan ke layar banyaknya permintaan pencetakan, banyaknya lembar pencetakan
        untuk seluruh data permintaan pencetakan, serta lembar pencetakan paling banyak
        Output: banyaknya permintaan pencetakan (banyaknya data cetak yang berhasil dibaca
                dari mesin pencetakan), banyaknya lembar pencetakan untuk seluruh
                data permintaan pencetakan, dan lembar pencetakan paling banyak (maksimal) dari
                seluruh data permintaan pencetakan. Misalnya, untuk contoh pita pertama di atas,
                akan tampil: 3, 60, 25
                Jika tidak ada data permintaan pencetakan yang, maka akan tampil: 0, 0, 0
}
```

```
USE MESIN-PENCETAKAN
USE QUEUE
```

KAMUS

```
nAkuisisi, nLembar, maxLembar : integer
AntrianCetak : Queue
.....
{ tambahkan variable lain yang dibutuhkan }

procedure Cetak (input DC : dataCetak)
{ I.S.: DC terdefinisi, DC adalah data cetak yang akan diproses
  F.S.: Pencetakan sudah dilakukan sesuai dengan data pencetakan pada DC}
```

ALGORITMA

```
{ mengisi AntrianCetak dengan data dari Mesin Pencetakan }

...
{ melakukan proses pencetakan untuk seluruh data pencetakan pada AntrianCetak }

...
{ menampilkan informasi ke layar }

...
procedure Cetak (input DC : dataCetak)
{ I.S.: DC terdefinisi, DC adalah data cetak yang akan diproses
  F.S.: Pencetakan sudah dilakukan sesuai dengan data pencetakan pada DC }
{ Realisasi procedure Cetak - TIDAK PERLU DIBUAT }
...
```

Soal 3. ADT MAP [Notasi Algoritmik] (Bobot : 25%)

Map adalah koleksi objek yang terdiri atas pasangan/*binding* (*key, value*) dengan nilai *key* unik dalam koleksi. Pada soal ini, ADT Map diimplementasikan sebagai array kontigu dengan representasi eksplisit dan alokasi statik. Memori array yang digunakan adalah [IdxMin..IdxMax]. Map kosong didefinisikan dengan NbElmt = 0.

Berikut adalah definisi ADT Map:

```

{ ADT Map }
{ Implementasi dengan array kontigu dengan representasi eksplisit }
type IdxType : integer { type indeks array }
type KeyType : ... { type key dari elemen map }
type InfoType : ... { type value dari elemen map }

constant IdxMin : IdxType = ... { indeks minimum memory array }
constant IdxMax : IdxType = ... { indeks maksimum memory array }
constant IdxUndef : IdxType = ... { indeks array tidak terdefinisi }
constant ValUndef : InfoType = ... { value elemen map tidak terdefinisi }

type ElmtMap : < key : KeyType, value : InfoType > { type elemen map }

{ Type koleksi objek Map }
type Map : < T : array [IdxMin..IdxMax] of ElmtMap,
NbElmt : integer ≥ 0 >

{ Jika M adalah Map, maka berikut adalah beberapa definisi: }
{ Map kosong adalah M.NbElmt = 0 }
{ Selektor: wajib digunakan sebagai selektor get maupun set (pada bagian left-hand-side ataupun right-hand-side dari ekspresi):
  NbElmt(M) adalah M.NbElmt
  Elmt(M,i) adalah M.Ti
  Key(M,i) adalah M.Ti.key
  Value(M,i) adalah M.Ti.value }

{ Operasi-Operasi pada Map }
procedure CreateEmpty-(output M : Map)
{ Membentuk M kosong. I.S. M sembarang; F.S. M kosong terbentuk. }

function IsEmpty (M : Map) → boolean
{ Menghasilkan true jika M adalah Map kosong, false jika tidak }

function IsFull (M : Map) → boolean
{ Menghasilkan true jika M sudah penuh, false jika tidak }

function LookUpIdx (M : Map, k : KeyType) → IdxType
{ Menghasilkan indeks array container M dari elemen dengan key = k jika k ada di M.
  Jika k tidak ada di M, menghasilkan IdxUndef. }

procedure Add (input/output M : Map, input k : KeyType, input v : InfoType)
{ Menambahkan pasangan (k,v) sebagai elemen M. Jika k belum ada di M, maka pasangan (k,v) ditambahkan sebagai elemen terakhir M. Jika k sudah ada di M, misalnya pada elemen E, maka dilakukan operasi reassign, yaitu E.value diganti dengan v. }
{ I.S. M mungkin kosong, tidak penuh. }
{ F.S. Jika k belum ada di M, maka pasangan (k,v) ditambahkan sebagai elemen terakhir M.
  Jika k sudah ada di M, maka nilai value-nya di-reassign dengan v. }

procedure Remove (input/output M : Map, input k : KeyType, output v : InfoType)
{ Menghapus elemen, misalnya E, dengan E.key = k jika k ada di M dan menyimpan E.value di v. Agar array tetap kontigu, elemen terakhir M dipindahkan ke posisi E yang dihapus.
  Jika k tidak ada di M, maka M tetap dan v berisi ValUndef. }
{ I.S. M mungkin kosong. }
{ F.S. Jika k ada di M, maka elemen dengan key = k dihapus dan v berisi bagian value elemen tsb. Jika k tidak ada di M, maka M tetap. }

procedure Merge (input M1 : Map, input M2 : Map, input prio : integer[1,2],
                output M3 : Map)
{ M3 diisi dengan gabungan nilai elemen M1 dan M2. Elemen M1 diprioritaskan untuk dikopikan ke M3 terlebih dahulu, baru elemen M2. Jika ada elemen M1 dan M2 dengan key yang sama, maka: jika prio=1, maka value yang diambil adalah value dari elemen M1; jika prio=2, maka value yang diambil adalah value dari elemen M2.
  M3 diisi dengan semua nilai elemen M1 dan M2 atau sampai M3 penuh. }
{ I.S. M1, M2 mungkin kosong, M3 sembarang. F.S. M3 berisi gabungan elemen M1 dan M2. }

```

Buatlah implementasi fungsi **LookUpIdx**, procedure **Add**, **Remove**, dan **Merge** sesuai dengan definisi ADT Map dan spesifikasi di atas (spesifikasi tidak perlu ditulis ulang). Fungsi/prosedur lain tinggal dipakai, tidak perlu direalisasikan. Tidak boleh membuat type/fungsi/prosedur tambahan. Gunakan sebaik-baiknya primitif-primitif yang disediakan pada ADT tersebut.

UTS SEMESTER I - 2019/2020

BACA DULU PETUNJUK DI BAWAH INI:

1. Lembar soal terdiri atas 4 soal dalam 6 halaman. Jika Anda belum mendapatkan lembar soal yang lengkap, tukarkan pada pengawas.
2. Tuliskan NIM, Nama, dan kelas di setiap lembar jawaban yang disediakan.
3. Jawaban soal dituliskan pada lembar jawaban yang disediakan. Jangan lupa memberikan nomor soal dengan jelas. Jika nomor soal tidak jelas, penilai berhak tidak menilai pekerjaan Anda.
4. Setelah selesai ujian, lembar jawaban diberikan pada pengawas.

Soal 1. ADT Matriks [Notasi Algoritmik] (Bobot: 25%)

Diberikan definisi dan spesifikasi ADT Matriks sebagai berikut:

```

{ ***** ADT Matriks ***** }
type indeks: integer { indeks baris dan kolom }
constant BrsMin: indeks = 1
constant BrsMax: indeks = 100
constant KolMin: indeks = 1
constant KolMax: indeks = 100
type el_type: integer
type Matriks:
    < Mem: matrix[BrsMin..BrsMax,KolMin..KolMax] of el_type,
      NBrsEff: integer > 0, { banyaknya/ukuran baris yg terdefinisi }
      NKolEff: integer > 0 { banyaknya/ukuran kolom yg terdefinisi } >
    { Memori matriks yang dipakai selalu di "pojok kiri atas" }

{***** DEFINISI PROTOTIPE PRIMITIF *****}
{*** Konstruktor membentuk Matriks ***}
procedure CreateMatriks (input NB, NK: integer, output M: Matriks)
{ Membentuk sebuah Matriks "kosong" berukuran NBxNK di "pojok kiri atas" memori }
{ I.S. NB dan NK adalah valid untuk memori matriks yang dibuat }
{ F.S. Matriks M terdefinisi dengan ukuran NBxNK }

{*** Selektor Matriks ***}
{*** Selektor Get ***}
function GetNBEff (M: Matriks) + integer
{ Mengirimkan banyaknya baris efektif M }
function GetNKEff (M: Matriks) + integer
{ Mengirimkan banyaknya kolom efektif M }
function GetFirstIdxB (M: Matriks) + indeks
{ Mengirimkan indeks baris terkecil M }
function GetFirstIdxK (M: Matriks) + indeks
{ Mengirimkan indeks kolom terkecil M }
function GetLastIdxB (M: Matriks) + indeks
{ Mengirimkan indeks baris terbesar M }
function GetLastIdxK (M: Matriks) + indeks
{ Mengirimkan indeks kolom terbesar M }
{*** Selektor Get dan Set: Operasi mengubah nilai elemen matriks ***}
function GetElmt (M: Matriks; i, j: integer) + el_type
{ Mengirimkan elemen matriks M[i,j] }
procedure SetElmt (input/output M: Matriks; input i, j: integer; input X: el_type)
{ I.S. M sudah terdefinisi; F.S. M[i,j] bernilai X }

```

Dengan menggunakan ADT di atas, buatlah implementasi a) prosedur SubMatriks dan b) fungsi IsMagicSquare berikut ini. Spesifikasi prosedur/fungsi yang sudah didefinisikan pada ADT tidak perlu ditulis ulang. Tidak diperkenankan membuat fungsi/prosedur tambahan lainnya.

```

procedure SubMatriks (input M: Matriks, input b, k: indeks, output MHsl: Matriks)
{ I.S. Matriks M serta indeks b dan k terdefinisi;
  GetFirstIdxB(M) ≤ b ≤ GetLastIdxB(M) dan GetFirstIdxK(M) ≤ k ≤ GetLastIdxK(M) }
{ F.S. MHsl (ukuran b x k) berisi seluruh elemen matriks M mulai dari baris
  GetFirstIdxB(M) hingga baris b dan kolom GetFirstIdxK(M) hingga kolom k }

```

Soal 2. ADT Queue [Notasi Algoritmik] (Bobot: 25%)

Dalam suatu antrian pembayaran supermarket, kadang kala dilakukan pemecahan (*branching*) atau penggabungan (*merging*) beberapa antrian, tergantung kebutuhan. Dalam hal ini struktur data queue bisa digunakan untuk menggambarkan proses penggabungan dan pemecahan antrian tersebut. Agar *fair*, penggabungan queue dilakukan dengan cara mengisi queue yang baru secara bergantian antar elemen dari dua queue yang berbeda, jika ada sisa dari salah satu queue, maka akan ditambahkan di belakangnya. Untuk pemecahan queue, dilakukan pengisian bersilangan ke Q1 dan Q2. Jika jumlah awal elemennya ganjil, maka Q1 akan 1 lebih banyak dari Q2, jika jumlah awal elemennya genap, maka akan terdistribusi sama banyak antara Q1 dan Q2. Asumsi Q3 akan selalu bisa menampung seluruh elemen Q1 dan Q2 (tanpa perlu mengecek apakah Q3 full atau tidak).

Contoh:

Merge Queue

Q1	<table border="1" style="display: inline-table;"><tr><td>11</td><td>27</td><td>58</td><td>49</td><td>55</td><td>63</td></tr></table>	11	27	58	49	55	63	Q2	<table border="1" style="display: inline-table;"><tr><td>77</td><td>54</td><td>30</td></tr></table>	77	54	30
11	27	58	49	55	63							
77	54	30										
Q3			<table border="1" style="display: inline-table;"><tr><td>11</td><td>77</td><td>27</td><td>54</td><td>58</td><td>30</td><td>49</td><td>55</td><td>63</td></tr></table>	11	77	27	54	58	30	49	55	63
11	77	27	54	58	30	49	55	63				

Branch Queue

Q3	<table border="1" style="display: inline-table;"><tr><td>11</td><td>77</td><td>27</td><td>54</td><td>58</td><td>30</td><td>49</td><td>55</td><td>63</td></tr></table>	11	77	27	54	58	30	49	55	63
11	77	27	54	58	30	49	55	63		
Q1	<table border="1" style="display: inline-table;"><tr><td>11</td><td>27</td><td>58</td><td>49</td><td>63</td></tr></table>	11	27	58	49	63				
11	27	58	49	63						

Q2

77	54	30	55
----	----	----	----

Diberikan definisi dan spesifikasi ADT Queue sebagai berikut:

```
{ Modul ADT Queue - Alternatif I }
{ *** Deklarasi Queue yang diimplementasi dengan tabel kontigu *** }
{ *** HEAD dan TAIL adalah alamat elemen pertama dan terakhir *** }
{ *** Queue mampu menampung MaxEl buah elemen *** }
{ *** Konstanta *** }
constant Nil: integer = 0
{ *** Definisi elemen dan address *** }
type infotype: integer
type address: integer { indeks tabel }

{ *** Definisi Type Queue *** }
type Queue: < T: ..., { tabel penyimpan elemen, tergantung bahasa }
           HEAD: address, { alamat penghapusan }
           TAIL: address, { alamat penambahan }
           MaxEl: integer { maksimum banyaknya elemen queue } >
{ Definisi Queue kosong: Head = Nil; TAIL = Nil. }
{ Catatan implementasi: T[0] tidak pernah dipakai }
{ Definisi akses dengan Selektor:
  Jika Q adalah Queue dan E adalah elemen Queue:
  Head(Q), Tail(Q), MaxEl(Q) }
{ *** Predikat Pemeriksaan Kondisi Queue *** }
function IsEmpty (Q: Queue) → boolean
{ Mengirim true jika Q kosong }
```

```

function IsFull (Q: Queue) -> boolean
{ Mengirim true jika tabel penampung elemen Q sudah penuh yaitu mengandung MaxEl elemen }
function NBElmt (Q: Queue) -> integer
{ Mengirimkan banyaknya elemen queue. Mengirimkan 0 jika Q kosong. }
{ *** Konstruktor dan Destruktor *** }
procedure CreateEmpty (output Q: Queue, input Max: integer > 0)
{ I.S. Max terdefinisi }
{ F.S. Sebuah Q kosong terbentuk dan salah satu kondisi sbb:
    { Jika alokasi berhasil, tabel memori dialokasi berukuran Max
    { atau: jika alokasi gagal, Q kosong dg Maksimum elemen=0
    { Proses: Melakukan alokasi memori dan membuat sebuah Q kosong
procedure DeAlokasi (input/output Q: Queue)
{ Proses: Mengembalikan memori Q }
{ I.S. Q pernah dialokasi }
{ F.S. Q menjadi tidak terdefinisi lagi, MaxEl(Q) diset 0
{ *** Operator-Operator Dasar Queue *** }
procedure Add (input/output Q: Queue, input X: infotype)
{ Proses: Menambahkan X pada Q dengan aturan FIFO }
{ I.S. Q mungkin kosong, tabel penampung elemen Q TIDAK penuh }
{ F.S. X menjadi TAIL yang baru, TAIL "maju" }
procedure Del (input/output Q: Queue, output X: infotype)
{ Proses: Menghapus elemen pertama pada Q dengan aturan FIFO }
{ I.S. Q tidak kosong }
{ F.S. X = nilai elemen HEAD pada I.S.,
    Jika Queue masih isi: HEAD diset tetap = 1, elemen-elemen setelah HEAD yang
    lama digeser ke "kiri", TAIL = TAIL - 1;Jika Queue menjadi kosong, HEAD=TAIL=Nil }

```

Buatlah implementasi kedua prosedur a) MergeQueue dan b) BranchQueue berikut ini (spesifikasi tidak perlu ditulis ulang).

```

Procedure MergeQueue (input Q1, Q2: Queue, output Q3: Queue)
{ Proses: mengecek apakah Q1 dan Q2 kosong atau tidak. Jika keduanya tidak kosong, maka Q3 akan
berisi elemen-elemen Q1 dan Q2 secara bersilangan. Elemen pertama Q3 berisi elemen
pertama Q1, elemen kedua Q3 adalah elemen pertama Q2, dan seterusnya. Jika jumlah
elemen Q1 dan Q2 tidak sama, maka sisanya akan dimasukkan di belakang elemen yang sudah
terisi secara bersilangan. Asumsi bahwa jumlah elemen Q3 pasti cukup untuk menampung
gabungan seluruh elemen Q1 dan Q2 sehingga tidak perlu dilakukan pengecekan apakah
penuh. }
{ I.S. Q1 dan Q2 terdefinisi, mungkin kosong. Q3 sembarang }
{ F.S. Q3 akan berisi elemen-elemen Q1 dan Q2 secara bersilangan }

Procedure BranchQueue (input Q3: Queue, output Q1, Q2: Queue)
{ Proses: mengecek apakah Q3 kosong atau tidak. Jika tidak kosong, maka elemen-elemen Q3 akan
dibagikan secara merata ke Q1 dan Q2 secara bersilangan. Jika jumlah elemen Q3 adalah
ganjil, maka Q1 akan mempunyai elemen lebih banyak daripada Q2. Jika jumlah elemen Q3
genap, maka jumlah elemen Q1 dan Q2 sama banyak }
{ I.S. Q3 terdefinisi, mungkin kosong. Q1 dan Q2 sembarang }
{ F.S. Q1 dan Q2 akan berisi elemen-elemen dari Q3 secara bersilangan}

```

Soal 3. ADT MAP [Notasi Algoritmik] (Bobot: 25%)

Map adalah koleksi pasangan *key* dan *value* di mana *key* bersifat unik. ADT Map dapat diimplementasikan dengan menggunakan Array Statik. Berikut ini adalah sebagian definisi dan spesifikasi dari ADT Map dengan *key* berupa integer dan *value* berupa string.

```
Kamus Umum
constant MaxEl: integer = 100 { Banyaknya elemen maksimum }
constant ValUndef: integer = -999
type keytype: integer
type infotype: string
type mapEntry: { key: keytype, value: infotype } { elemen map }
type address: integer { indeks tabel }
type Map:
{ T: array [1..MaxEl] of mapEntry, { tabel penyimpan elemen map }
  NbElmt: integer { jumlah elemen map } }
```

{ Deklarasi: M: Map, E: mapEntry
 Definisi akses (diimplementasikan sebagai selektor):
 Key(E) adalah E.key, Value(E) adalah E.value
 NbMember(M) adalah M.NbElmt }

```
procedure CreateEntry(output E:mapEntry, input k:keytype, input v:infotype)
{ I.S. E sembarang, k dan v terdefinisi.  

  F.S. Terbentuk mapEntry E dengan Key(E) = k dan Value(E) = v }
procedure CreateMap(output M:Map)
{ I.S. M sembarang; F.S. Terbentuk Map kosong M }
procedure Put(input/output M:Map, input k:keytype, input v:infotype)
{ I.S. M terdefinisi, tidak penuh. k dan v terdefinisi.  

  F.S. Terdapat mapEntry dari pasangan k,v di dalam M. Jika sebelumnya sudah ada mapEntry E  

  anggota M dengan Key(E) = k, maka entri tersebut ditimpas dengan entri yang baru. }
function Get(M:Map, k:keytype) → infotype
{ Prekondisi: M dan k terdefinisi.  

  Mengembalikan Value(E) di mana E adalah anggota M dan Key(E) = k.  

  Jika M tidak memiliki anggota dengan key k, kembalikan ValUndef. }
```

Pada soal ini, sebuah Map akan diciptakan untuk menyimpan pasangan *HTTP Status Code* dan deskripsi singkatnya. Contoh, jika pada Map kosong M dilakukan Put(M, 404, "Not Found"); Put(M, 503, "Service Unavailable"); Put(M, 200, "OK"); maka Get(M, 404) akan mengembalikan "Not Found" dan Get(M, 700) akan mengembalikan ValUndef. Anda bebas menentukan urutan peletakan entri Map di dalam array.

Tugas Anda adalah membuat realisasi a) prosedur Put dan b) fungsi Get tanpa menulis ulang spesifikasinya, serta sebuah c) program utama yang:

1. membuat sebuah Map kosong,
2. mengisinya dengan beberapa pasang *HTTP Status Code* dan deskripsi singkatnya (gunakan contoh di atas),
3. membaca masukan sebuah integer dari pengguna, yaitu sebuah *HTTP Status Code*, dan
4. mencetak deskripsi singkat dari *HTTP Status Code* yang diberikan pengguna berdasarkan Map yang sudah terbentuk. Jika masukan pengguna tidak ada di dalam Map, cetak "HTTP Status Code yang dimasukkan tidak valid."

Soal 4. Mesin Kata dan Stack [Notasi Algoritmik] (Bobot: 25%)

Sebuah mesin token (modifikasi dari mesin kata model 1) digunakan untuk membaca sebuah pita karakter yang terdiri atas kelompok "kata" yang merepresentasikan ekspresi dalam notasi infix dan diakhiri dengan '.'. Operan yang diterima adalah bilangan bulat positif dan 0, sedangkan operator yang diterima adalah +, -, *, /, dan ^.

Beberapa contoh isi pita karakter adalah sebagai berikut.

(10 + 2) ^ 2 - 16 * 5 .

.

(123 - 140) / 7 + 15 * 16 .

Pita karakter diasumsikan selalu dalam format yang benar ("kata" selalu dikenali dan ekspresi selalu benar) atau merupakan pita kosong hanya terdiri atas '.' (mungkin didahului sejumlah spasi/blank).

Hasil akuisisi "kata" dalam mesin ini adalah sebuah token CToken yang terdiri dari dua komponen <op,val>, dengan op berisi operan yang dibaca (bernilai OpUndef jika yang dibaca adalah bilangan) dan val berisi nilai dari bilangan yang dibaca (bernilai ValUndef jika yang dibaca adalah operator).

Contoh: token-token yang dihasilkan dari pita pertama berturut-turut adalah <'(',ValUndef,>, <OpUndef,10>, <'+',ValUndef,>, <OpUndef,2>, <')',ValUndef,>, <'^',ValUndef,>, <OpUndef,2>, <'-',ValUndef,>, <OpUndef,16>, <'*',ValUndef,>, <OpUndef,5>.

Berikut adalah definisi Mesin Token (hasil modifikasi Mesin Kata model 1):

```
{ ***** Mesin Token ***** }
use MESINKARAKTER
{ Primitif Mesin karakter adalah: START dan ADV.
  State Mesin karakter adalah: CC (karakter hasil akuisisi) dan
                                EOP (true jika CC = MARK) }

{ *** Konstanta *** }
constant MARK: character = '.'
constant BLANK: character = ' '
constant OpUndef: character = '#'
constant ValUndef: integer = -9999
{ *** Definisi State Mesin *** }
type Token = < op: character, {operator yang diakuisisi, OpUndef jika bilangan}
               val: integer {nilai bilangan/operan, ValUndef jika operator} >
CToken: Token
EndToken: boolean { penanda akhir akuisisi }

{ *** Primitif-Primitif Mesin Token *** }
procedure STARTTOKEN
{ I.S.: CC sembarang }
{ F.S.: Jika CC = MARK: EndToken = true, CToken tidak terdefinisi; }
{     Jika CC ≠ MAEK: EndToken = false, CToken = token yang selesai diakuisisi,
      CC karakter pertama sesudah karakter terakhir yang diakuisisi }

procedure ADVTOKEN
{ I.S.: EndToken = false; CC karakter sesudah karakter terakhir yang sudah diakuisisi }
{ F.S.: Jika CC = MARK: EndToken = true, CToken tidak terdefinisi; }
{     Jika CC ≠ MARK: EndToken = false, CToken = token yang selesai diakuisisi,
      CC karakter pertama sesudah karakter terakhir yang diakuisisi }

procedure SalinToken
{ Mengakuisisi token. Jika merupakan bilangan sambil dikonversi menjadi integer.
  Digunakan oleh prosedur STARTTOKEN dan ADVTOKEN. }
{ I.S.: CC adalah karakter pertama dari token (bukan MARK dan bukan BLANK). }
{ F.S.: CC = karakter sesudah karakter terakhir yang diakuisisi (BLANK atau MARK);
      CToken = token yang selesai diakuisisi; }
{     Jika token adalah integer, CToken.op=OpUndef, CToken.val=integer tsb. }
{     Jika token adalah operator, CToken.op=operator tersebut, CToken.val=ValUndef. }
{     Proses akuisisi untuk integer menggunakan fungsi KarakterToInteger. }

function KarakterToInteger (C: character) : integer
{ Mengirimkan integer [0..9] yang merupakan konversi dari character C yang merupakan salah satu
  dari karakter ['0'..'9']. }
```

Diketahui sebuah ADT Stack yang direpresentasikan dengan array statik sebagai berikut.

```
{
    ***** ADT Stack *****
    { *** Konstanta ***
    constant Nil: integer = 0
    constant MaxEl: integer = 10
    { *** Definisi elemen dan address ***
    type infotype: character
    type address : integer { indeks tabel }
    { *** Definisi Type Stack ***
    type Stack: < T : array [1..MaxEl] of infotype, { tabel penyimpan elemen }
        TOP: address { alamat top stack } >
    { Definisi Stack kosong: Top = Nil }
    { *** Konstruktor ***
    procedure CreateEmpty (output S: Stack)
    { I.S. Sembarang; F.S. Sebuah S kosong terbentuk dengan TOP = Nil }
    { *** Definisi Selektor Stack ***
    { Selektor tidak diberikan, akses elemen melalui Push dan Pop saja }
    { *** Predikat Pemeriksaan Kondisi Stack ***
    function IsEmpty (S: Stack) -> boolean
    { Mengirim true jika S kosong }
    function IsFull (S: Stack) -> boolean
    { Mengirim true jika tabel penampung elemen S sudah penuh yaitu mengandung MaxEl elemen }
    { *** Operator-Operator Dasar Queue ***
    procedure Push (input/output S: Stack, input X: infotype)
    { Proses: Menambahkan X pada S dengan aturan LIFO
    I.S. S mungkin kosong, tabel penampung elemen S TIDAK penuh.
    F.S. X menjadi TOP yang baru, TOP "maju". }
    procedure Pop (input/output S: Stack, output X: infotype)
    { Proses: Menghapus elemen pada S dengan aturan LIFO
    I.S. S tidak kosong
    F.S. X = nilai elemen TOP pada I.S., TOP "mundur". }
}
```

- Buatlah realisasi prosedur SalinToken pada mesin token sesuai dengan spesifikasi yang diberikan (spesifikasi tidak perlu ditulis ulang). Tidak boleh menambahkan fungsi/prosedur baru.
- Lengkapi program utama berikut ini, yang membaca sebuah pita karakter (berisi ekspresi infix) dengan menggunakan mesin token dan menuliskan kembali ekspresi tersebut ke layar dengan notasi postfix dengan memanfaatkan ADT Stack. Spesifikasi program tidak perlu ditulis kembali. Fungsi Prioritas tidak perlu dibuat realisasinya

```
program Infix2Postfix
{ Input: Pita karakter (yang dibaca menggunakan mesin token) berisi ekspresi infix.
Proses: Baca token dari pita karakter.
Jika yang dibaca integer, tampilkan ke layar.
Jika yang dibaca operator '(', tambahkan ke dalam stack.
Jika yang dibaca operator ')', hapus dan tampilkan semua operator dari
stack hingga menemukan '('.
Jika yang dibaca operator, tambahkan operator ke dalam stack dengan
terlebih dahulu menghapus dan menampilkan ke layar semua operator pada
stack yang memiliki prioritas lebih tinggi dari operator yang dibaca.
Setelah pembacaan pita karakter selesai, hapus dan tampilkan semua operator
yang masih tersisa di dalam stack.
Asumsi: Stack tidak pernah penuh, ekspresi pada pita karakter selalu benar
Output: ekspresi postfix untuk ekspresi yang dibaca dari pita karakter
jika pita karakter tidak mengandung ekspresi: pesan pita kosong. }
```

KAMUS

```

function Prioritas (op: character) -> integer
{ mengembalikan prioritas dari operator op.
Apabila operator op1 memiliki prioritas lebih tinggi dari op2:
    Prioritas(op1) > Prioritas(op2). }
```

ALGORITMA

...

Contoh: untuk masukan pita karakter $(10 + 2)^2 - 16 * 5$, akan dilakukan proses berikut:

Token	Proses	Isi Stack	Tampilan di Layar
<'(',ValUndef>	Tambah '(' ke stack	('')	
<OpUndef,10>	Tulis 10 ke layar	('')	10
<'+',ValUndef>	Tambah '+' ke stack	('', '+')	10
<OpUndef,2>	Tulis 2 ke layar	('', '+')	10 2
<')',ValUndef>	Hapus '+' dari stack dan tulis ke layar Hapus '(' dari stack	kosong	10 2 +
<'^',ValUndef>	Tambah '^' ke stack	('^')	10 2 +
<OpUndef,2>	Tulis 2 ke layar	('^')	10 2 + 2
<'-',ValUndef>	Hapus '^' dari stack dan tulis ke layar Tambah '-' ke stack	('-')	10 2 + 2 ^
<OpUndef,16>	Tulis 16 ke layar	('-')	10 2 + 2 ^ 16
<'*',ValUndef>	Tambah '*' ke stack	('-', '*')	10 2 + 2 ^ 16
<OpUndef,5>	Tulis 5 ke layar	('-', '*')	10 2 + 2 ^ 16 5
(EndToken=true)	Hapus '*' dari stack dan tulis ke layar Hapus '-' dari stack dan tulis ke layar	kosong	10 2 + 2 ^ 16 5 * -

UTS Algoritma Struktur Data 2020/2021

15 Oktober 2020

Petunjuk

- Waktu: 120 menit (13:00–15:00)
- Boleh membuka diktat Notasi Algoritmik
- Untuk menuliskan karakter spesial, dapat digunakan *shortcut*田; (di Windows), ^⌘Space (di Mac), atau silakan salin & tempel dari sini: → ← ≠ ≤ ≥

Bagian I. ADT Matriks (Notasi Algoritmik, bobot: 25%)

Diberikan definisi dan spesifikasi ADT Matriks sebagai berikut:

```

{ ADT Matriks }
type indeks: integer { indeks baris dan kolom }
constant BrsMin: indeks = 0
constant BrsMax: indeks = 99
constant KolMin: indeks = 0
constant KolMax: indeks = 99
type el_type: integer
type MATRIKS:
    < Mem: matrix[BrsMin..BrsMax,KolMin..KolMax] of el_type,
      NBrsEff: integer > 0, { banyaknya/ukuran baris yg terdefinisi }
      NKolEff: integer > 0 { banyaknya/ukuran kolom yg terdefinisi } >
{ Memori matriks yang dipakai selalu di "ujung kiri atas" }

{***** DEFINISI PROTOTIPE PRIMITIF *****}
{*** Konstruktor membentuk Matriks ***}
procedure MakeMATRIKS (input NB, NK: integer, output M: MATRIKS)
{ Membentuk sebuah Matriks "kosong" berukuran NB×NK di "ujung kiri atas"
memori.
I.S. NB dan NK adalah valid untuk memori matriks yang dibuat.
F.S. Matriks M terdefinisi dengan ukuran NB×NK. }

{*** Selektor Matriks ***}
{*** Selektor Get ***}
function GetNBrsEff (M: MATRIKS) → integer
{ Mengirimkan banyaknya baris efektif M }
function GetNKolEff (M: MATRIKS) → integer
{ Mengirimkan banyaknya kolom efektif M }
function GetFirstIdxBrs (M: MATRIKS) → indeks
{ Mengirimkan indeks baris terkecil M }
function GetFirstIdxKol (M: MATRIKS) → indeks
{ Mengirimkan indeks kolom terkecil M }
function GetLastIdxBrs (M: MATRIKS) → indeks
{ Mengirimkan indeks baris terbesar M }
function GetLastIdxKol (M: MATRIKS) → indeks
{ Mengirimkan indeks kolom terbesar M }

```

```

function GetElmtDiagonal (M: MATRIKS, i: indeks) → el_type
{ Mengirimkan elemen M(i,i) }
{*** Selektor Set: Operasi mengubah nilai elemen matriks ***}
function GetElmt (M: MATRIKS; i, j: integer) → el_type
{ Mengirimkan elemen matriks M[i,j] }
procedure SetElmt (input/output M: MATRIKS; input i, j: integer; input X: el_type)
{ I.S. M sudah terdefinisi; F.S. M[i,j] bernilai X }

```

Dengan menggunakan ADT di atas, buatlah implementasi prosedur dan fungsi di bawah ini. Spesifikasi prosedur/fungsi yang sudah didefinisikan pada ADT tidak perlu ditulis ulang. Tidak diperkenankan membuat fungsi/prosedur tambahan lainnya.

```

function IsSegitigaBawah (M: MATRIKS) → boolean
{ Mengirimkan true jika M adalah matriks segitiga bawah, dengan Mi,i = 0
untuk semua i < j.
Prekondisi: M adalah matriks bujur sangkar (berukuran n × n) }

procedure DetriMatriks (input M: MATRIKS, output Det: integer)
{ Menghitung nilai determinan matriks segitiga M, dimana determinant
dihitung berdasarkan perkalian semua elemen diagonalnya.
I.S. Matriks M terdefinisi;
F.S. Det berisi nilai determinan matriks segitiga bawah M atau 0 jika
bukan matriks segitiga bawah. }

```

Contoh:

Untuk matriks M1 dan M2 berikut:

M1:	1 0 0 0	M2:	7 12 0 0
	3 15 0 0		2 13 8 4
	-7 9 8 0		16 3 0 0
	-4 2 11 6		3 10 1 5

Maka:

- IsSegitigaBawah(M1) = true
- IsSegitigaBawah(M2) = false
- DetriMatriks(M1,Det) ⇒ Det=720
- DetriMatriks(M2,Det) ⇒ Det=0

Bagian II. Mesin Kata dan ADT Stack (Notasi Algoritmik, bobot: 25%)

Sebuah **mesin token** (**modifikasi dari mesin kata model 1**) digunakan untuk membaca sebuah pita karakter yang berisi data transaksi di sebuah ATM dalam satu hari, yang masing-masing direpresentasikan dalam sebuah token dan diakhiri dengan '.'. Sebuah token menggambarkan sebuah transaksi di ATM. Ada dua jenis transaksi yaitu: **tarik** dan **setor**. Beberapa contoh isi pita karakter adalah sebagai berikut:

tarik-2500000 tarik-500000 setor-250000 tarik-50000 .

.

setor-1000000.

Pita karakter diasumsikan selalu dalam format yang benar (tidak ada bentuk-bentuk "kata" lain yang yang tidak valid). Pita bisa merupakan pita kosong hanya terdiri atas '.' (mungkin didahului sejumlah spasi/blank). Hasil akuisisi "kata" dalam mesin ini adalah sebuah token **CToken** yang terdiri dari dua komponen **<opr, val>**, dengan **opr** berisi operan yang dibaca (bernilai "tarik" atau "setor") dan **val** berisi nilai uang yang ditarik atau disetor dari/ke ATM.

Contoh: token-token yang dihasilkan dari pita pertama berturut-turut adalah **<"tarik", 2500000>** , **<"tarik", 500000>** , **<"setor", 250000>** , **<"tarik", 50000>** .

Berikut adalah definisi Mesin Token (hasil modifikasi **Mesin Kata model 1**):

```

{ ***** MesinToken ***** }
use MESINKARAKTER
{ Primitif Mesin karakter adalah: START dan ADV.
  State Mesin karakter adalah: CC (karakter hasil akuisisi) dan
                                EOP (true jika CC = MARK) }

{ ***** Konstanta ***** }
constant MARK: character = '.'
constant BLANK: character = ' '

{ ***** Definisi State Mesin ***** }
type Token = < opr: string, { operator yang diakuisisi, bisa "tarik"
                               atau "setor" }
              val: integer { nilai uang yang ditarik atau disetor } >

CToken: Token
EndToken: boolean

{ ***** Primitif-Primitif Mesin Integer ***** }
procedure STARTTOKEN
{ I.S.: CToken sembarang }
{ F.S.: Jika CC = MARK, maka EndToken = true; }
{     atau EndToken = false, CToken adalah token yang sudah diakuisisi;
      CC karakter pertama sesudah karakter terakhir yang diakuisisi. }

procedure ADVTOKEN
{ I.S.: EndToken = false; CC adalah karakter sesudah karakter terakhir
      token yang sudah diakuisisi. }
{ F.S.: Jika CC = MARK, maka EndToken = true; }
{     atau EndToken = false, CToken adalah token yang sudah diakuisisi;
      CC karakter pertama sesudah karakter terakhir yang diakuisisi. }

procedure IgnoreBlank
{ I.S.: CC sembarang }
{ F.S.: CC <> BLANK atau CC = MARK. }

procedure SalinToken
{ Mengakuisisi token. Hasil akuisisi (token) disimpan ke dalam CToken. }
{ I.S.: CC adalah karakter pertama dari token. }
{ F.S.: CToken berisi token yang sudah diakuisisi; CC = BLANK
      atau CC = MARK;
      CC adalah karakter sesudah karakter terakhir yang diakuisisi;
      CToken bisa berupa <"tarik", nilai integer> atau
      <"setor", nilai integer>.
      Proses akuisisi untuk integer menggunakan fungsi KarakterToInteger. }

function KarakterToInteger (C: character) → integer

```

{ Mengirimkan integer [0..9] yang merupakan konversi dari character C yang merupakan salah satu dari karakter ['0 '..'9']. }

Diketahui sebuah ADT Stack yang direpresentasikan dengan **array statik** sebagai berikut:

```

{ *** ADT Stack *** }
{ *** Konstanta *** }
constant Nil: integer = 0
constant MaxEl: integer = 100
{ *** Definisi elemen dan address *** }
type infotype: integer
type address: integer { indeks tabel }
{ *** Definisi Type Queue *** }
type Stack: < T: array [1..MaxEl] of infotype, { tabel penyimpan elemen }
           TOP: address { alamat top stack } >
{ Definisi Stack kosong: Top = Nil }
{ *** Predikat Pemeriksaan Kondisi Stack *** }
function IsEmpty (S: Stack) → boolean
{ Mengirim true jika S kosong }
function IsFull (S: Stack) → boolean
{ Mengirim true jika tabel penampung elemen S sudah penuh yaitu mengandung
  MaxEl elemen }
{ *** Konstruktor *** }
procedure CreateEmpty (output S: Stack)
{ I.S. Sembarang
  F.S. Sebuah S kosong terbentuk dengan TOP = Nil }
{ *** Operator-Operator Dasar Stack *** }
procedure Push (input/output S: Stack, input X: infotype)
{ Proses: Menambahkan X pada S dengan aturan LIFO }
  I.S. S mungkin kosong, tabel penampung elemen S TIDAK penuh.
  F.S. X menjadi TOP yang baru, TOP "maju". }
procedure Pop (input/output S: Stack, output X: infotype)
{ Proses: Menghapus elemen pada S dengan aturan LIFO
  I.S. S tidak kosong
  F.S. X = nilai elemen TOP pada I.S., TOP "mundur". }

```

1. Buatlah realisasi prosedur **STARTTOKEN** dan **SalinToken** pada modul mesin token sesuai dengan spesifikasi yang diberikan (spesifikasi tidak perlu ditulis ulang). Tidak boleh menambahkan fungsi/prosedur baru.
2. Lengkapi program utama berikut ini, yang membaca sebuah pita karakter dengan menggunakan mesin token untuk mendapatkan nilai total setoran dikurangi total penarikan pada hari itu. Gunakan *stack of integer* untuk membantuk melakukan penghitungan. Spesifikasi program tidak perlu ditulis kembali.

```

program TransaksiATM
{ Input: Pita karakter (yang dibaca menggunakan mesin token).
Proses: Baca token dari pita karakter.
Selama Stack tidak penuh:
  Jika yang dibaca <"setor", nilai-integer>, push nilai
  integer tersebut ke dalam Stack
  Jika yang dibaca <"tarik", nilai-integer>, push nilai
  integer dikali (-1)
  ke dalam Stack.
Jika Stack penuh, tampilkan pesan "Stack penuh"

```

Jika Stack tidak penuh dan pita karakter selesai dibaca:

pop dan jumlahkan seluruh nilai yang ada pada Stack, lalu tampilkan hasilnya

- Output:
- pesan "tidak ada transaksi" jika pita karakter kosong atau hanya berisi blank, ATAU
 - pesan "Stack penuh" jika Stack penuh, ATAU
 - hasil penjumlahan seluruh elemen Stack, yang menggambarkan total setoran dikurangi total penarikan }

Use ...

KAMUS

...

ALGORITMA

...

Bagian III. ADT Queue (Notasi Algoritmik, bobot: 25%)

Diberikan definisi ADT Queue yang direpresentasikan dengan array statik menggunakan alternatif-1 sebagai berikut.

```

{ Modul ADT Queue - Alternatif I }
{ *** Deklarasi Queue yang diimplementasi dengan array *** }
{ *** HEAD dan TAIL adalah alamat elemen pertama dan terakhir *** }
{ *** Queue mampu menampung MaxEl buah elemen *** }
{ *** Konstanta *** }

constant Nil: integer = -1
constant MaxEl: integer = 100 { maksimum banyaknya elemen queue }
{ *** Definisi elemen dan address *** }

type infotype: integer
type address : integer {indeks tabel}

{ *** Definisi Type Queue *** }
type Queue: <      T: array [0..MaxEl-1] of infotype
              { array penyimpan elemen Queue }
              HEAD: address, { alamat penghapusan}
              TAIL: address, { alamat penambahan} >
{ Definisi Queue kosong: Head = Nil; TAIL = Nil.}
{ Definisi akses dengan Selektor:
  Head(Q), Tail(Q), MaxEl(Q), InfoHead(Q), InfoTail(Q)
  dengan Q adalah Queue }

{ *** Predikat Pemeriksaan Kondisi Queue *** }

function IsEmpty (Q: Queue) → boolean
{ Mengirim true jika Q kosong }

function IsFull (Q: Queue) → boolean
{ Mengirim true jika array penampung elemen Q sudah penuh yaitu mengandung MaxEl elemen }

function NBElem (Q: Queue) → integer
{ Mengirimkan banyaknya elemen queue. Mengirimkan 0 jika Q kosong. }

{ *** Konstruktor dan Destruktor *** }

procedure CreateEmpty (output Q: Queue, input Max: integer > 0)
{ I.S. Max terdefinisi }

{ F.S. Sebuah Q kosong terbentuk dan salah satu kondisi sbb: }
{     Jika alokasi berhasil, tabel memori dialokasi berukuran Max }
{     atau: jika alokasi gagal, Q kosong dg MaxEl=0 }

```

```

{ Proses: Melakukan alokasi memori dan membuat sebuah Q kosong }
procedure DeAlokasi (input/output Q: Queue)
{ Proses: Mengembalikan memori Q }
{ I.S. Q pernah dialokasi }
{ F.S. Q menjadi tidak terdefinisi lagi, MaxEl(Q) diset 0 }
{ *** Operator-Operator Dasar Queue *** }

procedure Enqueue (input/output Q: Queue, input X: infotype)
{ Proses: Menambahkan X pada Q dengan aturan FIFO }
{ I.S. Q mungkin kosong, tabel penampung elemen Q TIDAK penuh }
{ F.S. X menjadi TAIL yang baru}
procedure Dequeue (input/output Q: Queue, output X: infotype)
{ Proses: Menghapus elemen pertama pada Q dengan aturan FIFO }
{ I.S. Q tidak kosong }
{ F.S. X = nilai elemen HEAD pada I.S.,
      Jika Queue masih isi: HEAD diset tetap = 0, elemen-elemen setelah
      HEAD yang lama digeser ke "kiri", TAIL = TAIL - 1;
      Jika Queue menjadi kosong, HEAD=TAIL=Nil }

{ *** Operator Tambahan *** }

procedure Split (input Q: Queue, input ratio: real, output Q1, Q2: Queue)
{ I.S. Q tidak kosong, dan  $0 \leq X \leq 1$  }
{ F.S. Q1 dan Q2 akan berisi elemen-elemen dari Q secara bersilangan
      (elemen pertama Q ke Q1, elemen kedua Q ke Q2, elemen ketiga Q ke
      Q1, dan seterusnya), dengan NBELmt(Q1) = ratio  $\times$  NBELmt(Q).
      Jika hasilnya bukan bilangan bulat, maka dilakukan pembulatan ke
      atas.
      Contoh: NBELmt(Q)=5 dan ratio = 0,25, maka NBELmt(Q1)=0,25 $\times$ 5=1,25≈2.
      Sehingga Q1 akan memiliki 2 elemen (dibulatkan ke atas), dan Q2 akan
      memiliki 3 elemen. }

```

Dengan spesifikasi ADT Queue di atas, realisasikan prosedur-prosedur berikut (spesifikasi tidak perlu ditulis ulang). Gunakan semaksimal mungkin primitif dan selektor yang ada. Tidak diperkenankan untuk membuat fungsi/prosedur baru.

1. Prosedur Dequeue.
2. Prosedur Split.

Bagian IV. ADT Set dan Hash (Notasi Algoritmik, bobot: 25%)

Cpl. Gionno (dibaca: Kopral Jono) ingin berhenti dari tentara untuk membuat suatu sistem *e-commerce*. Untuk itu ia membutuhkan ADT Bag yang merepresentasikan keranjang belanja. ADT Bag mirip dengan ADT Set, bedanya untuk setiap item yang disimpan dalam sebuah Bag terdapat pencatat jumlah. Satu unit item dapat ditambahkan dan dikurangi dari Bag dengan operasi Add dan Remove. Sebagai ilustrasi, untuk sebuah Bag B yang semula kosong, urutan pemanggilan operasi Add(B,telur); Add(B,susu); Add(B,susu); Add(B,telur); Remove(B,susu); Add(B,roti); Remove(B,roti); menghasilkan isi B: [telur⇒2, susu⇒1]. Selain itu terdapat fungsi Get yang mengembalikan jumlah kemunculan sebuah item dalam Bag. Pada ilustrasi tadi, Get(B,telur) mengembalikan 2 sedangkan Get(B,roti) mengembalikan 0.

Agar penggunaan memori efisien, Gionno tidak mau menyimpan item dalam Bag sebagai **string**, melainkan sebagai **integer** yang merepresentasikan ID barang. Ia menggunakan ADT Map untuk menyimpan pemetaan ID barang ke *tuple Item* berisi nama dan harga satuan, misal [54⇒⟨"telur",3500⟩, 72⇒⟨"susu",12000⟩, 63⇒⟨"roti",16000⟩]. Nama dan harga sebuah **item** dapat diakses dengan **item.name** dan **item.price**. Bantulah Gionno membangun

sistem e-commerce ini dengan:

1. Membuat **definisi ADT Bag**. Gunakan *array* kontigu untuk menyimpan elemen-elemen Bag dan NbElmt untuk mencatat jumlah jenis item yang ada di dalam Bag. Tuliskan prototipe untuk konstruktor dan operasi-operasi Add, Remove, dan Get lengkap dengan spesifikasinya (prekondisi/I.S.,F.S.)
2. Membuat **implementasi operasi Add dan Remove**. Asumsikan item disimpan dalam *array* sesuai urutan penambahan item ke Bag. (Tidak sorted, tidak pakai *hash table*.)
3. Membuat **prosedur CetakInvoice(*input B:Bag, input M:Map*)** yang mencetak jumlah, nama, dan harga setiap item di dalam Bag B, serta total harga yang harus dibayar. Untuk memudahkan iterasi, Gionno sudah membuat fungsi ItemIDs(B: Bag) → array of integer yang mengembalikan *array* berisi ID semua item yang ada di B. ADT Map dengan key bertipe integer dan value bertipe Item juga sudah dibuat oleh Gionno dengan sebagian spesifikasi sebagai berikut—Anda tinggal menggunakan:

```

type Item: < name: string, price: integer >
type keytype: integer
type infotype: Item
type Map: ... { terdefinisi }

 $\{ \text{** aksesori **} \}$ 
function Get(M: Map, k: keytype) → infotype
{ Mengembalikan tuple Item yang berkorespondensi dengan key k dalam M
  atau Nil jika key k tidak ada dalam M }
  
```

Contoh hasil pemanggilan CetakInvoice pada Bag B [54⇒2,72⇒1] dan Map M [54⇒ < "telur",3500> , 72⇒ <"susu",12000> , 63⇒ <"roti",16000>]:

```

2 telur 7000
1 susu 12000
TOTAL 19000
  
```

Tidak memperkenankan membuat/mengasumsikan adanya fungsi/prosedur selain yang disebutkan pada soal.

UTS Algoritma Struktur Data 2021/2022

ADT MATRIX

1. ADT Matrix didefinisikan sebagai berikut

```

constant UNDEF: integer = -9999
{ Ukuran maksimum baris dan kolom: }
constant ROW_CAP: integer = 100
constant COL_CAP: integer = 100
{ Definisi Type Matrix dengan indeks integer: }
type ElType: integer
type Matrix:
    < mem: matrix[0..COL_CAP-1,0..ROW_CAP-1] of ElType,
      rowEff: integer, { ukuran baris yg terdefinisi }
      colEff: integer { ukuran kolom yg terdefinisi } >
{ Memori matriks yang dipakai selalu di "ujung kiri atas" }

{ *** Konstruktor membentuk Matrix *** }
procedure CreateMatrix(output m: Matrix, input nRows, nCols: integer)
{ Membentuk sebuah Matrix "kosong" berukuran nRows x nCols di "ujung kiri" memori }
{ I.S. nRows dan nCols adalah valid untuk memori matriks yang dibuat }
{ F.S. sebuah matriks m sesuai dengan def di atas terbentuk }

{ *** Untuk sebuah matriks m yang terdefinisi: *** }
function getRowEff(m: Matrix) → integer
{ Mengirimkan jumlah baris Matriks m }
function getCollEff(m: Matrix) → integer
{ Mengirimkan jumlah kolom Matriks m }
function isMatrixIdxValid(i, j: integer) → boolean
{ Mengirimkan true jika i, j adalah indeks yang valid sesuai kapasitas maksimum sebuah Matrix, ROW_CAP dan COL_CAP }
function isIdxEff (m: Matrix, i, j: integer) → boolean
{ Mengirimkan true jika i, j adalah indeks efektif bagi m }
function getElmt (m: Matrix, i, j: integer) → ElType
{ Mengirimkan elemen m dg nomor baris i dan nomor kolom j }

```

- a. Tulislah implementasi operasi isMatrixIdxValid(i,j) dan isIdxEff(m,i,j).
- b. Tulislah implementasi (dalam Notasi Algoritmik) untuk fungsi vlookup(val, m, col) dengan definisi di bawah ini. Anda boleh menggunakan primitif yang telah dideklarasikan di atas dan tidak boleh membuat/mengasumsikan fungsi tambahan.

```

function vlookup(val: ElType, m: Matrix, col: integer) → ElType
{ Mengembalikan nilai pada m di kolom ke-col dan baris yang kolom pertamanya bernilai val.
  Prekondisi: val, m, dan col terdefinisi.
  Proses:
    - mendapatkan di baris keberapa nilai val muncul di kolom pertama (indeks ke-0) Matrix m.
    - mengambil nilai di kolom ke-col pada baris yang didapatkan tadi.
    - jika val tidak ada di kolom pertama m, atau col di luar indeks yang valid untuk m, kembalikan UNDEF
    - jika val muncul beberapa kali di kolom pertama m, ambil baris dengan indeks terkecil }

```

Ilustrasi penggunaan fungsi: Untuk Matrix

$$m = \begin{bmatrix} 7 & 86 & -15 \\ 4 & 103 & -19 \\ 11 & 44 & -1 \end{bmatrix}$$

vlookup(4, m, 2) menghasilkan -19 dan vlookup(-1, m, 1) menghasilkan UNDEF.

MESIN KATA

2. Ali Baba mencoba masuk ke Gedung Labtek V untuk memberikan kuliah umum tetapi pintu

gedung dijaga oleh satpam. Agar dapat masuk, dia harus mengambil pita karakter berisi kata-kata yang diakhiri dengan titik, untuk diserahkan ke satpam. Ali Baba hanya bisa masuk jika dia mengambil pita karakter yang berisi kata-kata dengan huruf awal kata yang membentuk “Opensesame”. Bantu Ali Baba masuk dengan mesin kata untuk memastikan pita karakter yang diambil Ali Baba dapat membuat dia masuk ke gedung.

Contoh pita karakter yang sesuai:

Orang pintar enggan naik sampan entah senang atau malah emosi.

O p e n s e s a m e

Orang pintar enggan naik sampan entah senang atau malah emosi lalu pergi.

O p e n s e s a m e

Contoh pita karakter yang tidak sesuai:

Ongkos pulang esok nanti sama dengan hari ini tetap murah.

O p e n s d → “bukan pita yang tepat”

Buatlah program yang membaca pita karakter dengan mesin kata. Anda dapat memilih mesin kata (versi 1, 2, atau 3), dan mesin kata boleh langsung digunakan. Program akan menampilkan:

- pesan “Pita karakter kosong” jika pita kosong,
- pesan “Bukan pita yang tepat.” jika huruf-huruf awal dari kata-kata dalam pita tidak membentuk ‘Opensesame’
- pesan “Open sesame!” jika huruf-huruf awal dari kata-kata dalam pita membentuk ‘Opensesame’

ADT QUEUE

3. Diberikan definisi ADT Queue yang direpresentasikan dengan array statik menggunakan alternatif-2 sebagai berikut.

```

{ Modul ADT Queue - Alternatif II }
{ *** Deklarasi Queue yang diimplementasi dengan array *** }
{ *** HEAD dan TAIL adalah elemen pertama dan terakhir *** }
{ *** Queue mampu menampung elemen maksimal sejumlah CAPACITY *** }
{ *** Konstanta *** }
constant IDX_UNDEF : integer = -1
constant CAPACITY : integer = 100

{ *** Definisi elemen dan address *** }
type ElType : integer

{ *** Definisi Type Queue *** }
type Queue : < buffer : array [0..CAPACITY-1] of ElType
            idxHead : integer,
            idxTail : integer >
{ Definisi Queue kosong: idxHead = idxTail = IDX_UNDEF }

{ Definisi akses dengan Selektor :
  HEAD(Q), TAIL(Q), IDX_HEAD(Q), IDX_TAIL(Q)
  dengan Q adalah Queue}

{ *** Operasi: pemeriksaan status Queue *** }
function isEmpty (Q : Queue) → boolean
{ Mengirim true jika Q kosong}
function isFull (Q : Queue) → boolean
{ Mengirim true jika array penampung elemen Q sudah penuh yaitu mengandung MaxEl elemen}
function length (Q : Queue) → integer
{ Mengirimkan banyaknya elemen queue. Mengirimkan 0 jika Q kosong. }

{ *** Konstruktor *** }
procedure CreateQueue (output Q : Queue)
{ I.S. Sembarang }
{ F.S. Membuat sebuah Queue q yang kosong berkapasitas CAPACITY
  jadi indeksnya antara 0..CAPACITY-1
  Ciri Queue kosong: idxHead dan idxTail bernilai IDX_UNDEF }

{ *** Primitif Add/Delete *** }
procedure enqueue (input/output Q : Queue, input val : ElType)
{ Proses : Menambahkan val pada Q dengan aturan FIFO
  I.S. Q mungkin kosong, tabel penampung elemen Q TIDAK penuh
  F.S. queue bertambah elemen val sebagai tail yang baru, TAIL bergeser ke kanan
  Jika IDX_TAIL(queue)=CAPACITY-1, maka geser isi tabel, shg IDX_HEAD(queue)=0 }

procedure dequeue (input/output Q : Queue, output X : ElType)
{ Menghapus head dari Queue q.
  I.S. queue tidak kosong
  F.S. val berisi nilai head yang lama.
  Jika queue tidak menjadi kosong,
  queue.idxHead berpindah ke elemen berikutnya pada queue.
  Jika queue menjadi kosong,
  queue.idxHead dan queue.idxTail menjadi bernilai IDX_UNDEF. }

```

Dengan spesifikasi ADT Queue di atas, realisasikan prosedur-prosedur berikut (spesifikasi tidak perlu ditulis ulang). Gunakan semaksimal mungkin primitif dan selektor yang ada. Tidak diperkenankan untuk membuat fungsi/prosedur baru.

- Prosedur enqueue sesuai definisi dan spesifikasi pada ADT
- Prosedur SplitToThree sesuai definisi dan spesifikasi berikut:

```

procedure SplitToThree (input Q : Queue, input ratio : real, output Q1, Q2, Q3: Queue)
{I.S. Q tidak kosong, dan 0 ≤ ratio ≤ 1}
{F.S. Q1, Q2 dan Q3 merupakan pecahan dari antrian Q yang berisi elemen-elemen dari Q secara bergantian (elemen pertama Q ke Q1, elemen kedua Q ke Q2, elemen ketiga Q ke Q3, elemen keempat Q ke Q1, dan seterusnya). length(Q1) = ratio × length(Q) yang dibulatkan ke atas, lalu sisa yang berupa selisih antara length(Q) dan length(Q1) dibagi mendekati sama rata ke Q2 dan Q3. Jika sisa tersebut ganjil, maka length(Q2) = 1 + length(Q3). Contoh: length(Q)=5 dan ratio = 0,25, maka length(Q1)=0,25×5=1,25≈2 sehingga Q1 akan memiliki 2 elemen (dibulatkan ke atas). Lalu sisa selisihnya adalah 5-2=3, sehingga Q2 akan memiliki 2 elemen dan Q3 akan memiliki 1 elemen}

```

ADT SET/MAP

- Multiset merupakan bentuk lebih umum dari Set di mana setiap elemen dapat muncul berulang

kali di dalamnya. Sebagai contoh, $\{1,2,3\}$ dan $\{2,1,3\}$ merupakan Multiset yang sama, namun $\{1,1,2,3\}$ merupakan Multiset yang berbeda dari kedua Multiset sebelumnya. Jumlah kemunculan sebuah elemen dalam Multiset disebut sebagai multiplicity (dilambangkan v). Pada $\{1,2,3\}$, multiplicity dari nilai 1 adalah 1, sedangkan pada $\{1,1,2,3\}$, multiplicity dari nilai 1 adalah 2. Keanggotaan Multiset memiliki definisi yang sedikit berbeda dari Set. Pada Multiset, $\text{isIn}(a,x,s)$ bernilai true jika dan hanya jika elemen a muncul di Multiset s sekurang-kurangnya x kali.

Contoh: jika $s = \{1,2,3,1,1\}$, $\text{isIn}(1, 2, s)$ dan $\text{isIn}(1, 3, s)$ keduanya menghasilkan true.

Selain itu, $\text{union}(s, t)$ menghasilkan Multiset di mana multiplicity dari setiap elemen adalah $\max\{vs, vt\}$. Contoh: jika $s = \{1,1,2\}$ dan $t = \{2,2,3\}$, $\text{union}(s, t) = \{1,1,2,2,3\}$. (Perhatikan multiplicity dari nilai 2.) Di memori komputer, dari pada menyimpan sebuah nilai berulang kali, kita dapat menyusun sebuah struktur data yang menyimpan pasangan nilai dan multiplicity saja. Sebagai contoh, $\{1,2,3,1,1\}$ dapat kita simpan sebagai $\langle\langle1,3\rangle,\langle2,1\rangle,\langle3,1\rangle\rangle$. Jika diperhatikan dengan baik, struktur ini sebenarnya mirip dengan struktur untuk ADT Map dengan key adalah elemen Multiset dan value adalah multiplicity-nya.

- a. Buatlah definisi ADT Multiset yang merupakan modifikasi ADT Map (mencakup definisi tipe bentukan dan spesifikasi header operasi primitif yang diperlukan). Sebagai petunjuk, primitif-primitif yang dimaksud antara lain seperti pada ADT Map, yaitu $\text{set}(\text{key}, \text{value}, \text{map})$: mengubah nilai yang ditunjuk oleh key pada map menjadi value, $\text{unset}(\text{key}, \text{map})$: membuang key dari map, dan $\text{find}(\text{key}, \text{map})$: mengembalikan value yang ditunjuk oleh key pada map.
- b. Buatlah implementasi $\text{isIn}()$ dan $\text{union}()$. Untuk memudahkan traversal elemen-elemen Multiset, Anda dapat menganggap elemen disimpan secara rata kiri di dalam array. Selain itu, fungsi untuk mencari nilai maksimum antara dua integer $\max(a, b)$ juga dapat langsung digunakan.

UAS ALGORITMA DAN STRUKTUR DATA

UAS SEMESTER I – 2011/2012

8 Desember 2011

Waktu: 120 menit

Soal 1. ADT TRIANGLE, Notasi Algoritmik [Bobot: 25%]

Diberikan definisi dan spesifikasi ADT POINT sebagai berikut:

```
(*** ADT POINT ***)
(** Definisi Type Point **)
type Point : <x : real, y:
real>
(<x, y> adalah sebuah point dengan x adalah absis, y adalah ordinat)
(** Definisi dan spesifikasi konstruktor
***) function MakePoint (a, b : real) ⌈ Point
(membentuk sebuah point dari a dan b dengan a sebagai absis dan b sebagai ordinat)
(** Definisi dan spesifikasi selektor
***) function Absis (P: Point) ⌈ real
(memberikan absis point P)
function Ordinat (P: Point) ⌈ real
(memberikan ordinat point P)
function Jarak (P1, P2: Point) ⌈ real
(mengembalikan jarak antara titik P1 dan P2 dengan menggunakan rumus jarak)
```

Dengan memanfaatkan ADT POINT di atas, buat ADT TRIANGLE yang meliputi hal-hal sbb.

- Definisikan tipe bentukan TRIANGLE yang merepresentasikan sebuah segitiga, yaitu terdiri atas 3 buah POINT. Buatlah definisi, spesifikasi, dan realisasi fungsi konstruktor dan selektornya (*getter*).
- Buatlah definisi, spesifikasi, dan realisasi sebuah fungsi IsValidTriangle yang menerima 3 buah POINT dan melemparkan true jika ketiga POINT tersebut dapat membentuk sebuah TRIANGLE. Tiga buah POINT P1, P2, P3 dinyatakan tidak dapat membentuk segitiga jika ketiga POINT tsb dapat membentuk 2 garis dengan gradien (kemiringan) yang sama. Jika P1=<x1,y1> dan P2=<x2,y2>, maka gradiennya adalah $m=(y2-y1)/(x2-x1)$. Perhatikan bahwa pada kasus $x2-x1=0$, perlu dilakukan penanganan kasus khusus.
- Buatlah definisi, spesifikasi, dan realisasi sebuah fungsi JenisSegitiga yang menerima sebuah TRIANGLE yang valid dan mengembalikan nilai yang menggambarkan jenis segitiga sesuai dengan tabel berikut:

Segitiga siku-siku?	Segitiga sama kaki?	Keluaran fungsi JenisSegitiga
Ya	Ya	1
Ya	Tidak	2
Tidak	Ya	3
Tidak	Tidak	4

Perhatikan bahwa

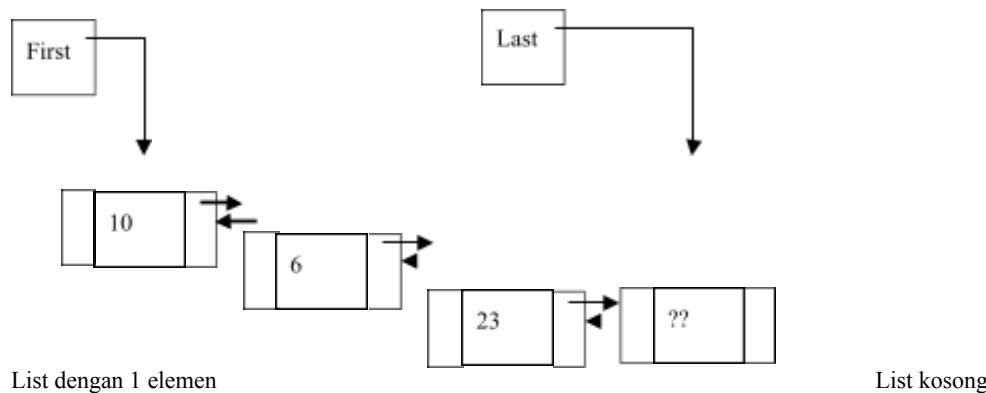
- Sebuah segitiga dikatakan sebagai siku-siku, jika segitiga tsb mempunyai sisi-sisi a, b, c, dan c adalah sisi terpanjang, maka $c^2=a^2+b^2$.
- Sebuah segitiga dikatakan segitiga sama kaki, jika 2 atau lebih sisinya memiliki panjang yang sama.

Buatlah definisi, spesifikasi, dan realisasi fungsi/prosedur antara jika diperlukan.

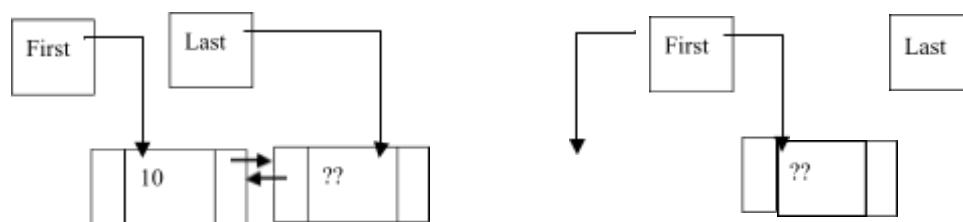
Soal 2. List Double Pointer dengan elemen dummy sebagai elemen terakhir, Notasi Algoritmik [Bobot: 25%]

Diberikan definisi list berkait dikenali First(L) alamat elemen pertama list, Last(L) alamat elemen terakhir. Elemen terakhir adalah elemen fiktif/dummy. Setiap elemen list terdiri dari informasi bertipe infotype, alamat elemen sebelumnya bertipe address, dan alamat elemen berikutnya bertipe address.

List dengan banyak elemen



List dengan 1 elemen



List kosong

Diberikan potongan ADT list sebagai berikut. Realisasikan function SearchX, procedure InsertLast, dan procedure DeleteLast (spesifikasi tidak perlu ditulis ulang). Tidak diperbolehkan membuat prosedur/fungsi/type/konstanta baru.

```

Kamus
{List direpresentasi dengan pointer}
constant Nil : ... {address tidak
terdefinisi} type_infotype : integer
type_ElmList : <Info: infotype, Prev, Next:
address> type_address : pointer to ElmList
type_List : <First: address, Last: address>
{Jika dideklarasikan L: List dan P: address
maka: L.First dapat diacu sebagai First(L);
L.Last dapat diacu sebagai Last(L);
P^.Prev dapat diacu sebagai
Prev(P); P^.Next dapat diacu
sebagai Next(P); P^.Info dapat
diacu sebagai Info(P)}
{Definisi list:
(List kosong: First(L) = Last(L) - dummy@)
{Definisi primitif yang boleh digunakan tanpa perlu
direalisasikan} function IsEmpty (L: List) @boolean
{mengirim true jika list kosong, false jika list tidak
kosong} function Alokasi (X: infotype) @address
{menghasilkan address elemen list hasil alokasi dengan Info=X, Next=Nil}
{bila alokasi gagal menghasilkan Nil}
procedure Dealokasi (input/output P:
address)
{I.S. : P terdefinisi}
{F.S. : P didealokasi (dikembalikan ke sistem)}

```

```
{Realisasikan fungsi dan prosedur berikut, tanpa membuat fungsi/prosedur/type baru}

procedure InsertLast (input/output L: List, input P: address)
{I.S. Sembarang, P sudah dialokasi)
{F.S. Pditambahkan sebagai elemen terakhir yang baru}

procedure DeleteLast (input/output L: List, output P: address)
{I.S. List tidak kosong)
{F.S. P adalah alamat elemen list (terakhir) yang dihapus)
{Elemen list berkurang satu (mungkin menjadi kosong)}

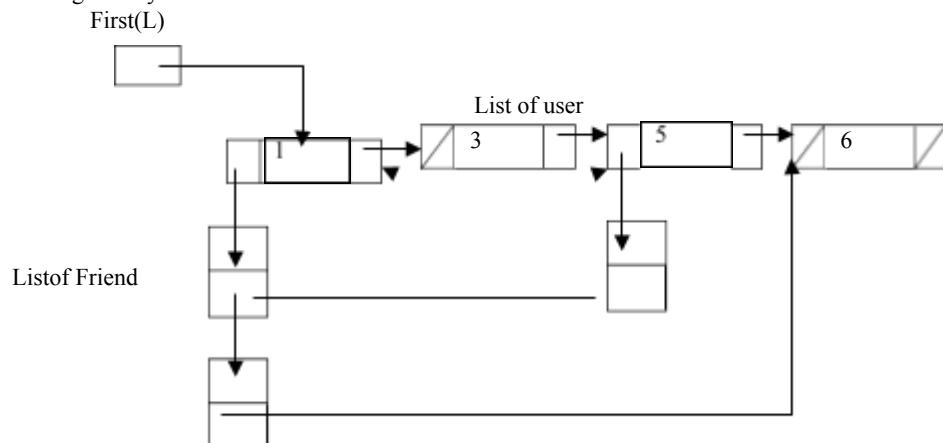
function SearchX(L: List, X: infotype) address
{mencari apakah ada elemen list dengan info(P)=X)
{Jika ada, mengirimkan address elemen tersebut.)
{Jika tidak ada, mengirimkan Nil}
```

Soal 3. List Pertemanan , Notasi Algoritmik [Bobot: 25%]

Di suatu program jejaring sosial, pertemanan antara satu user dengan user lain didefinisikan dalam suatu ADT list linier. Setiap elemen list menyimpan info mengenai id user (dinyatakan sebagai integer positif) dan mengenai list semua teman/friend dari user ybs. Dalam list ini, id user bersifat unik dan disusun terurut membesar. List dikenali dengan elemen pertamanya. Berikut adalah ilustrasi list pertemanan ini:

List kosong  First(L)=Nil

List dengan banyak elemen:



Keterangan untuk

list dengan banyak elemen:

- User 1 berteman dengan user 5 dan 6.
- User 5 berteman dengan user 1.
- User 3 tidak berteman dengan siapapun.
- Pertemanan user 1 dan 5 bersifat reciprocal (artinya user 1 berteman dengan 5 dan demikian sebaliknya). Pertemanan user 1 dan 6 tidak bersifat reciprocal karena hanya user 1 yang berteman dengan user 6, tapi tidak sebaliknya.

Berikut adalah deklarasi ADT list linier bersangkutan:

```
{Deklarasi list linier dengan representasi fisik pointer :}
constant Nil: ... {address tidak terdefinisi, baik untuk AdrUser maupun AdrFriend}
```

```

type InfoType: integer > 0
type ElmtListUser: <IdUser : InfoType,           {id user}
                  NextUser : AdrUser,        {next user}
                  FirstFriend :          {first dari list of
                  AdrFriend,             friend}>
type AdrUser: pointer to ElmtListUser, (address user yg menjadi
friend) NextFriend: AdrFriend, (next friend)>
type AdrFriend: pointer to ElmtListFriend
type List: < First : AdrUser>

{SELEKTOR: Gunakan selektor-selektor di bawah ini:}
{Jika L adalah List, PU adalah AdrUser, PF adalah AdrFriend:}
{    First(L) untuk mengakses L.First}
{    IdUser(PU) untuk mengakses komponen PU .IdUser}
{    NextUser(PU) untuk mengakses komponen PU .NextUser}
{    FirstFriend(PU) untuk mengakses komponen PU .FirstFriend}
{    UserFriend(PU) untuk mengakses komponen PU .UserFriend}
{    NextFriend(PU) untuk mengakses komponen PU .NextFriend}
{List kosong: First(L) = Nil}
{Jika Last adalah elemen terakhir, maka Next>Last)=Nil}
{Primitif memory management}
function AlokUser (x: InfoType) : AdrUser
{Mengalokasi elemen untuk user, misalnya P. Jika alokasi berhasil, IdUser(P)=x,
NextUser(P)
= Nil, FirstFriend(P) = Nil. Jika alokasi gagal, P=Nil.}

```

Realisasikan prosedur InserUser dan fungsi IsReciprocal di bawah ini (spesifikasi tidak perlu ditulis ulang). Jika perlu membuat fungsi/prosedur baru, buatlah definisi, spesifikasi, dan realasinya. Semua primitif ADT di atas dapat digunakan tanpa perlu direalisasikan.

```

procedure InsertUser (input/output L: List, input x: InfoType)
{I.S. : L terdefinisi, mungkin kosong. X terdefinisi.}
{F.S. : Jika elemen dengan id user bernilai x belum ada di L dan alokasi}
{x berhasil, maka elemen dengan id user bernilai x ditambahkan di} { list L dengan L
tetap terurut membesar. Jika alokasi x gagal, L} { tetap. Jika sudah ada elemen
dengan id user bernilai x di L, L} { tetap.}
{Proses: Searching elemen dengan id user bernilai x di list L yang terurut membesar}

Function IsReciprocal (L: List; x,y: InfoType) : boolean
{Menghasilkan true jika pada list L terdapat elemen dengan id user x dan y yang}
{memiliki pertemanan yang reciprocal, yaitu y muncul di list of friend dari x dan}
{x juga muncul di list of friend dari y.}
{Menghasilkan false jika kedua elemen tidak reciprocal.}
{Prekondisi: Pada list L,terdapat elemen dengan id user bernilai x dan y.}
{x dan y pasti berbeda.}

```

Soal 4. Pohon Biner, Notasi Algoritmik [Bobot: 25%]

Diberikan definisi ADT dari Pohon Biner yang digunakan untuk merepresentasikan sebuah ekspresi aritmatika biner sebagai berikut:

```

type infotype : string
type Node : <Info : infotype,
            Left : BinTree,
            Right : BinTree>

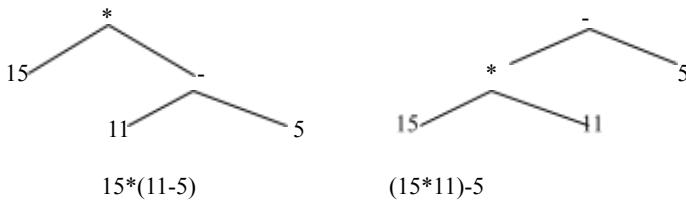
```

```

Right : BinTree>
type address : pointer to
Node type BinTree : address
(Selektor: Jika P adalah BinTree,
Akar(P) untuk mengakses bagian
P.Info Left(P) untuk mengakses
bagian P.Left
Right(P) untuk mengakses bagian P.Right}
(Primitif)
function IsTreeEmpty(P: BinTree) Boolean
(Mengirimkan true jika P adalah pohon biner yang
kosong) function IsOneElmt(P: BinTree) Boolean
(Mengirimkan true jika P tidak kosong dan hanya terdiri atas 1
elemen) function IsUnerLeft(P: BinTree) Boolean
(Mengirimkan true jika pohon biner tidak kosong P adalah pohon unerleft: hanya mempunyai
subpohon kiri)
function IsUnerRight(P: BinTree) Boolean
(Mengirimkan true jika pohon biner tidak kosong P adalah pohon unerleft: hanya mempunyai
subpohon kanan)
function IsBiner(P: BinTree) Boolean
(Mengirimkan true jika pohon biner tidak kosong P adalah pohon unerleft: hanya mempunyai
subpohon kiri dan subpohon kanan)

```

Berikut adalah 2 contoh instansiasi dari Pohon Biner tersebut:



Info dari sebuah Node adalah sebuah string yang memuat angka atau operator (contoh: “+”, “-”, “*”, dan “/”). Ekspresi aritmatika yang valid dan tidak kosong memiliki satu operator dan dua buah operan. Operan dapat berupa angka atau ekspresi aritmatika valid yang lain.

Realisasikan prosedur SmashedTree sesuai spesifikasi di bawah ini (spesifikasi tidak perlu ditulis ulang). Jika membutuhkan fungsi/prosedur antara, buatlah definisi, spesifikasi, dan realisasinya. Semua primitif pada ADT di atas dapat digunakan tanpa harus dibuat realisasinya.

```
procedure SmashedTree (input P: BinTree, output x: string)
{I.S. : P terdefinisi; tidak mungkin kosong dan merepresentasikan suatu
{     ekspresi aritmatika yang valid.}
{F.S. : Menghasilkan x dimana sebuah string yang merepresentasikan pohon P
{     secara linier.}
{Proses : Menghasilkan struktur flat berdasarkan struktur bertingkat dari
{     sebuah pohon. Pada kasus kedua contoh pohon di atas, prosedur
{     akan menghasilkan string: "15*11-5"}}
```

Catatan: Untuk melakukan konkatenasi dua buah string, gunakan operator &. Contoh: "aaa" & "bbb" = "aaabbba"

Diberikan sebuah prosedur `PrintBFS` yang melakukan pencetakan ke layar semua elemen dari pohon biner secara *breadth-first*. Pada kasus kedua pohon di atas, prosedur akan mencetak: * 15 - 11 5 dan - * 5 15 11 ke layar.

```
procedure PrintFBS (input P: BinTree)
{I.S. : P terdefinisi; tidak mungkin kosong dan merepresentasikan suatu
{      ekspresi aritmatika yang valid.}
{F.S. : Seluruh elemen P tercetak ke layar secara Breadth-First: terurut
{      mulai dari level teratas sampai terbawah, dan untuk setiap level
{          dari node terkiri ke node terkanan.}
```

```
(Proses : Mencetak semua
elemen.) KAMUS LOKAL
Q : Queue
Pt : BinTree
ALGORITMA
Add(Q,P)
iterate:
    Del(Q,Pt)
    output (Akar(Pt))
stop: (IsEmpty(Q) and
        IsTreeEmpty(Pt)) if (not
        IsUnerLeft(Pt)) then
            Add(Q,Right(Pt)) then
            If (not IsUnerRight(Pt))
            then Add(Q,Left(Pt))
        IsEmpty(Q) and IsTreeEmpty(Pt) }
```

Digunakan struktur data Queue sebagai berikut:

```
type infoQ: BinTree
type ElmtQ: <Info: infoQ,
            Next: AddressQ>
type AddressQ: pointer to ElmtQ
type Queue: <Head: AddressQ,
            Tail: AddressQ>
{Selektor: Jika Q adalah Queue,
 Head(Q) untuk mengakses bagian Q.Head
 Tail(Q) untuk mengakses bagian Q.Tail
 InfoHead(Q) untuk mengakses bagian (Q.Head).Info
 InfoTail(Q) untuk mengakses bagian (Q.Tail).Info}
{Primitif}
function IsEmpty(Q: Queue) boolean
{Tes terhadap Q: true jika Q kosong, false jika Q tidak kosong}
procedure CreateEmpty(output Q: Queue)
{Membuat sebuah antrian kosong}
{I.S. : sembarang}
{F.S. : terbentuk sebuah Queue Q kosong}
procedure Add(input/output Q: Queue, input X: infoQ)
{Menambah sebuah elemen setelah elemen ekor/tail QUEUE}
{I.S. : Q mungkin kosong}
{F.S. : X menjadi tail Q}
procedure Del(input/output Q: Queue, output X: infoQ)
{Menghapus kepala QUEUE, mungkin Q menjadi kosong}
{I.S. : Q tidak mungkin kosong}
{F.S. : X berisi nilai elemen head pada queue awal; head yang baru adalah
elemen kedua pada queue awal atau Q menjadi kosong}
```

Melihat kedua contoh pohon tersebut di atas, bagaimana hasil pencetakan di layar dari prosedur tersebut? Apakah hasil tersebut telah sesuai dengan spesifikasi dari prosedur tersebut? Jika TIDAK, tuliskan realisasi prosedur yang tepat menurut Anda.

UJIAN AKHIR SEMESTER

13 Desember 2010

120 menit

Soal 1. Priority Queue dengan Representasi Tabel Secara Sirkuler, Notasi Algoritmik [Bobot 15%]

Priority queue adalah antrian dimana urutan dalam antrian ditentukan oleh prioritas elemennya. HEAD akan menunjuk elemen dengan prioritas tertinggi dan TAIL akan menunjuk elemen dengan prioritas terendah. Prioritas dinyatakan sebagai bilangan bulat 1..n, dengan 1 adalah prioritas tertinggi dan n adalah prioritas terendah. Elemen prioritas queue terdiri atas prio yang menyatakan prioritas elemen dan info yang menyatakan informasi yang dipegang oleh elemen.

prio	info
------	------

Elemen priority queue:

Contoh priority queue dengan 5 elemen:

				1	12	3	10	3	7	4	18	6	33				
1	2	3	4	5	6	7	8	9	10								

HEAD: 3 TAIL: 7

Priority queue ini direpresentasikan dengan tabel secara sirkuler. Artinya pada saat penambahan elemen, jika elemen terakhir sudah terisi dan tabel belum penuh, maka penambahan dilakukan dari awal tabel. Namun demikian, penambahan ini juga harus mempertimbangkan urutan prioritas elemen.

Diberikan potongan ADT queue sebagai berikut. Buatlah realisasi prosedur Add sesuai spesifikasi yang diberikan (spesifikasi tidak perlu ditulis ulang). Tidak diperbolehkan membuat prosedur/fungsi/type/konstanta baru.

```

{Deklarasi priority queue yang diimplementasi dgn tabel kontigu statik}
{HEAD dan TAIL adalah alamat elemen pertama dan terakhir}
{Tabel yang dipakai untuk menyimpan nilai elemen queue: T[1]..T[MaxEl]}
{MaxEl adalah ukuran maksimum tabel penampung elemen queue}
constant MaxEl : integer = 10
constant Nil : integer = 0
constant N : integer = 10

{Definisi queue kosong : HEAD=Nil dan TAIL=Nil}
type infotype : <info:integer, prio : integer[1..N]>
type address : integer[0..MaxEl] {indeks elemen queue}
type PrQueue : <T : array[1..MaxEl] of infotype,
               HEAD : address,
               TAIL : address>
{Definisi primitif, boleh digunakan tanpa perlu realisasi}
function IsEmpty (PQ: PrQueue) : boolean
{mengirim true jika PQ kosong, false jika tidak kosong}
function IsFull (PQ: PrQueue) : boolean
{mengirim true jika PQ penuh yaitu mengandung MaxEl elemen, false jika tidak penuh}
procedure CreateEmpty (output PQ:PrQueue)
{I.S.: Sembarang}
{F.S.: sebuah queue kosong terbentuk dengan kondisi sbb: HEAD=Nil; TAIL=Nil}

{Realisasikan prosedur Add sesuai spesifikasi yang diberikan}
procedure Add (input/output PQ: PrQueue, input X: infotype)
{menambahkan X pada Q dengan aturan priority queue}

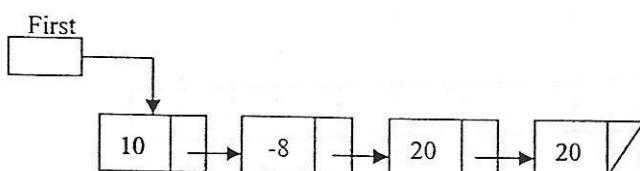
```

{I.S.: PQ mungkin kosong, tapi tidak penuh}
 {F.S.: X disisipkan pada priority queue dengan menggeser elemen tabel sehingga urutan elemen queue tetap terurut berdasarkan priority-nya. Jika priority X sama dengan priority elemen queue, penyisipan berdasarkan FIFO.
Setelah X ditambahkan, penunjuk TAIL maju dan sirkuler}}

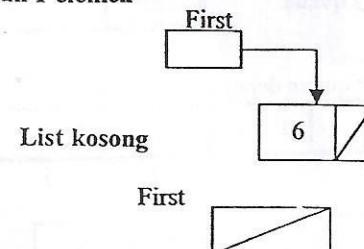
Soal 2. List Linier Berkait, Representasi dengan Pointer, Notasi Algoritmik [Bobot 20%]

Diberikan definisi list berkait dengan First(L) alamat elemen pertama list. Setiap elemen list terdiri dari informasi bertipe infotype dan alamat elemen berikutnya bertipe address.

List dengan banyak elemen



List dengan 1 elemen



List kosong

First

Diberikan potongan kamus ADT list linier sebagai berikut. Realisasikan function SearchMax dan procedure DelMax (spesifikasi tidak perlu ditulis ulang). Tidak diperbolehkan membuat prosedur/fungsi/type/konstanta baru.

KAMUS

```

{List direpresentasikan dengan pointer}
type infotype : integer
type ElmtList : <Info:infotype, Next:address>
type address : pointer to ElmtList
type List : <First:address>
{Jika dideklarasikan L>List dan P:address maka:
L.First dapat diacu sebagai First(L)
P_.Next dapat diacu sebagai Next(P)
P_.Info dapat diacu sebagai Info(P) }

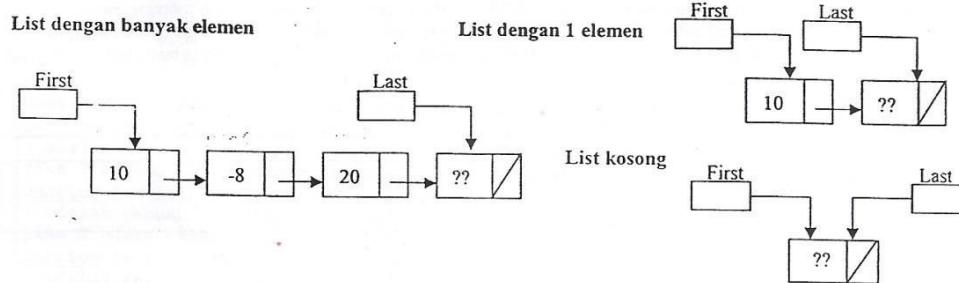
{Definisi list:}
{List kosong: First(L)=Nil}
{elemen terakhir List: jika addressnya Last, maka Next(Last)=Nil}
{Definisi primitif, boleh digunakan tanpa perlu realisasi}
function IsEmpty (L: List) --> boolean
{mengirim true jika L kosong, false jika tidak kosong}
procedure Dealokasi (input/output P: address)
{I.S.: P terdefinisi}
{F.S.: P didealokasi (dikembalikan ke sistem) }

{Realisasikan fungsi dan prosedur berikut}
function SearchMax (L: List) infotype
{Menghasilkan nilai maksimum elemen L, prekondisi: L tidak kosong}
procedure DelMax (input/output L: List)
{I.S.: L tidak kosong}
{F.S.: Semua elemen list yang bernilai maksimum dihapus dari list L dan
didealokasi.}

```

Soal 3. List dengan elemen dummy sebagai elemen terakhir, Notasi Algoritmik [Bobot 35%]

Diberikan definisi list berkait dengan First(L) alamat elemen pertama list, Last(L) alamat elemen terakhir. Elemen terakhir adalah elemen fiktif/dummy. Setiap elemen list terdiri dari informasi bertipe infotype dan alamat elemen berikutnya bertipe address.



Diberikan potongan kamus ADT list sebagai berikut. Realisasikan function SearchX, procedure InsertLast, dan procedure DeleteLast (spesifikasi tidak perlu ditulis ulang). Tidak diperbolehkan membuat prosedur/fungsi/type/konstanta baru.

```

KAMUS
{List direpresentasikan dengan pointer}
type infotype : integer
type ElmtList : <Info:infotype, Next:address>
type address : pointer to ElmtList
type List : <First:address, Last: address>
{Jika dideklarasikan L>List dan P:address maka:
L.First dapat diacu sebagai First(L)
L.Last dapat diacu sebagai Last(L)
P_.Next dapat diacu sebagai Next(P)
P_.Info dapat diacu sebagai Info(P) }

{Definisi list:}
{List kosong: First(L)=Last(L)=dummy@}

{Definisi primitif, boleh digunakan tanpa perlu realisasi}
function IsEmpty (L: List) boolean
{mengirim true jika L kosong, false jika tidak kosong}
function Alokasi(X: infotype) address
{menghasilkan address elemen list hasil alokasi dengan Info=X, Next=Nil}
{bila alokasi gagal menghasilkan Nil}
procedure Dealokasi (input/output P: address)
{I.S.: P terdefinisi}
{F.S.: P dealokasi (dikembalikan ke sistem) }

{Realisasikan fungsi dan prosedur berikut}
procedure InsertLast(input/output L: List, input P:address)
{I.S.: Sembarang, P sudah dialokasi}
{F.S.: P ditambahkan sebagai elemen terakhir yang baru}
procedure DeleteLast(input/output L: List, output P:address)
{I.S.: List tidak kosong}
{F.S.: P alamat elemen terakhir list yang dihapus, elemen list berkurang
satu (mungkin jadi kosong)}
function SearchX (L: List, X: infotype) address
{Mencari apakah ada elemen dengan Info(P)=X}
{Jika ada, kembalikan address elemen tersebut, jika tidak kirimkan Nil}

```

Soal 4. Expression Tree, Representasi berkait, Notasi Algoritmik [Bobot 30%]

Sebuah Expression Tree adalah sebuah Binary Tree yang setiap nodenya bertipe Token dengan simpul bukan daun adalah operator dan simpul daun adalah operan. Terdapat empat jenis operator, yaitu +, -, *, /, dan keempat operator tersebut adalah operator binary (memiliki tepat dua operan). Sebuah operan merepresentasikan sebuah bilangan real. Pada soal ini sebuah Expression Tree mungkin kosong, mungkin berisi 1 node (berisi bilangan real) atau berisi lebih dari 1 node (perhatikan bahwa sebuah node bukan daun harus memiliki tepat dua anak).

Berikut adalah contoh sebuah expression tree:

Contoh-0	(tidak ada ekspresi)	(pohon kosong)
Contoh-1	1.5	
Contoh-2	2.0*3.1	
Contoh-3	(5.0-2.0)/3.0+4.0*1.5	

Diberikan potongan kamus ADT binary tree dan primitif-primitif token sebagai berikut. Realisasikan function EvalExpTree dan procedure PrintExpTree (spesifikasi tidak perlu ditulis ulang).

```

KAMUS
(Deklarasi tipe pohon biner)
constant Nil = ... {konstanta pohon kosong,
terdefinisi} constant UnderVal = -99999.9 {nilai tak
terdefinisi} type Token : ... {terdefinisi}
type infotype : Token
type address : ... {terdefinisi}
{Type Pohon Biner}
type BinTree :
address
type node : <Info : infotype, {simpul/akar}
Left : BinTree, {subpohon kiri}
Right : BinTree {subpohon
kanan}>
(Primitif BinTree)
function Akar(B: BinTree) □ infotype
{melemparkan akar pohon biner B}
function Left(B: BinTree) □ BinTree
{melemparkan anak kiri pohon biner
B} function Right(B: BinTree) □
BinTree
{melemparkan anak kanan pohon biner B}
function IsEmpty (B: BinTree) □ boolean
{mengirim true jika B pohon kosong}

(Primitif token)
function IsOperator(T: Token) □ boolean
{melemparkan true jika T adalah sebuah
operator} function IsAdd(T: Token) □ boolean
{T adalah sebuah operator. Melemparkan true jika T adalah sebuah operator
add} function IsSubtract(T: Token) □ boolean

```

```

{T adalah sebuah operator. Melemparkan true jika T adalah sebuah operator pengurang} function IsMultiply(T: Token) Boolean
{T adalah sebuah operator. Melemparkan true jika T adalah sebuah operator pengali} function IsDivide(T: Token) Boolean
{T adalah sebuah operator. Melemparkan true jika T adalah sebuah operator pembagi} function IsOperan(T: Token) Boolean
{melemparkan true jika T adalah sebuah operand} function getRealValue(T: Token) Real
{T adalah sebuah operand. Melemparkan nilai konversi T menjadi sebuah bilangan real}

{Realisasikan fungsi dan prosedur berikut} function EvalExpTree (B: BinTree) Real
{melemparkan nilai yang merupakan hasil evaluasi B. Jika kosong, melemparkan UndefVal} procedure PrintExpTree (Input B: BinTree)
{I.S.: B mungkin kosong}
{F.S.: ekspresi yang direpresentasikan oleh B tercetak. Lihat contoh dibawah ini}

```

Berikut adalah contoh hasil/keluaran fungsi EvalExpTree dan prosedur PrintExpTree (contoh ini bersesuaian dengan contoh pada tabel berikutnya):

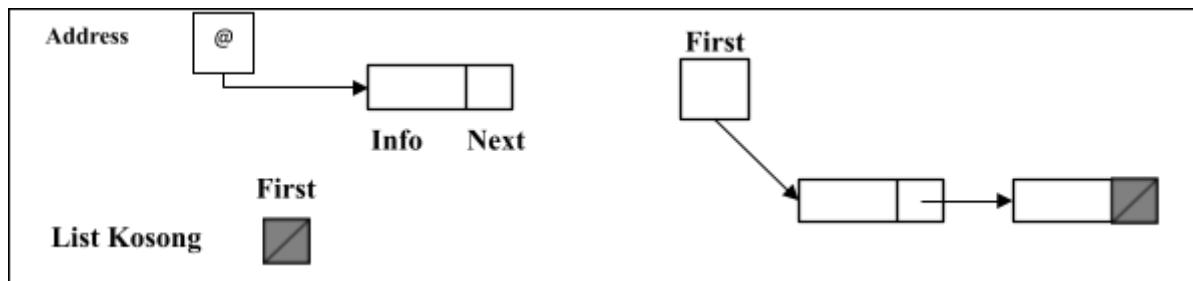
	Hasil fungsi EvalExpTree	Keluaran PrintExpTree (Alternatif 1, tanpa bonus nilai)	Keluaran PrintExpTree (Alternatif 2, Bonus nilai: 10)
Contoh-0	UndefVal	(tidak ada)	(tidak ada)
Contoh-1	1.5	(1.5)	1.5
Contoh-2	6.2	(2.0*3.1)	2.0*3.1
Contoh-3	6.0	((5.0-2.0)/3.0)+(4.0*1.5))	(5.0-2.0)/3.0+4.0*1.5

Perhatikan bahwa alternatif 2 PrintExpTree merupakan penyempurnaan dari alternatif 1 dengan menghilangkan tanda kurung yang tidak perlu. Anda cukup mengimplementasikan salah satu alternatif prosedur PrintExpTree. Tuliskan nomor alternatif yang dikerjakan sebagai spesifikasi prosedur.

UAS SEMESTER I – 2012/2013
IF2030 – Algoritma dan Struktur Data

Soal 1. List, Stack, Queue [Total Bobot: 40%]

Berikut adalah ilustrasi suatu lis linier dan elemennya:



Berikut adalah kamus umum ADT list linier di atas:

```

1 KAMUS
2   constant Nil : ... { terdefinisi }
3
4   type infotype : ... { terdefinisi }
5   } type address : pointer to
6   ElmtList
7   type ElmtList : < info : infotype, next : address
8   > type List : < First : address >
9   { Deklarasi nama untuk variabel kerja }
10  {   L : List }
11  {   P : address }
12  { Maka penulisan First(L) menjadi L.First }
13  {   P↑.info menjadi Info(P); P↑.next menjadi Next(P) }
{ List kosong: First(L) = Nil }

{ Primitif-primitif yang TIDAK PERLU direalisasikan tapi boleh
digunakan } procedure CreateEmpty (output L : List)
{ I.S. Sembarang }
{ F.S. Terbentuk list L kosong: First(L) diinisialisasi dengan Nil
} function Alokasi (X : infotype) address
{ Mengalokasikan memori untuk 1 elemen list dengan Info(P)=X dan
  Next(P)=Nil Jika alokasi berhasil mengembalikan address elemen baru,
  jika gagal Nil }
procedure DeAlokasi (input/output P : address)
{ I.S. P adalah hasil alokasi, P ≠ Nil }
{ F.S. Alamat P didealokasi, dikembalikan ke sistem. P tidak redefinisi
} function IsEmpty (L : List) boolean
{ Mengirimkan true jika list kosong, false jika tidak kosong }
procedure InsertFirst (input/output L : List, input P :
address)
{ I.S. List L mungkin kosong, P sudah dialokasi, P ≠ Nil, Next(P)=Nil }
{ F.S. P adalah elemen pertama list L }
{ Insert sebuah elemen beralamat P sebagai elemen pertama list linier L
  yang mungkin kosong }
procedure InsertLast (input/output L : List, input P : address)
{ I.S. List L mungkin kosong, P sudah dialokasi, P ≠ Nil, Next(P)=Nil }
{ F.S. P adalah elemen terakhir list L }
{ Insert sebuah elemen beralamat P sebagai elemen terakhir dari list
  linier L yang mungkin kosong }
procedure DeleteFirst (input/output L : List, output P : address)

```

	{ I.S. List L tidak kosong, minimal 1 elemen }
--	--

```

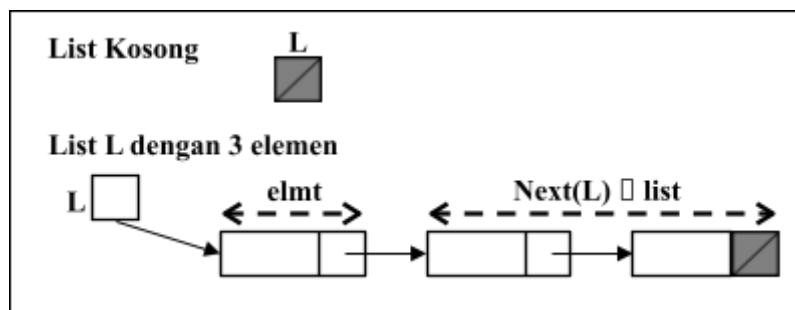
{ F.S. First(L) "maju", mungkin bernilai Nil (List L menjadi
kosong), P berisi alamat elemen pertama L }
{ Menghapus elemen pertama L, P adalah alamat elemen pertama L
sebelum penghapusan, First(L) yang baru adalah Next(L) }
procedure DeleteLast (input/output L : List, output P : address)
{ I.S. List L tidak kosong, minimal mengandung 1 elemen }
{ F.S. P berisi alamat elemen terakhir L, L berkurang elemen terakhirnya,
mungkin menjadi kosong }
{ Menghapus elemen terakhir L, P adalah alamat elemen L terakhir
sebelum penghapusan }

```

Soal 2. List Rekursif [Total Bobot : 30%]

Diketahui definisi List Rekursif adalah sebagai berikut:

- Basis: list kosong adalah list
- Rekuren: list tidak kosong terdiri atas sebuah elemen dan sisanya adalah list



Berikut ini adalah kamus umum untuk list rekursif:

```

KAMUS
{ List rekursif direpresentasikan dengan pointer
} constant Nil : ... { redefinisi }

type infotype : ... { redefinisi }
type address : pointer to ElmtList
type ElmtList : < info : infotype, next : address
> type List : address
{ Deklarasi nama untuk variabel kerja }
{ L : List }
{ P : address }
{ Maka penulisan First(L) menjadi L }
{ P.info menjadi Info(P); P.next menjadi Next(P) }
{ Primitif-primitif yang TIDAK PERLU direalisasikan tapi boleh digunakan }

{ function Alokasi (e : infotype) address

{ Mengalokasi memori untuk 1 elemen list dengan Info(P)=e dan
Next(P)=Nil. Jika alokasi berhasil mengembalikan address elemen baru,
jika gagal Nil }
function IsEmpty (L : List) boolean
{ Tes apakah sebuah list L kosong }
{ Mengirimkan true jika list kosong, false jika tidak kosong }
{ function FirstElmt (L : List) infotype
{ Mengirimkan elemen pertama sebuah list L yang tidak kosong
{ function Tail (L : List) List
{ Mengirimkan list L yang tidak kosong tanpa elemen pertamanya }
{ Mungkin yang dikirimkan adalah sebuah list kosong
{ function Konso (e : infotype, L : List) List

```

```

{ Mengirimkan list L dengan tambahan e sebagai elemen pertamanya }
{ Jika alokasi gagal, mengirimkan L }
function Kons• {L : List, e : infotype} : List
{ Mengirimkan list L dengan tambahan e sebagai elemen terakhir }
{ Jika alokasi gagal, mengirimkan L }

```

Realisasikan semua primitif berikut dalam notasi algoritmik tanpa menuliskan kembali spesifikasinya. Semua pemrosesan list, jika memungkinkan harus dilakukan secara rekursif (tanpa menggunakan *loop*). Realisasi suatu primitif boleh menggunakan primitif lainnya. Jika perlu menambah fungsi/prosedur lain, buatlah definisi, spesifikasi, dan realisasinya.

- Jika list linier tersebut akan diadaptasi menjadi stack dan queue dengan elemen bertipe character, perubahan apa saja yang harus dilakukan terhadap struktur data list linier tersebut, yaitu kide pada baris 1 s.d. 13 (termasuk adaptasi nama jika diperlukan)?
 - Stack direpresentaskan sebagai list yang hanya dapat diakses melalui elemen TOP, sebagai tempat baik untuk melakukan penambahan atau pun penghapusan elemen.
 - Queue direpresentasikan sebagai sebuah list dengan elemen HEAD sebagai tempat penghapusan elemen dan elemen TAIL sebagai tempat untuk penambahan elemen baru.

Cara menjawab pertanyaan tersebut: untuk masing-masing struktur data, tuliskan nomor baris yang harus diubah dan kode barunya.

- Mengikuti struktur stack dan queue yang Anda buat pada butir a, tuliskan implementasi dari primitif-primitif struktur data stack dan queue sesuai dengan spesifikasi di bawah ini (spesifikasi tidak perlu ditulis ulang). Anda dipersilakan untuk menggunakan primitif-primitif di list linier jika diperlukan

```

{ *** PRIMITIF STACK *** }
procedure CreateEmpty (output S: Stack)
{ I.S. S Sembarang }
{ F.S. Stack S terdefinisi sebagai stack kosong. Ciri stack kosong : TOP bernilai Nil } function IsEmpty (S: Stack) boolean
{ Mengirim true jika S adalah Stack kosong: lihat definisi di atas }
procedure Push (input/output S: Stack, input X : infotype)
{ Menambahkan X sebagai elemen TOP pada Stack S }
{ I.S. S terdefinisi, mungkin kosong. X terdefinisi. }
{ F.S. Jika alokasi X berhasil, menjadi nilai dari TOP yang baru dan elemen S bertambah satu. Jika alokasi gagal, S tetap. }
procedure Pop (input/output S: Stack, output X : infotype)
{ Menghapus elemen TOP dari Stack S }
{ I.S. S tidak mungkin kosong. X sembarang. }
{ F.S. X adalah nilai elemen TOP yang lama. Elemen S berkurang satu, mungkin menjadi kosong. Elemen TOP lama didealokasi. }

{ *** PRIMITIF QUEUE *** }
procedure CreateEmpty (output Q: Queue)
{ I.S. Q Sembarang }
{ F.S. Sebuah Q kosong terbentuk. }
{ Ciri queue kosong : HEAD dan TAIL bernilai Nil }
function IsEmpty (Q: Queue) boolean
{ Mengirim true jika Q adalah Queue kosong: lihat definisi di atas }
procedure Add (input/output Q: Queue, input X : infotype)
{ Proses: Mengalokasi X dan menambahkan X pada bagian akhir dari Q jika alokasi berhasil, jika alokasi gagal, Q tetap. }
{ I.S. Q mungkin kosong. X terdefinisi. }
{ F.S. Jika alokasi X berhasil, X menjadi nilai dari TAIL yang baru. Elemen Q bertambah satu}
procedure Del (input/output Q: Queue, output X : infotype)
{ Proses: Menghapus X pada bagian HEAD dari Q dan mendealokasi elemen HEAD }
{ I.S. Q tidak mungkin kosong, X sembarang. }
{ F.S. X berisi nilai elemen HEAD yang lama, HEAD "maju" HEAD lama didealokasi. Elemen Q berkurang satu, mungkin menjadi kosong. }

```

- c. Buatlah sebuah program yg menerima masukan karakter huruf kecil ['a'..'z'] secara berulang-ulang sampai pengguna memasukkan '#' (karakter '#' diabaikan). Kemudian program akan mencetak ke layar "Masukan adalah Palindrom" jika masukan merupakan palindrom . dan jika tidak mencetak "Masukan bukan Palindrom". Palindrom adalah suatu kata yang terdiri atas huruf-huruf yang sama jika ditelusuri dari depanmaupun dari belakang. Jika hanya karakter '#' yang dimasukkan, tuliskan pesan "Tidak ada yang diproses". Gunakan definisi dan primitif-primitif struktur data stack dan/atau queue di atas untuk menentukan palindrom.

Masukan Kumpulan Karakter Valid	Keluaran di Layar
kasurrusak	Masukan adalah Palindrom
cobamasuk	Masukan bukan Palindrom
tamat	Masukan adalah Palindrom

Soal 3. Pohon Biner [Total Bobot : 30%]

Diberikan definisi ADT Pohon Biner yang digunakan untuk menyimpan nilai bertipe integer sebagai berikut:

```

{ *** Definisi Type Pohon Biner ***
} constant Nil : ... { terdefinisi }
type infotype : integer
type addressTree : pointer to
node type node : < info      :
infotype,
        left : addressTree,
        right : addressTree >
type BinTree : addressTree

{ Selektor : Jika P adalah BinTree,
  Akar(P) untuk mengakses P^.Info
  Left (P) untuk mengakses
  P^.Left
  Right (P) untuk mengakses P^.Right )
{ Definsi pohon biner P kosong : P = Nil }

{ *** Konstruktor, boleh digunakan tanpa direalisasikan ***
procedure MakeTree {input Akar : infotype, input L : BinTree, input R : BinTree, output P :
: BinTree }
{ I.S. Sembarang }
{ F.S. Menghasilkan sebuah pohon P }
{ Menghasilkan sebuah pohon biner P dari A, L, dan R, jika alokasi berhasil }
{ Menghasilkan sebuah pohon biner P kosong jika ada alokasi yang gagal }

{ *** Predikat-predikat penting, boleh digunakan tanpa direalisasikan ***
} function IsTreeEmpty (P : BinTree) : boolean
{ Mengirimkan true jika P adalah pohon biner yang kosong
} function IsOneElmt (P : BinTree) : boolean
{ Mengirimkan true jika P tidak kosong dan hanya terdiri atas satu elemen
} function IsUnderLeft (P : BinTree) : boolean
{ Mengirimkan true jika P tidak kosong dan P hanya memiliki subpohon kiri
} function IsUnderRight (P : BinTree) : boolean
{ Mengirimkan true jika P tidak kosong dan P hanya memiliki subpohon
kanan } function IsBiner (P : BinTree) : boolean
{ Mengirimkan true jika P tidak kosong dan P subpohon kiri dan kanan }

{ *** Fungsi-fungsi lain, buatlah realisasinya ***
} function NBElmtTree (P : BinTree) : integer
{ Mengirimkan banyaknya node pada P yang mungkin kosong }

function LevelTree (P : BinTree) : integer

```

```
function IsBalancedTree (P : BinTree) : integer
{ Mengirimkan true jika P adalah pohon seimbang.
  Definisi pohon seimbang:
    • Perbedaan tinggi subpohon kiri dengan kanan maksimum 1.
    • Perbedaan banyaknya simpul subpohon kiri dengan kanan maksimum 1.
    • Subpohon kiri dan kanan adalah pohon seimbang juga.
  Prekondisi : P tidak kosong, minimum terdiri atas 1 elemen yaitu akar. }

function RemoveDaun (P : BinTree, X : infotype) : BinTree
{ Menghasilkan sebuah pohon biner baru yang merupakan salinan dari P tanpa semua daun
  yang bernilai X. P mungkin kosong. Diasumsikan alokasi elemen baru selalu berhasil }
```

Realisasikan fungsi-fungsi yang telah disebutkan di atas. Jika perlu menambah fungsi/prosedur baru, tuliskan definisi, spesifikasi, dan realisasinya.

UAS SEMESTER I – 2014/2015

UAS IF2110/Algoritma & StrukturData
Tanggal: 15 Desember 2014

Waktu: 120 menit
Sifat: Tutup Buku dan Catatan

Petunjuk Pengerajan:

1. Periksalah lembar soal yang Anda terima, apakah Anda sudah memiliki lengkap **4 soal dalam 6 halaman**. Jika belum lengkap, mintalah ganti pada pengawas.
2. Bacalah setiap soal dengan cermat. Kerjakan yang lebih mudah terlebih dahulu.
3. Tuliskan jawaban setiap soal pada lembar jawaban yang disediakan. Jawaban tidak harus terurut, namun setiap nomor soal harus dimulai pada halaman baru (boleh di lembar yang sama dengan soal sebelumnya).
4. Semua soal dikerjakan dalam **notasi algoritmik**.
5. Berikan **nama, NIM, dan kelas** pada tiap lembar jawaban yang terisi.
6. Kerjakan setiap jawaban dengan **menggunakan pensil**. Jawaban yang tidak menggunakan pensil akan dikenakan penalti.

Soal I. ADT Array

Diketahui ADT Array of Integer yang diimplementasikan secara statik dan implisit sebagai berikut:

```
{
  MODUL TABEL INTEGER DENGAN ALOKASI STATIK DAN IMPLISIT
  { Berisi Definisi dan semua primitif pemrosesan tabel statik dan implisit }
  { Penempatan elemen selalu rapat kiri, dan elemen-elemen selalu kontigu }
  { Kamus Umum }
  constant IdxMax : integer = 100
  constant IdxUndef : integer = 0  { indeks tak terdefinisi }
  constant ValUndef : integer = -999 { value tak terdefinisi }
  { Definisi elemen dan koleksi objek }
  type IdxType : integer { type indeks }
  type ElType : integer { type elemen tabel }
  type TabInt : < TI : array [1..IdxMax] of ElType, { memori tempat penyimpan elemen }

  { lengkapi deklarasinya jika perlu } >
  { Definisi, jika T : TabInt :
    Definisi elemen pertama : T.TI[GetFirstIdx(T)]
    Definisi elemen terakhir : T.TI[GetLastIdx(T)] }
  { Konstruktor: create tabel kosong }
  procedure MakeEmpty {output} T : TabInt
  { I.S. sembarang }
  { F.S. Terbentuk tabel T kosong dengan kapasitas IdxMax elemen }
  function GetFirstIdx (T : TabInt) → IdxType
  { Mengirimkan indeks elemen pertama. Prekondisi : Tabel tidak kosong}
  function GetLastIdx (T : TabInt) → IdxType
  { Mengirimkan indeks elemen terakhir. Prekondisi : Tabel tidak kosong}
  function GetNBElem (T : TabInt) → integer
  { Mengirimkan banyaknya elemen dalam tabel. Prekondisi : Tabel tidak kosong }
}
```

- a. Tuliskan deklarasi tipe **TabInt** secara lengkap, kemudian implementasikan prosedur **MakeEmpty**, fungsi **GetLastIdx**, dan fungsi **GetNBElem** sesuai dengan definisi yang dibuat, tanpa menulis ulang spesifikasi (definisi/header fungsi/prosedur tetap harus dituliskan).

- b. Implementasi fungsi **MaxInRange** berikut tanpa menulis ulang spesifikasi.

```
function MaxInRange (T : TabInt, iAw : IdxType, iAk : IdxType) → IdxType
{ Mengembalikan indeks nilai maksimum pada T yang terletak antara indeks [iAw..iAk].
  Prekondisi: 1 ≤ iAw ≤ iAk ≤ GetLastIdx(T) }
{ Contoh: jika elemen T = [5,3,6,8,2], MaxInRange(T,1,5) = 4 (8 adalah T.TI[4])
  MaxInRange(T,1,3) = 3 (6 adalah T.TI[3]) }
```

- c. Tuliskan deklarasi, spesifikasi, dan implementasi prosedur **MaxSort** yang akan mengurutkan elemen tabel T pada parameter prosedur sehingga **terurut menurun**. Tabel mungkin kosong.

Tidak diperkenankan membuat type/fungsi/prosedur lain. Fungsi/prosedur yang sudah didefinisikan di soal sebelumnya dapat digunakan pada soal sesudahnya.

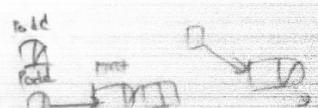
Soal 2. ADT List Linier dengan elemen unik dan tidak terurut

Berikut ini definisi ADT List Linier yang elemennya unik dan tidak terurut, dengan First eksplisit.

```

{ MODUL ADT List Linier dengan representasi list linier berelemen unik dan tidak terurut}
{ Berisi definisi dan primitif pemrosesan ADT Set }
{ Kamus Umum }
constant Nil : ...
type infotype : integer≥0
type elmtList : < Info : infotype, Next : address >
type address : pointer to elmtList
{ Definisi elemen dan koleksi objek }
type LIST : < First : address>
{ *** Selektor *** }
{ Jika L : LIST dan P : address, maka penulisan :
  First(L) menjadi L.First; Next(P) menjadi P^.Next; Info(P) menjadi P^.Info }
{ ***Konstruktur: create set kosong*** }
procedure CreateEmpty (output L : LIST)
{ I.S. sembarang }
{ F.S. Terbentuk himpunan kosong L }
{ ***Primitif Alokasi dan Dealokasi *** }
procedure Alokasi (output P : address, input X : infotype)
{ I.S. X terdefinisi }
{ F.S. P berisi alamat elemen list dengan Info(P) = X dan Next(P)=Nil; P Nil jika alokasi
  tidak berhasil }
procedure Dealokasi (input/output P : address)
{ I.S. P terdefinisi }
{ F.S. P didealokasi dan jadi Nil }
{ ***Primitif Lain*** }
function IsEmpty (L : LIST) → boolean
{ Mengirimkan true jika L adalah himpunan kosong. }
function NbGTx (L : LIST, X : integer) → integer≥0
{ Mengembalikan jumlah elemen L yang bernilai lebih besar dari X }
procedure CopyOdd (input Lin : LIST, output Lout : LIST)
{ I.S. Lin terdefinisi, mungkin kosong. Lout sembarang. }
{ F.S. Lout terdefinisi, berisi semua elemen ganjil pada Lin dengan urutan yang sama.
  Jika ada alokasi yang gagal, Lout berisi elemen-elemen yang sudah berhasil
  Disalin sebelum terjadi kegagalan. }
procedure PrintSubList (input pAw, pAkh : address)
{ I.S. pAw dan pAkh terdefinisi dan merupakan elemen dari list yang sama;
  pAw muncul sebelum atau sama dengan pAkh }
{ F.S. Semua elemen mulai dari pAw hingga pAkh ditampilkan di layar }
{ Contoh: L = [ 3 2 1 5 6 ], maka PrintSubList(@2,@4) dimana @i merupakan alamat elemen
  ke-i dari list, menghasilkan 2 1 5 }
procedure PrintSubListEqX (input L : LIST, X : integer≥0)
{ I.S. L terdefinisi, mungkin kosong }
{ F.S. Semua sublist L yang jika dijumlahkan elemennya = X akan ditampilkan di layar
  Jika L kosong, tampilkan pesan "List kosong" }
{ Contoh: L = [ 3 2 1 5 6 ], maka PrintSubListEqX(L,6) menghasilkan
  3 2 1
  1 5
  6 }
```

Tuliskan algoritma fungsi **NbGTx**, prosedur **CopyOdd**, prosedur **PrintSubList**, dan prosedur **PrintSubListEqX** tanpa menulis ulang spesifikasi, dengan pendekatan iteratif. Tidak diperkenankan menambahkan fungsi/prosedur antara baru.



Soal 3. Pohon Biner

Berikut adalah definisi kamus ADT Pohon Biner yang direpresentasikan dengan pointer dengan elemen integer yang tidak unik dan tidak terurut.

```

{ ADT BinTree }
{ Pohon Biner dengan representasi pointer }
constant Nil : ... { terdefinisi }
type infotype : integer
type Node : < Info : Infotype, Left : BinTree, Right : BinTree >
type address : pointer to Node
type BinTree : address

{ Selektor : Jika P adalah BinTree: Akar(P) untuk mengakses bagian P^.Info
Left(P) untuk mengakses bagian P^.Left
Right(P) untuk mengakses bagian P^.Right }

{ Primitif-Primitif }
procedure BuildTree (output P : BinTree)
{ Membentuk pohon biner P dari pita karakter. }
{ I.S.: Sembarang, F.S.: P terdefinisi }
function IsTreeEmpty (P : BinTree) → boolean
{ Mengirimkan true jika P adalah pohon biner yang kosong }
function IsOneElmt (P : BinTree) → boolean
{ Mengirimkan true jika P tidak kosong dan hanya terdiri atas 1 elemen }
function IsUnderLeft (P : BinTree) → boolean
{ Mengirimkan true jika pohon biner tidak kosong P adalah pohon underleft: hanya mempunyai subpohon kiri }
function IsUnderRight (P : BinTree) → boolean
{ Mengirimkan true jika pohon biner tidak kosong P adalah pohon unerright: hanya mempunyai subpohon kanan }
function IsBiner (P : BinTree) → boolean
{ Mengirimkan true jika pohon biner tidak kosong P adalah pohon biner: mempunyai subpohon kiri dan subpohon kanan }

```

Berikut adalah definisi kamus ADT Queue yang direpresentasikan dengan tabel alternatif 1 (Head selalu di awal):

```

{ ADT Queue dengan representasi Tabel alternatif 1 : Head selalu di 1}
constant Nil = 0
constant Max = 100
{ Definisi elemen dan address }
type infotype : integer
type address : integer { indeks tabel }
{ Versi 1 : tabel statik, Head dan Tail eksplisit }
type Queue : < T : array[1..Max] of infotype; { tabel penyimpan elemen }
          HEAD : address, { alamat penghapusan }
          TAIL : address { alamat penambahan }

{ Definisi Queue kosong: HEAD=Nil; TAIL=Nil. }
{ Catatan implementasi: T[0] tidak pernah dipakai }
{***** AKSES (Selektor) *****}
{ Jika Q adalah Queue, maka akses elemen :
  Head(Q) (Q).HEAD
  Tail(Q) (Q).TAIL
  InfoHead(Q) (Q).T[(Q).HEAD]
  InfoTail(Q) (Q).T[(Q).TAIL] }

/***** Prototype *****/
function IsEmpty(Q : Queue) → boolean
{ Mengirim true jika Q kosong: lihat definisi di atas }
function IsFull(Q : Queue) → boolean
{ Mengirim true jika tabel penampung elemen Q sudah penuh }
{ yaitu mengandung elemen sebanyak MaxEl }
function NBElmt(Queue Q) → integer
{ Mengirimkan banyaknya elemen queue. Mengirimkan 0 jika Q kosong }
{*** Creator ***}
procedure CreateEmpty(input/output Q : Queue)
{ I.S. sembarang
  F.S. Sebuah Q kosong terbentuk }

```

```
(*** Primitif Add/Delete ***)
procedure Add (input/output Q : Queue, input X : infotype)
{ Proses: Menambahkan X pada Q dengan aturan FIFO }
{ I.S. Q mungkin kosong, tabel penampung elemen Q TIDAK penuh }
{ F.S. X menjadi TAIL yang baru, TAIL "maju" }
procedure Del(input/output Q : Queue, input/output X : infotype)
{ Proses: Menghapus X pada Q dengan aturan FIFO }
{ I.S. Q tidak mungkin kosong }
{ F.S. X = nilai elemen HEAD pd I.S., seluruh elemen tabel "maju"; Q mungkin kosong }
```

Dengan mengasumsikan bahwa konflik penamaan beberapa tipe dari kedua ADT tersebut tidak berpengaruh, buatlah **program utama** (jangan lupa untuk **use BinTree** dan **use Queue**) yang:

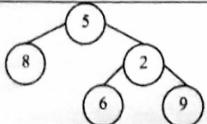
- a. mengandung definisi, spesifikasi, dan implementasi dari fungsi **SearchX**, prosedur **TreeToQ**, dan prosedur **FindDescendantsOfX**, yang diberikan spesifikasinya di bawah ini dalam notasi algoritmik (spesifikasi tidak perlu ditulis, definisi/header fungsi/prosedur tetap harus ditulis). Semua fungsi dan procedure yang tersedia di ADT Pohon Biner dan Queue boleh digunakan. Jika harus membuat fungsi/prosedur lain, buat definisi/header, spesifikasi, dan implementasi/body-nya.

```
function SearchAddrX (P : BinTree, X : Infotype) → address
{ Mengirimkan alamat X pada pohon P; mengirimkan Nil jika X tidak ada pada pohon P
  Jika ada lebih dari satu X di pohon P, ambil X yang paling kiri }
procedure TreeToQ (input P : BinTree, input/output Q : Queue)
{ I.S.: P terdefinisi
  F.S.: Seluruh info node pada pohon P masuk ke Queue Q secara pre-order}
procedure FindDescendantsOfX (input P : BinTree, input X : Infotype,
                               input/output Q : Queue)
{ I.S.: P terdefinisi, X pasti ada sebagai info pada salah satu node pohon P
  F.S.: Jika PtrX adalah alamat node pada pohon P dengan info=X, info dari seluruh node
        yang merupakan descendant PtrX masuk ke Queue Q}
```

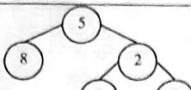
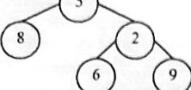
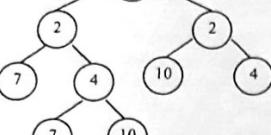
- b. Buat program utama yang melakukan hal-hal berikut:
- Mendefinisikan sebuah pohon biner misalnya P dengan memanggil prosedur **BuildTree**. Asumsikan semua proses pembentukan pohon P seluruhnya ditangani oleh prosedur tersebut sehingga tidak perlu menambah baris kode apa pun.
 - Menerima masukan sebuah infotype, misalnya X, dari keyboard.
 - Jika X ada di pohon biner P, maka cetak ke **layar** semua descendant dari X di P. Format pencetakan bebas. Jika X adalah daun, tuliskan ke layar "X tidak punya descendant" dengan X diganti dengan nilai aktualnya.
Jika pohon biner P kosong, tuliskan ke layar "Pohon biner kosong"
 - Jika X tidak ada di P, tuliskan ke layar "X tidak ada di pohon biner" dengan X diganti nilai aktualnya.

Gunakan sedapat mungkin semua fungsi/prosedur yang tersedia.

Contoh masukan dan keluaran:

Masukan P (BinTree)	Masukan X (infotype)	Isi Queue setelah pemanggilan FindDescendantsOfX, dan tampilan di layar
P=Nil {pohon kosong}	5	Tampilan layar: Pohon biner kosong
	10	Tampilan layar: 10 tidak ada di pohon biner



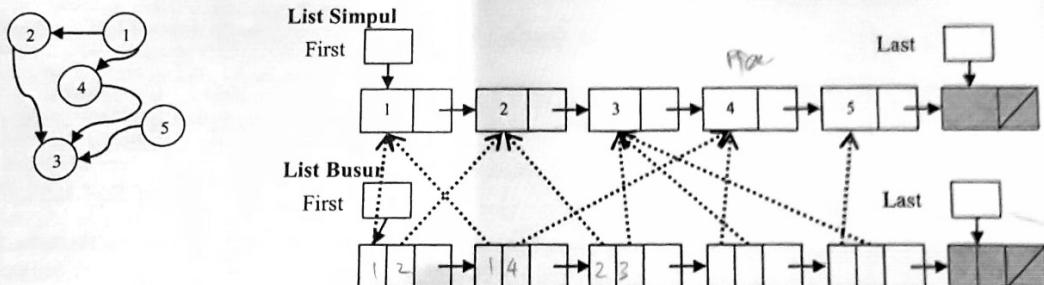
Masukan P (BinTree)	Masukan X (infotype)	Isi Queue setelah pemanggilan FindDescendantsOfX, dan tampilan di layar
	8	Queue Q kosong Tampilan layar: 8 tidak punya descendant
	2	Isi Q: [6, 9] Tampilan layar: 6 9
	2	Isi Q: [7, 4, 7, 10] Tampilan layar: 7 4 7 10

Soal 4. Graph Berarah

Graph Berarah adalah graph dengan busur yang memiliki arah. Artinya, jika pada graph terdefinisi busur (a,k) berarti ada busur dengan titik awal simpul a (*predecessor*) menuju simpul k (*successor*).

Graph Berarah akan diimplementasikan dengan menggunakan variasi *edge list*, representasi berkait dengan elemen *dummy* di akhir.

Berikut contoh sebuah graph berarah dan representasinya:



Berikut adalah definisi ADT untuk Graph Berarah tersebut:

```
{ Graph berarah diimplementasi sebagai Edge List dengan representasi berkait }
constant Nil : ...
type adrVertex : pointer to Vertex
type Vertex : < Id : integer,
               Next : adrVertex >
type VList : < First, Last : adrVertex >
type adrEdge : pointer to Edge
type Edge : < Prec : adrVertex,           {address simpul asal (predecessor)}
            Succ : adrVertex,           {address simpul tujuan (successor)}
            Next : adrEdge >
type EList : < First, Last : adrEdge >
type Graph : < V : VList, E : EList >
{ *** Selektor *** }
{ Jika G : Graph, V : VList, E : EList, Pv : adrVertex, Pe : adrEdge, maka penulisan:
Vertices(G) menjadi G.V; Edges(G) menjadi G.E;
FirstV(V) menjadi V.First; LastV(V) menjadi V.Last;
FirstE(E) menjadi E.First; LastE(E) menjadi E.Last;
Id(Pv) menjadi Pv.Id; Next(Pv) menjadi Pv.Next;
Prec(Pe) menjadi Pe.Prec; Succ(Pe) menjadi Pe.Succ; Next(Pe) menjadi Pe.Next }
```

UAS IF2110/Algoritma & StrukturData
Tanggal: 15 Desember 2014

Waktu: 120 menit

Sifat: Tutup Buku dan Catatan

```
{ *** Konstruktor *** }
procedure CreateGraph (X : integer, output L : Graph)
{ I.S. Sembarang; F.S. Terbentuk Graph satu simpul dengan Id=X dan busur kosong }
procedure CopyGraph (input GIn : Graph, output GOut : Graph)
{ I.S. GIn terdefinisi; F.S. GOut berisi copy dari GIn }
{ *** Manajemen Memory List Simpul *** }
function AlokVertex (X : integer) → adrVertex
{ Mengembalikan address (misalnya Pv) hasil alokasi simpul dengan Id(Pv)=X dan
  Next(Pv)=Nil. Jika alokasi gagal, mengembalikan Nil. }
procedure DealokVertex (input/output Pv : adrVertex)
{ I.S. Pv terdefinisi; F.S. Pv dikembalikan ke sistem }
{ *** Manajemen Memory List Busur *** }
function AlokEdge (Pv1, Pv2 : adrVertex) → adrEdge
{ Mengembalikan address (misalnya Pe) hasil alokasi busur (v1,v2) dengan Prec(Pe)=Pv1,
  Succ(Pe)=Pv2, Next(Pe)=Nil. Jika alokasi gagal, mengembalikan Nil. }
procedure DealokEdge (input/output Pe : adrEdge)
{ I.S. Pe terdefinisi; F.S. Pe dikembalikan ke sistem }
{ *** Fungsi/Prosedur Lain *** }
function SearchVertex (G : Graph, X : integer) → adrVertex
{ mengembalikan address simpul dengan Id=X jika ada pada graph G, Nil jika tidak ada }
function SearchEdge (G : Graph, prec : integer, succ : integer) → adrEdge
{ mengembalikan address busur yang menyimpan info busur (prec,succ) jika ada pada graph
  G, Nil jika tidak ada. prec dan succ mungkin bukan simpul pada G. }
procedure InsVertex (input/output G : Graph, input X : integer)
{ Menambahkan simpul X ke dalam G }
{ I.S. G terdefinisi, X terdefinisi dan belum ada pada G. }
{ F.S. Jika alokasi berhasil, X menjadi elemen terakhir Vertices(G).
  Jika alokasi gagal, G tetap }
procedure InsEdge (input/output G : Graph, input prec, succ : integer)
{ Menambahkan busur dari prec menuju succ ke dalam G }
{ I.S. G terdefinisi; prec, succ adalah simpul G; busur (prec,succ) belum ada di G. }
{ F.S. Jika alokasi berhasil, busur (prec,succ) menjadi elemen terakhir Edges(G).
  Jika alokasi gagal, G tetap }
procedure DelVertex (input/output G : Graph, input X : integer)
{ Menghapus simpul X dari G }
{ I.S. G terdefinisi, X terdefinisi dan ada pada G, jumlah simpul pada G lebih dari 1. }
{ F.S. simpul X dan semua busur yang terhubung ke X dihapus dari G. }
procedure DelEdge (input/output G : Graph, input prec, succ : integer)
{ Menghapus busur (prec,succ) dari G }
{ I.S. G terdefinisi, busur (prec,succ) ada pada G. }
{ F.S. busur (prec,succ) dihapus dari G. }
```

Implementasikan prosedur **SearchEdge**, **InsEdge**, dan **DelVertex** sesuai definisi dan spesifikasi di atas, tanpa menulis ulang spesifikasinya. Semua primitif lain yang disediakan dapat dipakai tanpa perlu diimplementasikan. Kerjakan dengan pendekatan iteratif. Tidak boleh membuat fungsi/prosedur baru.

UAS SEMESTER I – 2016/2017

UAS IF2110/Algoritma dan Struktur Data
Tanggal: 14 Desember 2016 (Waktu: 120 menit)
Halaman 1 dari 5

PETUNJUK UJIAN:

1. Lembar soal ini terdiri atas **5 halaman** yang terdiri atas **soal 1 s.d. soal 4**. Cek lembar soal, apakah Anda sudah mendapatkan lembar soal yang benar. Jika masih ada kesalahan, tukarkan pada pengawas.
2. Semua soal dikerjakan di lembar jawaban yang disediakan dan jawaban setiap soal diletakkan di lembar yang terpisah dengan soal yang lain.
3. Jawaban ditulis dengan pensil. Bagian identitas tulis dengan bolpoin. Jangan lupa menulis identitas (minimum nama, NIM, Kelas) pada setiap lembar jawaban.
4. Lembar soal boleh dibawa pulang setelah ujian selesai. Kumpulkan hanya lembar jawaban.
5. Semua soal dikerjakan dengan **Notasi Algoritmik**.

Soal 1. LIST REKURSIF (Bobot: 25%)

Diketahui ADT List of Integer dengan first element direpresentasikan secara implisit sebagai berikut:

```
{ List linier direpresentasi dengan pointer }
constant Nil : ...

type infotype : integer
type ElmtList : < Info : infotype, Next : address >
type address : pointer to ElmtList
type List : address { alamat elemen pertama list }

{ *** Selektor *** }
{ Jika L : List dan P : address, First(L) untuk mengakses L (elemen pertama L);
  Next(P) untuk mengakses P^.Next; dan Info(P) untuk mengakses P^.Info }

{*** Predikat pemeriksaan ***}
function IsEmpty (L : List) → boolean
{ Mengirimkan true jika list kosong, false jika tidak kosong }
function IsOneElmt (L : List) → boolean
{ Mengirimkan true jika list hanya berisi satu elemen, false jika tidak }
function SumGenap (L : List) → integer
{ Mengirimkan hasil penjumlahan seluruh elemen list yang bernilai genap, atau nol jika
list kosong }
function IsOrdered (L : List) → boolean
{ Mengirimkan true jika list terurut membesar, false jika tidak.
Asumsi: List tidak kosong }
procedure DelAllMulOfX (input/output L : List, input X : integer)
{ I.S. L terdefinisi dan mungkin kosong }
{ F.S. Seluruh elemen pada L yang merupakan kelipatan dari X sudah dihapus dari L }
```

Tuliskan implementasi fungsi **SumGenap**, fungsi **IsOrdered**, dan prosedur **DelAllMulOfX** (tidak perlu menulis ulang spesifikasinya, definisi/header fungsi/prosedur tetap harus dituliskan). Seluruh fungsi dan prosedur harus diimplementasikan secara **rekursif**.

UAS IF2110/Algoritma dan Struktur Data
Tanggal: 14 Desember 2016 (Waktu: 120 menit)
Halaman 2 dari 5

Soal 2. POHON BINER (Bobot : 25%)

Berikut adalah definisi kamus ADT Pohon Biner yang direpresentasikan dengan pointer dengan elemen karakter yang merepresentasikan ekspresi aritmatika untuk bilangan integer 1 digit. Ekspresi selalu valid. Sebuah ekspresi yang valid:

- Bukan ekspresi kosong
- Ekspresi dapat terdiri atas:
 - o Sebuah **operator**: - (kurang), + (tambah), * (kali), / (div: bagi bulat), % (mod: sisa bagi bulat), ^ (pangkat); diikuti **dua buah operan**, yang masing-masing bisa berupa: sebuah integer 1 digit [0..9] atau sebuah ekspresi aritmatika dalam notasi prefix. Tidak ada pemisah apa pun antara operator dan setiap operan.
 - o Sebuah konstanta integer 1 digit [0..9].

```

{ ADT BinTree }
{ Pohon Biner dengan representasi pointer }
constant Nil : ... { terdefinisi }
type infotype : character { merepresentasikan operan atau operator ekspresi aritmatika
                           bilangan integer 1 digit }
type Node : < Info : infotype, Left : BinTree, Right : BinTree >
type address : pointer to Node
type BinTree : address
{ Selektor : Jika P adalah BinTree: Akar(P) untuk mengakses bagian P^.Info
             Left(P) untuk mengakses bagian P^.Left
             Right(P) untuk mengakses bagian P^.Right }

{ Primitif-Primitif }
procedure BuildTreeFromString (input str : string, input/output idx : integer,
                               output P : BinTree)
{ Membentuk pohon biner P yang merepresentasikan ekspresi aritmatika bilangan integer 1
  digit dari string masukan dalam notasi PREFIX. Pohon P dan subpohnnya SELALU merupakan
  pohon biner. }
{ I.S.: str dan idx terdefinisi, str bukan string kosong.
  F.S.: pohon biner P terbentuk }

function IsTreeEmpty (P : BinTree) → boolean
{ Mengirimkan true jika P adalah pohon biner yang kosong }

function IsOneElmt (P : BinTree) → boolean
{ Mengirimkan true jika P tidak kosong dan hanya terdiri atas 1 elemen }

function Alokasi (X : infotype) → address
{ mengirimkan address hasil alokasi sebuah elemen pohon dengan X sbg inf, atau nil jika
  alokasi tidak berhasil. }

```

- Buatlah implementasi prosedur **BuildTreeFromString** yang telah diberikan spesifikasinya dalam notasi algoritmik (spesifikasi tidak perlu ditulis, definisi/header fungsi/prosedur tetap harus ditulis).
 - Buatlah sebuah program utama (jangan lupa use BinTree) yang mengandung kamus yang diperlukan serta definisi dari fungsi CtoI, Operasi, isOperator, EvalEkspresi dan prosedur TampilkanEkspresi yang diberikan spesifikasinya di bawah ini (spesifikasi tidak perlu ditulis ulang).
- Buat implementasi fungsi **EvalEkspresi** dan prosedur **TampilkanEkspresi**. Implementasi HARUS dilakukan secara rekursif (tanpa menggunakan *loop*). Semua fungsi dan prosedur yang tersedia di ADT Pohon Biner boleh digunakan. Jika harus membuat fungsi/prosedur lain, buat definisi/header, spesifikasi, dan implementasi/body-nya.

```

function CtoI (cc : character) → integer
{ menerima karakter angka cc dan menghasilkan bilangan integer hasil konversinya. }

function Operasi (opr1,opr2 : integer, opt : character ['*', '/', '%', '+', '-', '^']) → integer
{ menerima dua buah operan integer (opr1 dan opr2) dan sebuah operator (opt),
  menghasilkan hasil operasi kedua operan tersebut dengan operator masukan. }

function isOperator(cc : character ['*', '/', '%', '+', '-', '^', '0'..'9']) → boolean
{ menerima sebuah karakter dan menghasilkan true jika cc adalah operator yang valid,
  false jika bukan. }

function EvalEkspresi(P : BinTree) → integer
{ Menerima sebuah pohon P yang merepresentasikan ekspresi aritmatika.
  P tidak kosong, dan fungsi akan mengembalikan hasil evaluasi ekspresi.
  EvalEkspresi memanfaatkan fungsi CtoI dan Operasi. }

procedure TampilkanEkspresi (input P : BinTree)
{ Menampilkan ekspresi aritmatika dalam notasi PREFIX ke layar. }
{ I.S.: P terdefinisi, tidak kosong }
{ F.S.: Ekspresi aritmatika yang direpresentasikannya akan tampil di layar }

```

UAS IF2110/Algoritma dan Struktur Data
Tanggal: 14 Desember 2016 (Waktu: 120 menit)
Halaman 3 dari 5

Selanjutnya, program utama akan melakukan hal-hal berikut.

1. Membaca sebuah string (misalnya **ekspresi**) berisi ekspresi aritmatika dalam notasi prefix (lihat contoh).
2. Mendefinisikan sebuah pohon biner misalnya **P** dengan memanggil prosedur **BuildTreeFromString** yang akan membangun pohon ekspresi berdasarkan string **ekspresi** yang telah dibaca. Pohon **P** diasumsikan selalu berhasil dibangun, dan merupakan pohon ekspresi yang valid dan tidak kosong.
3. Menampilkan ekspresi aritmatika dengan memanggil prosedur **TampilkanEkspresi** serta menampilkan hasil evaluasinya (dengan memanggil fungsi **Evaluasi**)

Gunakan sedapat mungkin semua fungsi/prosedur yang tersedia.

Contoh masukan dan keluaran:

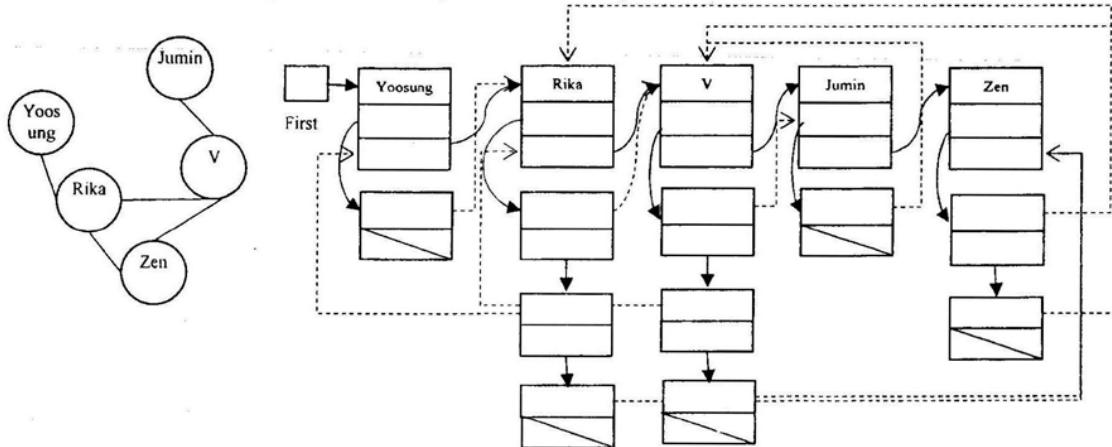
Masukan ekspresi dan P (BinTree) yang terbentuk setelah memanggil BuildTreeFromString	Evaluasi(P)	TampilkanEkspresi(P)	Program Utama
 ekspresi = "+8*69"	62	+ 8 * 6 9	Ekspresi: + 8 * 6 9 Hasil evaluasi = 62
 ekspresi = "+*7+71-41"	59	+ * 7 + 7 1 - 4 1	Ekspresi: + * 7 + 7 1 - 4 1 Hasil evaluasi = 59

Soal 3. Graph Pertemanan (Bobot : 25%)

Graph pertemanan menggambarkan hubungan pertemanan sekelompok orang. Setiap node berisi nama orang dan edge yang menghubungkan dua node menandakan bahwa kedua orang tersebut berteman. Graph ini direpresentasikan dalam *adjacency list*, dalam bentuk multilist sebagai berikut.

1. List simpul (*leader list*) adalah list dengan elemen seluruh simpul yang terdapat di dalam graph. Setiap elemen list berisi identitas nama, pointer ke list simpul temannya (trail), dan pointer ke elemen simpul berikutnya (next). Implementasi dalam bentuk list dengan pencatatan First.
2. List teman (*trailer list*) adalah list yang berisi pointer ke simpul-simpul yang menjadi teman dari simpul utamanya. Apabila dua orang berteman, maka akan terdapat pointer dari simpul orang pertama ke orang kedua dan sebaliknya. Sebagai contoh pada gambar, Yoosung dan Rika berteman, sehingga
 - a. pada list successor dari simpul Yoosung akan terdapat elemen list yang memiliki pointer ke simpul Rika, dan
 - b. pada list successor dari simpul Rika akan terdapat elemen list yang memiliki pointer ke simpul Yoosung.

Berikut ini adalah ilustrasi graph pertemanan dan representasi multilistnya.



UAS IF2110/Algoritma dan Struktur Data
Tanggal: 14 Desember 2016 (Waktu: 120 menit)
Halaman 4 dari 5

Berikut adalah definisi ADT untuk Graph Pertemanan tersebut:

```

{ Graph pertemanan diimplementasi sebagai Multilist }
constant Nil : ...
type adrNode : pointer to Node
type adrFriendNode : pointer to FriendNode
type Node : < Name : String,           (identitas simpul)
            Trail : adrFriendNode,   (pointer ke list trailer (simpul teman))
            Next : adrNode >
type FriendNode : < Friend : adrNode,   (address simpul successor)
                  Next : adrFriendNode >
type Graph : < First : adrNode >
{ *** Selektor *** }
{ Jika G : Graph, Pn : adrNode, Pt : adrFriendNode, maka:
  First(G) untuk mengakses G.First;
  Name(Pn) untuk mengakses Pn.Name; Trail(Pn) untuk mengakses Pn.Trail;
  Next(Pn) untuk mengakses Pn.Next;
  Friend(Pt) untuk mengakses Pt.Friend; Next(Pt) untuk mengakses Pt.Next }
{ *** Konstruktor *** }
procedure CreateGraph (input X : String, output L : Graph)
{ I.S. X terdefinisi; F.S. Terbentuk Graph dengan satu simpul dengan Name=X. }
{ *** Manajemen Memory List Simpul (Leader) *** }
function AlokNode (X : String) → adrNode
{ Mengembalikan address hasil alokasi Simpul X. }
{ Jika alokasi berhasil, maka address tidak Nil, misalnya menghasilkan P, maka Name(P)=X,
  Trail(P)=Nil, dan Next(P)=Nil. Jika alokasi gagal, mengembalikan Nil. }
procedure DealokNode (input P : adrNode)
{ I.S. P terdefinisi; F.S. P dikembalikan ke sistem }
{ *** Manajemen Memory List Successor (Trailer) *** }
function AlokFriendNode (Pn : adrNode) → adrFriendNode
{ Mengembalikan address hasil alokasi. }
{ Jika alokasi berhasil, maka address tidak Nil, misalnya menghasilkan Pt, maka
  Friend(Pt)=Pn dan Next(Pt)=Nil. Jika alokasi gagal, mengembalikan Nil. }
procedure DealokFriendNode (input P : adrFriendNode)
{ I.S. P terdefinisi; F.S. P dikembalikan ke sistem }

{ *** Fungsi/Prosedur Lain *** }
function SearchPerson (G : Graph, X : String) → adrNode
{ Mengembalikan address simpul dengan Name=X jika sudah ada pada graph G,
  Nil jika belum }
procedure InsertPerson (input/output G : Graph, input X : String, output Pn : adrNode)
{ Menambahkan simpul X ke dalam graph, jika alokasi X berhasil. }
{ I.S. G terdefinisi, X terdefinisi dan belum ada pada G. }
{ F.S. Jika alokasi berhasil, X menjadi elemen terakhir G, Pn berisi address simpul X.
  Jika alokasi gagal, G tetap, Pn berisi Nil }
procedure InsertFriendship (input/output G : Graph, input personA, personB : String)
{ Menambahkan busur pertemanan dari personA ke personB dan dari personB ke personA dalam
  G }
{ I.S. G, personA, personB terdefinisi. }
{ F.S. Jika belum ada busur antara personA dan personB di G, maka tambahkan busur
  antara personA dan personB ke G. Perhatikan bahwa dalam representasi multilist ini
  perlu dilakukan dua kali penambahan busur, satu yang berasal dari simpul personA
  dan satu yang berasal dari simpul personB.
  Jika simpul personA/personB belum ada pada G, tambahkan simpul tersebut dahulu.
  Jika sudah ada busur antara personA dan personB di G, maka G tetap. }
function IsFriend (personA, personB: String, G: Graph) → boolean
{Mengembalikan true apabila terdapat busur antara personA dan personB pada G.
Mengembalikan false apabila tidak terdapat busur antara personA dan personB.
Prekondisi: personA dan personB sudah terdefinisi sebagai simpul dalam G}

```

Implementasikan primitif **CreateGraph**, **InsertFriendship**, dan **IsFriend** yang ada sesuai definisi dan spesifikasi di atas (spesifikasi tidak perlu ditulis, definisi/header fungsi/prosedur tetap harus ditulis). Gunakan pendekatan iteratif. Tidak boleh membuat fungsi/prosedur baru.

UAS IF2110/Algoritma dan Struktur Data
Tanggal: 14 Desember 2016 (Waktu: 120 menit)
Halaman 5 dari 5

Soal 4. Studi Kasus Permainan Ular Tangga (Bobot : 25%)

Ular tangga adalah permainan papan yang dimainkan oleh 2 hingga 8 pemain. Papan permainan dibagi dalam kotak-kotak kecil dan di beberapa kotak digambar sejumlah "tangga" atau "ular" yang menghubungkannya dengan kotak lain. Contoh papan permainan ular tangga dapat dilihat pada gambar berikut.

Tidak ada papan permainan standar – setiap orang dapat menentukan papan mereka sendiri dengan jumlah kotak, ular, dan tangga yang berlainan.

Aturan pergerakan bidak adalah sebagai berikut:

- Setiap pemain mulai dengan bidaknya di kotak pertama (kotak dengan nomor terkecil) dan secara bergiliran melemparkan dadu.
- Bidak dijalankan sesuai dengan jumlah mata dadu yang muncul.
- Bila pemain mendarat di ujung bawah sebuah tangga, mereka dapat langsung pergi ke ujung tangga yang lain (misalnya pemain yang mendarat di kotak nomor 11 pada papan contoh akan langsung pindah ke kotak nomor 26).
- Bila mendarat di kotak dengan kepala ular, mereka harus turun ke kotak di ujung bawah ular (misalnya pemain yang mendarat di kotak nomor 19 pada papan contoh akan kembali mundur ke kotak nomor 7).
- Pemenang adalah pemain pertama yang mencapai kotak terakhir (kotak dengan nomor terbesar).
- Bila pemain sudah mendekati kotak terakhir dan ternyata mendapatkan angka yang melebihi jumlah kotak yang tersisa (misalnya bidak di kotak 28 dan pemain mendapatkan angka 5), maka bidak tidak bergerak.

25	26	27	28	29	30
24	23	22	21	19	
14		15	16	20	18
13				17	
12					
11	10		9	8	7
1	2	3	4	5	6

Seorang pemain yang mendapatkan angka 6 dari dadu mendapat giliran jalan kembali, kecuali jika sudah tiba di kotak terakhir (permainan telah selesai).

Berikut ini primitif penting yang harus terdefinisi.

```

procedure initBoard
{I.S. sembarang}
{F.S. papan ular tangga terdefinisi (jumlah kotak serta posisi ular dan tangga) dan siap
digunakan. Jumlah kotak, posisi ular dan tangga merupakan masukan pengguna dan
diasumsikan selalu benar.}
procedure initPlayer
{I.S. sembarang}
{F.S. daftar pemain terdefinisi, bidak setiap pemain ada di kotak pertama}
*** primitif gerakan bidak ***
procedure movePlayer (input/output p : player, input numSquare: integer[1..6])
{I.S. p dan numSquare terdefinisi}
{F.S. bidak pemain p berada pada posisi akhir setelah bergerak sejauh numSquare.
Penentuan posisi akhir memperhitungkan aturan pergerakan bidak di permainan ular tangga}
*** primitif lain ***
function nextPlayer (curPlayer: player) → player
{Menghasilkan informasi pemain berikutnya yang akan bermain}
procedure printBoard
{I.S. Papan ular tangga terdefinisi, posisi semua pemain terdefinisi}
{F.S. Informasi tentang papan ular tangga dan semua pemain tampil di layar}

```

Square adalah sebuah tipe yang menyatakan sebuah kotak pada papan ular tangga.

Player adalah sebuah tipe yang menyatakan seorang pemain (identitas dan posisi bidaknya).

- Jelaskan representasi Papan Ular Tangga (board), square, player, dan Daftar Pemain yang sesuai dengan kebutuhan permainan ular tangga serta definisi primitif yang telah diberikan. Berikan penjelasan sebaik mungkin (bila perlu berikan ilustrasi gambar), karena kejelasan representasi yang dipilih akan menentukan penilaian soal berikutnya. Tuliskan Kamus untuk struktur data yang sesuai dengan representasi yang Anda pilih dalam notasi algoritmik.
 - Tuliskan realisasi dari prosedur `initBoard` dan fungsi `movePlayer`. Anda tidak perlu menulis ulang spesifikasi fungsi `movePlayer`, namun Anda harus melengkapi spesifikasi prosedur `initBoard` dengan definisi prosesnya. Tuliskan definisi, spesifikasi, dan realisasi dari semua fungsi atau prosedur tambahan yang digunakan (jika ada).
 - Buatlah program PermainanUlarTangga yang akan mensimulasikan satu kali permainan ular tangga (hingga ada pemain yang menang). Pemain secara bergantian akan mendapat giliran dan bidak pemain yang sedang mendapatkan giliran akan digerakkan sesuai dengan hasil lemparan dadu yang diperoleh. Untuk setiap giliran pemain, tampilkan data identitas pemain, informasi papan ular tangga sebelum pemain bergerak, hasil lemparan dadu, dan informasi papan ular tangga sesudah pemain bergerak. Kamus yang sudah didefinisikan pada jawaban soal a, b, dan primitif pada soal tidak perlu ditulis ulang.
- Lemparan dadu divakili fungsi berikut.

```

function rollDice → integer[1..6]
{Menghasilkan sebuah integer antara 1-6 secara acak}

```

UAS SEMESTER I – 2017/2018**PETUNJUK UJIAN:**

1. Lembar soal ini terdiri atas 4 halaman yang terdiri atas soal 1 s.d. soal 4. Cek lembar soal, apakah Anda sudah mendapatkan lembar soal yang benar. Jika masih ada kesalahan, tukarkan pada pengawas.
2. Semua soal dikerjakan di lembar jawaban yang disediakan dan jawaban setiap soal diletakkan di lembar yang terpisah dengan soal yang lain.
3. Jawaban ditulis dengan pensil. Bagian identitas tulis dengan bolpoin. Jangan lupa menulis identitas (minimum nama, NIM, Kelas) pada setiap lembar jawaban.
4. Lembar soal boleh dibawa pulang setelah ujian selesai. Kumpulkan hanya lembar jawaban.
5. Semua soal dikerjakan dengan Notasi Algoritmik.

Soal 1. LIST REKURSIF (Bobot: 25%)

Diketahui ADT List of Integer dengan first element direpresentasikan secara implisit sebagai berikut:

```

{ List linier direpresentasi dengan pointer }
constant Nil : ...

type infotype : integer
type ElmtList : < Info : infotype, Next : address >
type address : pointer to ElmtList
type List : address { alamat elemen pertama list }

{ *** Selektor *** }
{ Jika L : List dan P : address, First(L) untuk mengakses L (elemen pertama L);
  Next(P) untuk mengakses P^.Next; dan Info(P) untuk mengakses P^.Info }

{*** Predikat pemeriksaan *** }
function IsEmpty (L : List) → boolean
{ Mengirimkan true jika list kosong, false jika tidak kosong }
function IsOneElmt (L : List) → boolean
{ Mengirimkan true jika list hanya berisi satu elemen, false jika tidak }
function IsMember (L : List, X : infotype) → boolean
{ Mengirimkan true jika X merupakan elemen list L, false jika tidak }
function IsDistinct (L : List) → boolean
{ Mengirimkan true jika elemen list unik, false jika tidak unik.
  Asumsi: List tidak kosong } (L ≠ List(X))
function SumAllMulOfX (L : List) → integer
{ Mengirimkan hasil penjumlahan seluruh elemen list yang bernilai kelipatan X, atau
  nol jika list kosong }
procedure AddElmtXX (input/output L : List, input X : integer)
{ I.S. L terdefinisi dan terurut menaik dan mungkin kosong }
{ F.S. X ditambahkan pada L dengan posisi yang sesuai agar terjaga keterurutannya,
  Dan jika pada posisi tersebut sudah ada elemen yang bernilai sama, maka X dikali
  2 dan ditempatkan pada posisi yang sesuai. Dan seterusnya hingga X bernilai unik
  dan dapat menempati posisi yang sesuai. }
```

Tuliskan implementasi (a) fungsi **SumAllMulOfX**, (b) fungsi **IsDistinct**, dan (c) prosedur **AddElmtXX** (tidak perlu menulis ulang spesifikasinya, definisi/header fungsi/prosedur tetap harus ditulis). Seluruh fungsi dan prosedur harus diimplementasikan secara rekursif. Tidak boleh menambahkan prosedur atau fungsi lain.

Soal 2. POHON BINER (Bobot : 25%)

Berikut adalah definisi kamus ADT Binary Search Tree (BST) dengan elemen bilangan integer yang tidak unik; dengan syarat **simpul-simpul di subpohon kiri ≤ akar < simpul-simpul di subpohon kanan**. BST direpresentasikan dengan pointer.

```

{ ADT Binary Search Tree (BST) dengan elemen tidak unik }
{ Pohon Biner dengan representasi pointer }
constant Nil : ... { terdefinisi }

type infotype : integer { merepresentasikan nomor elemen }
type Node : < Info : infotype, Left : BinTree, Right : BinTree >
type address : pointer to Node
type BinTree : address
```

```

{ Selektor : Jika P adalah BinTree: Akar(P) untuk mengakses bagian P^.Info
    Left(P) untuk mengakses bagian P^.Left
    Right(P) untuk mengakses bagian P^.Right }

type Couple : < address, address >

{ Primitif-Primitif }
procedure AddElmt (input X : infotype, input/output P : BinTree)
{ Menambahkan sebuah elemen X ke dalam BST P. Jika ada nilai elemen yang sama, maka
  elemen tersebut ditambahkan sebagai anak kiri }
{ I.S.: BST P asal. Dimungkinkan pohon biner yang kosong
  F.S.: BST P setelah ditambahkan elemen X }
function IsTreeEmpty (P : BinTree) → boolean
{ Mengirimkan true jika P kosong, false jika tidak }
function IsOneElmt (P : BinTree) → boolean
{ Mengirimkan true jika P tidak kosong dan hanya terdiri atas 1 elemen }
function Alokasi (X : infotype) → address
{ mengirimkan address hasil alokasi sebuah elemen pohon dengan X sbg Info, atau nil jika
  alokasi tidak berhasil. }
procedure PrintBST (input P : BinTree)
{ Menampilkan nilai seluruh elemen list pohon biner P ke layar secara terurut membesar}
function CountNodeLevelX (P : BinTree, X : integer) → integer
{ mengirimkan banyaknya node yang berada pada level X.
  Mengembalikan 0 (nol) jika X > tinggi pohon. }

```

Tuliskan implementasi (a) prosedur **AddElmt**, (b) prosedur **PrintBST**, dan (c) fungsi **CountNodeLevelX** (tidak perlu menulis ulang spesifikasinya, definisi/header fungsi/prosedur tetap harus dituliskan). Seluruh fungsi dan prosedur harus diimplementasikan secara **rekursif**. Tidak boleh menambahkan prosedur atau fungsi lain.

Soal 3. Graph dengan Incidence List (Bobot : 25%)

Representasi graph dalam bentuk incidence list adalah sebagai berikut:

```

{ Graph dengan Incidence List }
constant Nil : ...

type adrV : pointer to Vertex
type adrIncEdge : pointer to IncEdge
type adrE : pointer to Edge
type Vertex : < Id : Integer, {identitas simpul}
              FirstInc : adrIncEdge, {pointer ke list incidence edge}
              Next : adrV >
type IncEdge : < PtrEdge: Integer, {identitas simpul}
               Next : adrIncEdge >
type Edge : < Id : Integer, {identitas edge}
             Next : adrEdge >
type Graph : < FirstV : adrV, {First list vertex}
              FirstE : adrE > {First list edge}

{ *** Selektor *** }
{ Jika G : Graph, PV : adrV, PE : adrE, PIE : adrIncEdge, maka:
  FirstV(G) dan FirstE(G) untuk mengakses G.FirstV dan G.FirstE;
  IdV(PV) dan IdE(PE) untuk mengakses PV^.Id dan PE^.Id
  FirstIL(PV) untuk mengakses PV^.FirstInc
  NextV(PV), NextE(PE), dan NextIE(PIE) untuk mengakses PE^.Next, PE^.Next, PIE^.Next }

{ *** Konstruktor *** }
procedure CreateGraph (input idV : Integer, output L : Graph)
{ I.S. idV terdefinisi; F.S. Terbentuk Graph dengan satu simpul dengan Id=idV. }

{ *** Manajemen Memory List *** }
function AlokV (id : Integer) → adrV dan procedure DealokV (input P : adrV)
function AlokE (id : Integer) → adrE dan procedure DealokE (input P : adrE)
function AlokIE (pE : adrE) → adrIncEdge dan procedure DealokIE (input P : adrIncEdge)

{ *** Fungsi/Prosedur Lain *** }
function SearchV (G : Graph, idV : Integer) → adrV
{ Mengembalikan address simpul/vertex dengan Id=idV jika sudah ada pada graph G,
  Nil jika belum }
function SearchE (G : Graph, idE : Integer) → adrE
{ Mengembalikan address busur/edge dengan Id=idE jika sudah ada pada graph G,
  Nil jika belum }

```

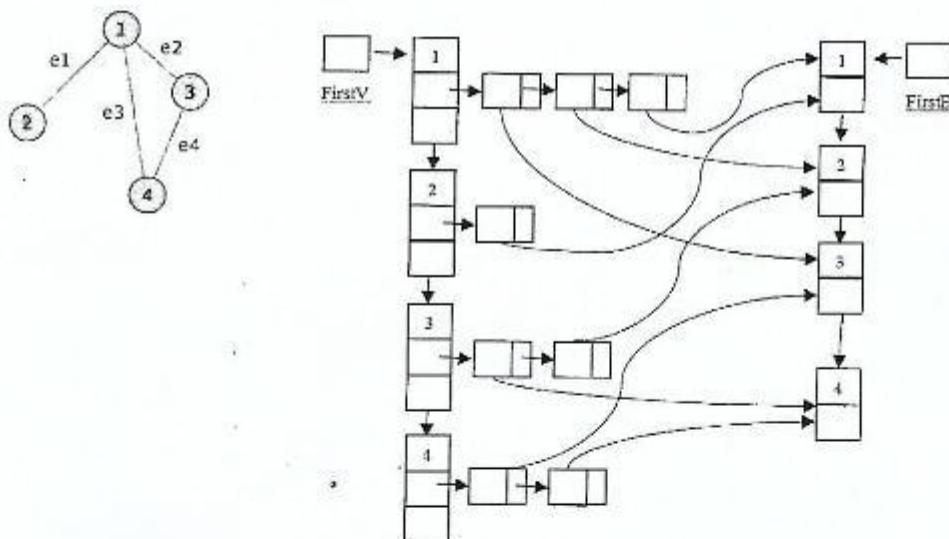
```

function SearchIE (G : Graph, PV: adrV, PE : adrE) → adrIE
{ Mengembalikan address elemen list incidence dari simpul PV yang menunjuk PE jika ada;
Nil jika belum }
procedure InsertV (input/output G : Graph, input idV : Integer, output PV : adrV)
{ Menambahkan simpul idV ke dalam graph, jika alokasi berhasil. }
{ I.S. G terdefinisi, idV terdefinisi dan belum ada pada G. }
{ F.S. Jika alokasi berhasil, idV menjadi elemen pertama G.FirstV,
PV berisi address simpul idV. Jika alokasi gagal, G tetap, PV berisi Nil }
procedure InsertE (input/output G : Graph, input idV1,idV2, idE : Integer)
{ Menambahkan busur idE ke dalam graph, jika alokasi berhasil. }
{ I.S. G terdefinisi, idV1 dan idV2 terdefinisi dan ada pada G;
idE terdefinisi dan belum ada pada G. }
{ F.S. Jika seluruh alokasi berhasil, idE menjadi elemen pertama G.FirstE dan menjadi
busur yang menghubungkan simpul idV1 dan idV2. Jika ada alokasi yg gagal, G tetap}
procedure DeleteE (input/output G : Graph, input idE : Integer)
{ Menghapus busur idE dari graph G }
{ I.S. G terdefinisi, idE terdefinisi dan ada pada G;
F.S. Busur dg Id=idE dihapus dari graph G dan didealokasi
Penghapusan dilakukan di list egde dan di list incidence edge dari simpul2 yang
incidence dengan edge tersebut
Asumsi: busur selalu menghubungkan 2 simpul berbeda }

```

Implementasikan fungsi **SearchV**, prosedur **InsertE**, dan prosedur **DeleteE** sesuai definisi dan spesifikasi di atas (spesifikasi tidak perlu ditulis, definisi/header fungsi/prosedur tetap harus ditulis). Gunakan pendekatan iteratif. Tidak boleh membuat fungsi/prosedur baru.

Berikut adalah ilustrasi graph dan representasinya dalam bentuk Incidence List:



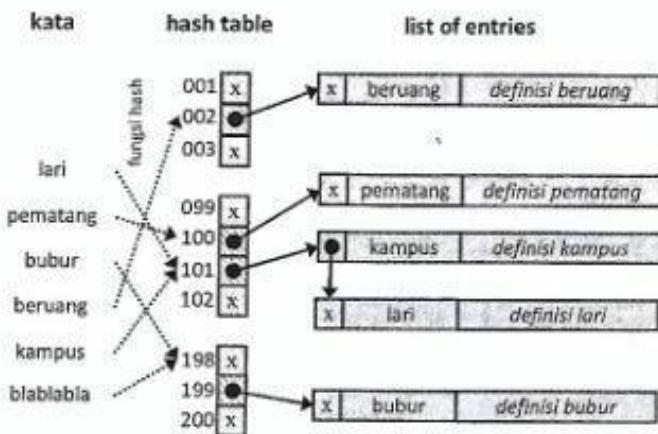
Soal 4. Studi Kasus Kamus/Dictionary Online (Bobot : 25%)

Sebuah aplikasi *dictionary online* mengelola sejumlah data pasangan kata-definisi, dengan kata merupakan kunci atau *key*. Untuk kemudahan, sebuah kata hanya dapat memiliki satu definisi. Operasi yang dapat dilakukan terhadap data *dictionary* adalah:

- TambahKata: menambahkan sebuah entri (pasangan kata-definisi) baru.
- HapusKata: menghapus sebuah entri.
- UbahDefinisi: mengubah definisi untuk sebuah kata.
- DefinisiKata: mengembalikan definisi sebuah kata.

Data *dictionary* disimpan dalam bentuk *map* dengan implementasi menggunakan *hash table*. *Hash table* adalah sebuah tabel *bindings*, yang bersama dengan sebuah fungsi *hash* memetakan setiap nilai *key* ke suatu nilai indeks tabel. Karena jumlah entri data *dictionary* sangat besar, digunakan *hash chaining* untuk menyimpan sebuah *association list* berukuran kecil pada setiap elemen *hash table*.

Ilustrasinya dapat dilihat pada gambar berikut. Misalnya, entri untuk kata lari terdapat pada list dengan *first* di elemen tabel indeks 101 (hasil fungsi *hash* untuk kata lari). Kata blablabla tidak terdefinisi di dalam *dictionary* karena tidak ada entri untuk kata tersebut pada list di indeks tabel 199.



Berikut ini beberapa kamus dan definisi fungsi *hash* yang dapat digunakan.

```
constant MaxBucket : integer = ... {jumlah slot pada hash table}
type keytype : string {tipe untuk menyimpan sebuah key (kata)}
type idxtype : integer [1..MaxBucket] {tipe untuk indeks}
function hashKata (kata: keytype) → idxtype
{Menghasilkan indeks tabel yang seharusnya menampung first dari list yang memuat entri
untuk kata}
```

- Tuliskan definisi dan spesifikasi kamus data (ADT) dan selektor yang sesuai dengan representasi tersebut. Gunakan nama yang sesuai dan mudah dipahami.
Apabila diperlukan, tuliskan definisi dan spesifikasi prosedur/fungsi untuk alokasi/dealokasi.
- Tuliskan definisi, spesifikasi, dan realisasi dari keempat operasi terhadap *dictionary* yang telah disebutkan sebelumnya. Tuliskan definisi, spesifikasi, dan realisasi dari semua fungsi atau prosedur tambahan yang Anda gunakan (jika ada). (*Petunjuk*: bisa dibuat fungsi antara untuk mencari entri berdasarkan kata)

UAS SEMESTER I 2019/2020

PETUNJUK UJIAN:

1. Lembar soal ini terdiri atas 4 halaman yang terdiri atas soal 1 s.d. soal 4. Cek lembar soal, apakah Anda sudah mendapatkan lembar soal yang benar. Jika masih ada kesalahan, tukarkan pada pengawas.
2. Semua soal dikerjakan di lembar jawaban yang disediakan dan jawaban setiap soal diletakkan di lembar yang terpisah dengan soal yang lain.
3. Jawaban ditulis dengan pensil. Bagian identitas tulis dengan bolpoin. Jangan lupa menulis identitas (minimum nama, NIM, Kelas) pada setiap lembar jawaban.
4. Lembar soal boleh dibawa pulang setelah ujian selesai. Kumpulkan hanya lembar jawaban.
5. Semua soal dikerjakan dengan Notasi Algoritmik.

Soal 1. List Rekursif (Bobot: 25%)

Diketahui ADT List of Integer dengan first element direpresentasikan secara implisit sebagai berikut:

```
{ List linier direpresentasi dengan pointer }
constant Nil : ...

type infotype : integer
type ElmtList : < Info : infotype, Next : address >
type address : pointer to ElmtList
type List : address { alamat elemen pertama list }

{ *** Selektor *** }
{ Jika L : List dan P : address, L untuk mengakses elemen pertama L;
  Next(P) untuk mengakses P.Next; dan Info(P) untuk mengakses P.Info }

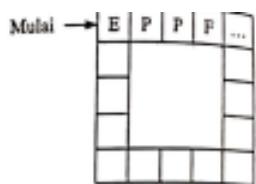
{*** Predikat pemeriksaan ***}
function IsEmpty (L : List) + boolean
{ Mengirimkan true jika list kosong, false jika tidak kosong }
function IsOneElmt (L : List) + boolean
{ Mengirimkan true jika list hanya berisi satu elemen, false jika tidak }
function IsL1AboveL2 (L1 : List, L2 : List) + boolean
{ Mengirimkan true jika untuk setiap nilai elemen ke i dari L1 lebih besar dari nilai elemen ke i dari L2, di mana rentang i adalah dari elemen pertama hingga terakhir, dan mengirimkan false jika tidak. Prekondisi: L1 dan L2 memiliki jumlah elemen yang sama. Untuk L1 dan L2 berupa list kosong, fungsi mengirimkan nilai false.}
procedure DelAboveX (input/output L : List, input X : infotype)
{ I.S. L terdefinisi dan mungkin kosong
  F.S. Semua elemen L yang bernilai di atas X dihapus dari L.}
procedure MinMax (input L : List, output min, max : integer)
{ I.S. L terdefinisi dan tidak kosong
  F.S. min adalah nilai elemen minimum pada list L dan max adalah nilai elemen maksimum pada list L }
```

Tuliskan implementasi a) fungsi IsL1AboveL2, b) procedure DelAboveX, dan c) prosedur MinMax (tidak perlu menulis ulang spesifikasinya, definisi/header fungsi/prosedur tetap harus ditulis). Seluruh fungsi dan prosedur harus diimplementasikan secara rekursif. Boleh menambahkan prosedur atau fungsi lain, dengan catatan prosedur atau fungsi tersebut harus dibuat definisi, spesifikasi dan realisasinya.

Soal 2. Variasi List Linier (Bobot: 25%)

Gionno (dibaca: Jono) ingin membuat program permainan Monopoli. Namun karena ia baru belajar memrogram, ia hanya akan mengimplementasikan sebagian kecil dari fitur permainan Monopoli. Yang jelas, untuk permainan ini ia membutuhkan ADT untuk merepresentasikan papan permainan, serta ADT untuk merepresentasikan pemain-pemain yang ikut bermain.

Papan permainan Monopoli terdiri atas kotak-kotak yang disebut *sel*, yang disusun dalam bentuk melingkar (lihat gambar di samping). Pada Monopoli sederhana versi Gionno ini, sebuah Sel hanya mengandung informasi sebuah karakter, yang menunjukkan aksi yang terjadi pada pemain yang singgah pada Sel tersebut. Karakter "E" artinya pemain mendapatkan \$100, karakter "P" artinya pemain harus membayar \$50, dan karakter "F" artinya pemain tidak membayar atau mendapatkan apa-apa. Papan Permainan direpresentasikan sebagai list of Sel, dengan jumlah Sel yang tertentu.



Setiap Pemain memiliki ID berupa integer, Saldo yang juga berupa integer, serta indikator di Sel mana ia sedang berada. Satu sesi permainan dapat diikuti oleh 2 hingga 4 orang pemain, dan setiap pemain bergiliran melemparkan dadu untuk kemudian maju sebanyak X Sel dengan X adalah angka hasil lemparan dadu. Untuk itu diperlukan sebuah list of Pemain yang dapat langsung menunjukkan urutan giliran melempar dadu.

Program utama yang Gionno buat kira-kira sebagai berikut:

```

PROGRAM Monopoli
KAMUS
papan: ...
list_pemain: ...
current_pemain, pemain: ...
i, jumlah_pemain, X: integer
game_over: boolean
ALGORITMA
CreateBoard(papan,16) { Membuat papan berisi 16 Sel, setiap Sel sudah berisi karakter yg valid }
CreateEmpty(list_pemain) { Membuat list of pemain, masih kosong }
input(jumlah_pemain) { Diasumsikan pasti 2 ≤ jumlah_pemain ≤ 4 }
i traversal [1..jumlah_pemain]
    pemain + Alokasi(i, 100, First(papan)) { Buat pemain dengan ID i, Saldo awal 100,
                                                dan posisi di Sel pertama papan }
InsertSorted(list_pemain, pemain) { Tambahkan ke list pemain, terurut berdasarkan ID }
current_pemain + First(list_pemain) { Pemain yang mendapat giliran pertama }
game_over + False { Penanda game over }
while not game_over do
    X + lempar_dadu() { X berisi hasil lemparan dadu }
    MoveForward(current_pemain, X) { current_pemain maju sebanyak X Sel, dan aksi yang
                                    terkandung pada Sel tersebut diterapkan pada current_pemain }
    game_over + check_game_over_condition(papan,list_pemain) { fungsi dianggap sudah terdefinisi }
    current_pemain + Next(current_pemain) { Giliran pemain berikutnya }

```

Bantulah Gionno membuat program Monopoli dengan menjawab hal-hal berikut:

- Variasi list linier apa yang cocok untuk merepresentasikan Papan Permainan?
- Variasi list linier apa yang cocok untuk merepresentasikan list of Pemain?
- Gambarkan diagram yang mengilustrasikan struktur berkait kedua list tersebut, lengkap dengan informasi apa saja yang terkandung dalam setiap elemen list. Sebutkan juga selektor-selektornya agar dapat dirujuk di jawaban soal berikutnya:
- Implementasikan prosedur-prosedur berikut tanpa menulis ulang spesifikasinya, dan tanpa menggunakan prosedur/fungsi antara.

```

procedure InsertSorted(input/output list_pemain:..., input pemain:...)
{ I.S. list_pemain terdefinisi dan mungkin kosong, pemain adalah address yang valid ke
  sebuah Pemain.
  F.S. pemain menjadi anggota list_pemain dengan posisi terurut menaik berdasarkan ID. }
procedure MoveForward(input/output pemain:..., input x: integer)
{ I.S. pemain dan x terdefinisi, pemain berada pada suatu Sel di Papan Permainan.
  F.S. pemain pindah ke Sel yang berada pada x langkah dari posisi semula, aksi yang
  ditunjukkan oleh karakter pada Sel tujuan diterapkan pada pemain. }

```

Soal 3. Pohon Biner (Bobot : 25%)

1	C3PO	66
2	Chewbacca	234
3	Finn	23
4	Han Solo	63
5	Kylo Ren	29
6	Lando	65
7	Princess Leia	53
8	Luke Skywalker	53
9	Rey	19
10	Baby Yoda	50

Tabel di atas menyatakan usia dari beberapa karakter dalam film Star Wars. Data tersebut ingin disimpan dalam sebuah struktur data Binary Search Tree (BST). Penyimpanan data tersebut harus sesuai dengan aturan pohon biner BST, yaitu:

- semua simpul pada subpohon kiri $<$ Akar(P)
- semua simpul pada subpohon kanan \geq Akar(P)

Dalam kasus ini, keterurutan ditentukan oleh data umur. Tugas Anda adalah:

- a) Membuat ADT terkait BST untuk data tersebut dan gambarkan ilustrasi struktur data dalam bentuk diagram. Pastikan semua data yang terkandung dalam ADT yang Anda rancang tergambar dalam diagram dan diberi label secara jelas.
- b) Membuat prosedur untuk menambahkan data baru (karakter dan umurnya) ke dalam pohon biner BST sehingga menempati posisi yang tepat.
- c) Menggambarkan pohon BST yang terbentuk dari data pada tabel di atas.

Soal 4. Studi kasus Topological Sort (Bobot: 25%)

Berikut ini adalah definisi ADT Graph dalam bentuk Adjacency List yang digunakan untuk merepresentasikan graf berarah:

```
{ Graph berarah diimplementasi sebagai Adjacency list }
constant Nil: ...
type adrNode : pointer to Node
type adrAdjNode : pointer to AdjNode
type Node : < Id: integer, {identitas simpul}
           InDegree: integer, {jumlah busur yang masuk ke simpul}
           Printed: boolean,
           LAdj: adrAdjNode, {pointer ke list adjacent (simpul adjacent)}
           Next: adrNode >
type AdjNode : < Id: integer, {identitas simpul adjacent}
               Next: adrAdjNode >
type Graph : < First: adrNode >

{ *** Selektor *** }
{ Jika G: Graph, Pn: adrNode, Pa: adrAdjNode, maka:
  First(G) = G.First;
  Id(Pn) = Pn.Id; InDegree(Pn) = Pn.InDegree; Printed(Pn) = Pn.Printed
  LAdj(Pn) = Pn.LAdj; Next(Pn) = Pn.Next;
  Id(Pa) = Pa.Id; Next(Pa) = Pa.Next }

{ *** Konstruktor *** }
procedure CreateGraph (input X: integer, output L: Graph)
{ I.S. Sembarang; F.S. Terbentuk Graph dengan satu simpul dengan Id=X }

{ *** Manajemen Memory List Simpul *** }
function AlokNode (X: integer) → adrNode
```

```
{ Mengembalikan address hasil alokasi Simpul X. }
procedure DealokNode (input P: adrNode)
{ I.S. P terdefinisi; F.S. P dikembalikan ke sistem }

{ *** Manajemen Memory List Adjacent *** }
function AlokAdjNode (X: integer) → adrAdjNode
{ Mengembalikan address hasil alokasi. }
procedure DealokAdjNode (input P: adrAdjNode)
{ I.S. P terdefinisi; F.S. P dikembalikan ke sistem }

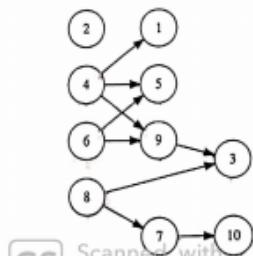
{ *** Fungsi/Prosedur Lain *** }
function SearchNode (G: Graph, X: integer) → adrNode
{ mengembalikan address simpul dengan Id=X jika sudah ada pada graph G, Nil jika belum }
procedure InsertNode (input/output G: Graph, input X: integer, output Pn: adrNode)
{ Menambahkan simpul X ke dalam graph, jika alokasi X berhasil. }
procedure InsertEdge (input/output G: Graph, input prec, succ: integer)
{ Menambahkan busur dari prec menuju succ ke dalam G }
procedure DeleteNode (input/output G: Graph, input X: integer)
{ Menghapus simpul X dari G }
function HasAdjacent (input G: Graph, P: adrNode) → adrAdjNode
{ mengembalikan address simpul adjacent jika ada, Nil jika tidak ada }
function CountNode (input G: Graph) → integer
{ mengembalikan jumlah simpul yang ada pada graph G, 0 jika graf masih kosong }

Procedure TopoSort(input/output G: Graph)
{Mengurutkan simpul berdasarkan keterurutan linier}
{ I.S. G terdefinisi }
{ F.S. menampilkan linear order simpul dalam graf G }
```

Tugas Anda adalah:

- Menggambarkan ilustrasi struktur data dalam bentuk diagram. Pastikan semua data yang terkandung dalam ADT tergambar dalam diagram dan diberi label secara jelas
- Membuat realisasi fungsi SearchNode dan HasAdjacent tanpa menulis ulang spesifikasinya
- Membuat realisasi procedure TopoSort yang melakukan topological sort dengan cara seperti berikut ini (dengan asumsi graf sudah terdefinisi), dan dengan memanfaatkan primitif yang ada pada ADT:
 - Selama masih ada node yang in-degree nya ≠ 0:
 - Cari node dengan in-degree = 0, jika ada tampilkan dan beri tanda sudah ditampilkan
 - Jika node tersebut memiliki adjacent node, maka
 - In-degree dari setiap node yang adjacent dengan node tersebut dikurangi dengan satu
 - jika in-degree mereka pun = 0 maka tampilkan dan beri tanda sudah ditampilkan

Sebagai ilustrasi, pada contoh graf berarah di bawah ini, jika diterapkan algoritma topological sort maka linear order yang dihasilkan adalah 2→4→1→6→5→9→8→3→7→10.



UAS Algoritma Struktur Data 2020/2021

10 Desember 2020

Petunjuk

- Waktu: 150 menit
- Boleh membuka diktat Notasi Algoritmik
- Tuliskan jawaban dalam file teks pper nomor soal, dengan nama file <NIM> <NomorSoal>.txt. Misalnya 13506070_1.txt untuk soal 1. File dengan nama yang tidak sesuai tidak dijamin akan dinilai.
- Untuk menuliskan karakter spesial, dapat digunakan *shortcut*田; (di Windows), ^⌘Space (di Mac), atau silakan salin & tempel dari sini: → ← ≠ ≤ ≥

Soal 1. List Rekursif (Bobot: 25%)

Diketahui ADT List of Integer dengan first element direpresentasikan secara implisit sebagai berikut:

```
{ List linier direpresentasikan dengan pointer }
constant Nil: ...

type infotype: integer
type ElmtList: < Info: infotype, Next: address >
type address: pointer to ElmtList
type List: address { alamat elemen pertama list }

{ *** Selektor *** }
{ Jika L: List dan P: address, L untuk mengakses elemen pertama L;
  Next(P) untuk mengakses P↑.Next; dan Info(P) untuk mengakses P↑.Info }

{ *** Predikat pemeriksaan *** }
function IsEmpty (L: List) → boolean
{ Mengirimkan true jika list kosong, false jika tidak kosong }
function IsOneElmt (L: List) → boolean
{ Mengirimkan true jika list hanya berisi satu elemen, false jika tidak }
function IsMultiplied (L1, L2: List, N: integer) → boolean
{ I.S. List L1, L2 terdefinisi, memiliki jumlah elemen yang sama dan mungkin kosong.
  N terdefinisi.
  F.S. Mengembalikan nilai true jika semua elemen L1 dikalikan dengan N, hasilnya
    akan sama dengan L2, dan mengembalikan false jika tidak. Contoh: L1 = 2->1->4,
    L2 = 6->3->12 dan N = 3, akan mengembalikan nilai true. Untuk kasus list
    kosong, akan mengembalikan true.}

procedure PrintAlternate (input L: List, input isOdd: boolean)
{ I.S. List L terdefinisi dan mungkin kosong, isOdd terdefinisi.
  F.S. Setiap elemen urutan ganjil dari list L tercetak jika isOdd bernilai true,
```

atau elemen urutan genap akan tercetak jika `isOdd` bernilai `false`. Sebagai contoh, jika list `L` berisi nilai `(2,4,3,2,8)` dan `isOdd = true`, maka yang tercetak adalah `2, 3` dan `8`. Begitu pula sebaliknya, jika `isOdd = false`, yang tercetak adalah `4` dan `2`.

```
procedure DeleteKth (input/output L: List, input K: integer)
{ I.S. L terdefinisi dan mungkin kosong. K berupa integer dengan rentang nilai 1 hingga banyaknya elemen dari list L untuk L tidak kosong, dan K bisa terdefinisi sembarang jika L kosong.
 F.S. Elemen ke-K dari list L terhapus list L.}
```

Tuliskan implementasi (a) fungsi **IsMultiplied**, (b) prosedur **PrintAlternate** dan (c) prosedur **DeleteKth** (tidak perlu menulis ulang spesifikasinya). Seluruh fungsi dan prosedur harus diimplementasikan secara **REKURSIF**. Tidak diperkenankan untuk membuat fungsi/prosedur lain.

Soal 2. Pohon Biner (Bobot: 25%)

Berikut adalah defutisi kamus ADT Binaly Search Tree (BST) dengan elemen berupa pasangan **Value** dan **Count**. Value berupa nilai integer yang **unik** dan **Count** yang menyatakan **banyaknya kemunculan Value** tersebut. Sesuai aturan BST, semua value pada subpohon kiri P selalu lebih kecil dari `Value(P)`, dan value pada subpohon kanan P selalu lebih besar dari `Value(P)`. BST direpresentasikan dengan pointer.

```
{ ADT Pohon Biner Terurut (BST) dengan elemen unik }
{ Pohon Biner dengan representasi pointer }
constant Nil: ... ( terdefinisi )
type Infotype: <Value: integer, Count: integertype Node: <Info: Infotype, Left: BinTree, Right: BinTree>
type adrNode: pointer to Node
type BinTree: adrNode
{ Selektor: Jika P adalah BinTree:
    Value(P) untuk mengakses bagian P↑.Info.Value;
    Count(P) untuk mengakses bagian P↑.Info.Count;
    Left(P) untuk mengakses bagian P↑.Left;
    Right(P) untuk mengakses bagian P↑.Right }

{*** Konstruktor ***}
function MakeTree (X: integer) → BinTree
{ Mengirimkan sebuah pohon biner, misalnya P. Value(P)=X, Count(P)=1,
  Left(P)=Right(P)=Nil. }
{ Jika alokasi gagal, menghasilkan pohon kosong }

{*** Predikat ***}
function IsTreeEmpty (P: BinTree) → boolean
{ Mengirimkan true jika P kosong, false jika tidak }
function IsTreeOneElmt (P: BinTree) → boolean
{ Mengirimkan true jika P tidak kosong dan hanya terdiri atas 1 elemen }
function IsUnderLeft (P: BinTree) → boolean
{ Mengirimkan true jika P hanya memiliki subpohon kiri. Prekondisi: P tidak kosong }
function IsUnderRight (P: BinTree) → boolean
{ Mengirimkan true jika P hanya memiliki subpohon kanan. Prekondisi: P tidak kosong }
function IsBiner (P: BinTree) → boolean
{ Mengirimkan true jika P memiliki subpohon kiri dan kanan. Prekondisi: P tidak kosong }
```

```

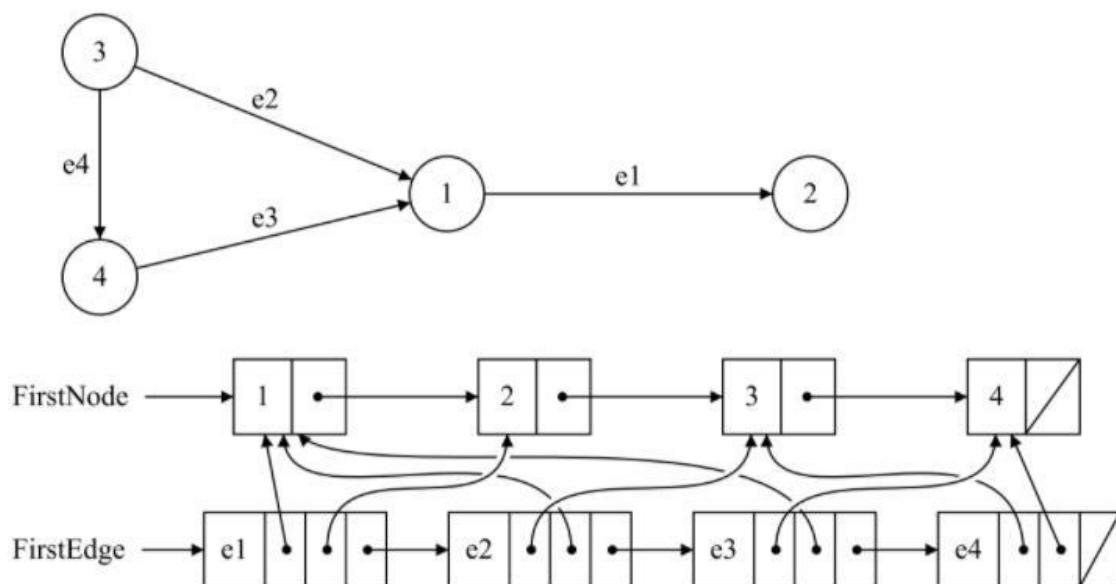
{*** Primitif lain ***}
function SearchLessThen (P: BinTree, X: integer) → boolean
{ Mengirimkan true jika ada Value pada pohon P yang nilainya kurang dari X; false
jika tidak ada }
function SumAllValue (P: BinTree) → integer
{ Mengirimkan jumlah total seluruh Value pada pohon P; jika Count(pt) > 1 maka
dihitung sebesar Value (pt) dikali Count(pt) }
function MaxOccurence (P: BinTree) → infotype
{ Mengirimkan alamat node pada pohon P yang Value-nya muncul paling banyak (Count-nya
terbanyak); jika ada lebih dari satu node yang dengan Count tertinggi, kirimkan
alamat node dengan Value tertinggi; pohon tidak kosong }

```

Tuliskan implementasi (a) fungsi **SearchLessThen**, (b) fungsi **SumAllValue** dan (c) fungsi **MaxOccurence** (tidak perlu menulis ulang spesifikasinya, definisi/header fungsi/prosedur tetap harus ditulis). Seluruh fungsi dan prosedur harus diimplementasikan secara **REKURSIF**. Tidak boleh menambahkan prosedur atau fungsi lain, kecuali prosedur atau fungsi yang diperlukan untuk menngani kasus khusus (misalnya pohon kosong pada persoalan basis 1).

Soal 3. Graf berarah dengan Edge List (Bobot: 25%)

Graf berarah direpresentasikan dengan dua buah list unik, yaitu list yang berisi semua node yang ada dalam graf, dan list yang berisi semua edge yang ada di dalam graf.



Ilustrasi Graf dengan Edge List

```

constant Nil: ...
type addrNode: pointer to Node
type addrEdge: pointer to Edge
type Node: < Id: integer,           { identitas Node }
          Next: addrNode >      { alamat Node berikutnya }
type Edge: < Source: addrNode, Target: addrNode, { alamat Node asal & tujuan }
          Next: addrEdge > { alamat Edge berikutnya }
type Graph: < FirstNode: addrNode, { alamat Node pertama }
          FirstEdge: addrEdge > { alamat Edge pertama }
{ Selektor untuk Graph G, Node N, dan Edge E:
- Id(N) dan Next(N) adalah N.Id dan N.Next

```

- Source(E), Target(E), dan Next(E) adalah E.Source, E.Target, dan E.Next
- FirstNode(G) adalah G.FirstNode
- FirstEdge(G) adalah G.FirstEdge }

{*** Konstruktor ***}

procedure CreateGraph(input X: integer, output G: Graph)

{ I.S. Sembarang;

F.S. Terbentuk Graph dengan satu node dengan Id=X. }

{*** Manajemen memori ***}

function AlokNode(id: integer) → addrNode

procedure DealokNode(input N: addrNode)

function AlokEdge(s, t: addrNode) → addrEdge

procedure DealokEdge(input E: addrEdge)

{*** Fungsi/prosedur lain ***}

function SearchNode(G: Graph, X: integer) → addrNode

{ Mengembalikan address node dengan Id=X jika sudah ada pada graph G, Nil jika belum. }

function SearchEdge(G: Graph, s, t: integer) → addrEdge

{ Mengembalikan address edge yang menyimpan info edge dari node ber-id s ke node ber-id t jika sudah ada pada Graph G, Nil jika belum. Perhatikan: s & t adalah id Node, bukan address Node. }

procedure InsertNode(input/output G: Graph,

input X: integer,

output Pn: addrNode)

{ Menambahkan node X ke dalam G jika alokasi berhasil. }

{ I.S. G terdefinisi, X terdefinisi dan belum ada pada G. }

{ F.S. Jika alokasi berhasil, X menjadi elemen pada G, Pn berisi address node X. Jika alokasi gagal, G tetap, Pn berisi Nil. }

procedure InsertEdge(input/output G: Graph, input s, t: integer)

{ Menambahkan edge dari s menuju t ke dalam G }

{ I.S. G, s, dan t terdefinisi. }

{ F.S. Jika belum ada edge (s,t) di G, maka tambahkan edge (s,t) ke G. Jika node s atau t belum ada pada G, tambahkan node tersebut dahulu. Jika sudah ada edge (s,t) di G, maka G tetap. Perhatikan: s & t adalah id Node, bukan address Node. }

function CountInboundEdges(G: Graph, X: integer) → integer

{ Mengembalikan jumlah edge yang menuju node ber-id X. }

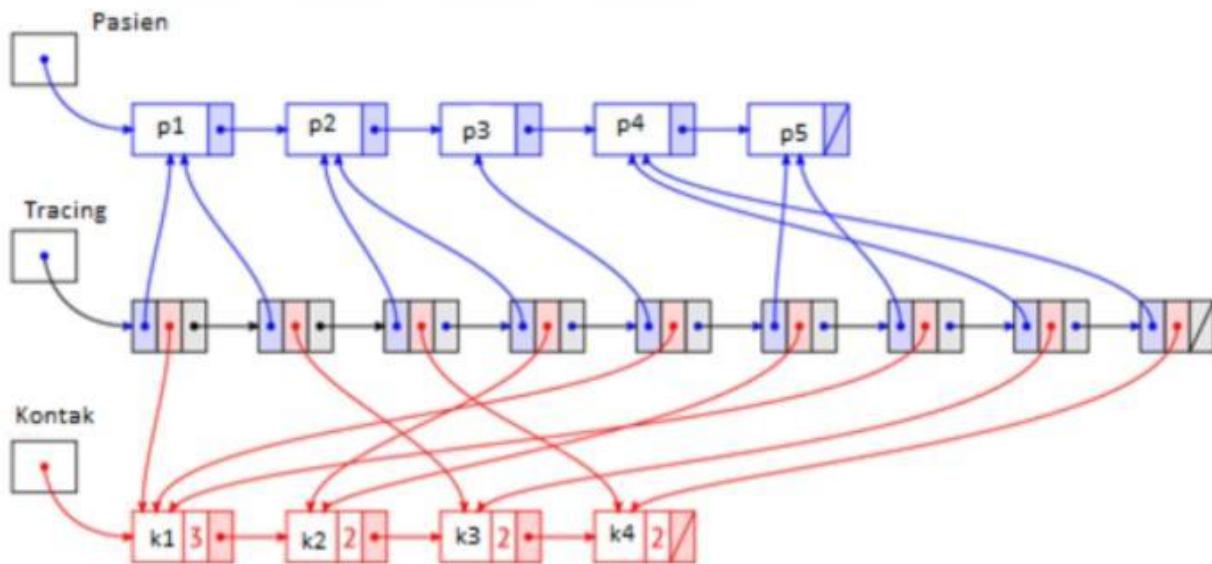
function CountOutboundEdges(G: Graph, X: integer) → integer

{ Mengembalikan jumlah edge yang berasal dari node ber-id X. }

Buatlah implementasi fungsi (a) **SearchEdge**, prosedur (b) **InsertEdge**, dan fungsi (c) **CountInboundEdges** sesuai spesifikasi. Boleh menggunakan fungsi/prosedur yang ada sesuai urutan deklarasi (misal: **SearchEdge** boleh menggunakan **SearchNode** tapi tidak boleh menggunakan **InsertNode**), tidak boleh membuat/mengasumsikan fungsi/prosedur tambahan.

Soal 4. Studi Kasus Relasi M-N

Di kota Asgard dilakukan *tracing* atau penelusuran terhadap orang-orang yang sudah pernah kontak dengan pasien positif COVID-19. Setiap pasien COVID-19 tercatat bertemu dengan beberapa orang yang belum positif COVID-19. Setiap orang yang belum positif COVID-19 tercatat bisa saja pernah kontak dengan lebih dari satu pasien COVID-19. Seorang peneliti hendak mencari tahu bagaimana perilaku penyebaran virus COVID-19 tersebut di kota Asgard. Untuk itu, ia memodelkan riwayat kontak tersebut dalam relasi M-N sebagai berikut:



KAMUS

```

type AdrPasien: { type terdefinisi, alamat sebuah elemen list pasien }
type AdrKontak: { type terdefinisi, alamat sebuah elemen list kontak }
type AdrTracing: { type terdefinisi, alamat sebuah elemen list tracing }
type Pasien: < Nama: string, NextPasien: AdrPasien >
type Kontak: < NamaKontak: string, JumlahKontakPasienCovid: integer
           NextKontak: AdrKontak >
type Tracing: < PPasien: AdrPasien, PKontak: AdrKontak,
                  NextTracing: AdrTracing >
                  { PPasien adalah tracing kontak PKontak }
{ Anggap setiap atribut 3 tipe di atas dapat diakses dengan selektor sesuai nama
atributnya, Misalnya Nama(PP) untuk atribut Nama dari pasien PP. Elemen list dapat diciptakan dengan fungsi bernama Alokasi(), misal PP ← Alokasi("Thanos")}

type ListPasien: AdrPasien
type ListKontak: AdrKontak
type ListTracing: AdrTracing

{*** Fungsi/Prosedur ***}
function AlokPasien(nama: string) → AdrPasien
{ Mengalokasikan elemen list Pasien bertipe AdrPasien, jika gagal akan mengirimkan
Nil }
function AlokTracing(P1: AdrPasien, K1: AdrKontak) → AdrTracing
{ Mengalokasikan elemen list Tracing bertipe AdrTracing, jika gagal akan mengirimkan
Nil }
function AlokKontak(namaKontak: string, jumlah: integer) → AdrKontak
  
```

```

{ Mengalokasikan elemen list Kontak bertipe AdrKontak, jika gagal akan mengirimkan
Nil }

procedure AddTracing(input FirstPasien: ListPasien,
                     input/output FirstKontak: ListKontak,
                     input/output FirstTracing: ListTracing
                     input PP: AdrPasien, input PK: AdrKontak)
{ Menambah tracing baru yg menyatakan adanya kontak antar pasien PP dengan kontak PK.
I.S. PP dan PK masing-masing ada di ListPasien dan ListKontak, PP bukan orang
yang sama dengan PK
F.S. Jika alokasi berhasil, sebuah tracing baru yang menyatakan adanya kontak
antara PP dan PR ditambahkan; jumlah kontak pasien covid PK disesuaikan. Jika
alokasi tidak berhasil, tidak ada perubahan pada list tracing.}

function Epidemic (FirstPasien: ListPasien, FirstKontak: ListKontak,
                  FirstTracing: ListTracing, Ps: string) → real
{ Menghitung berapa jumlah kontak yang pernah ditemui seorang pasien bernama Ps,
dibandingkan dengan total jumlah kontak di kota Asgard. Ps ada pada list Pasien.
Misalkan jumlah kontak yang pernah ditemui Ps 5, total jumlah di kota Asgard 36,
sehingga nilai yang dikirimkan adalah 5/36 atau 0.14. }

function IsCommon(FirstPasien: ListPasien, FirstKontak: ListKontak,
                  FirstTracing: ListTracing, P1: string, P2: string) → boolean
{ Mencari tahu apakah pasien bernama P1 dan P2 memiliki minimal 1 kontak yang sama. }

```

Implementasikan fungsi/prosedur (a) **AddTracing**, (b) **Epidemic**, dan (c) **IsCommon** yang ada sesuai definisi dan spesifikasi di atas, tanpa menulis ulang spesifikasinya.

UAS Algoritma Struktur Data 2021/2022

LIST REKURSIF

1. Diketahui ADT List of Integer dengan first element direpresentasikan secara implisit sebagai berikut:

```

{ List linier direpresentasi dengan pointer }
constant Nil : ...
type infotype : integer
type ElmtList : < Info : infotype, Next : address >
type address : pointer to ElmtList
type List : address { alamat elemen pertama list }
{ *** Selektor *** }
{ Jika L : List dan P : address, L untuk mengakses elemen pertama L;
  Next(P) untuk mengakses P^.Next; dan Info(P) untuk mengakses P^.Info }
{*** Predikat pemeriksaan ***}
function isEmpty (L : List) → boolean
{ Mengirimkan true jika list kosong, false jika tidak kosong }
function isOneElmt (L : List) → boolean
{ Mengirimkan true jika list hanya berisi satu elemen, false jika tidak }
{*** Manajemen Memori ***}
function newNode(x : infotype) → Address
{ Mengirimkan address hasil alokasi sebuah elemen }
{ Jika alokasi berhasil, maka address tidak NIL, dan misalnya
  menghasilkan p, maka Info(p)=x, Next(p)=NIL }
{ Jika alokasi gagal, mengirimkan NIL }
procedure dealokasi(input/output addr : Address)
{ I.S. addr tidak Nil }
{ F.S. addr sudah dibebaskan dari memori }
function isKelipatanX (L : List, x : integer) → List
{ menerima list L yg tidak kosong dan x integer, mengembalikan true jika seluruh elemen
  list L kelipatan x, false jika tidak }
function copyList(L : List) → List
{ Menerima list L yang mungkin kosong, mengembalikan list baru dengan elemen yang sama
  persis dengan elemen list L }
function sortedMerge (L1, L2 : List) → List
{ Menerima list L1 dan L2 yang terurut membesar dan mungkin kosong
  Mengembalikan list yang merupakan gabungan dari L1 & L2 dan tetap terurut membesar
  Asumsi: alokasi selalu berhasil }
procedure deleteAlternate (input/output L : List, input isOdd : boolean)
{ I.S. List L terdefinisi dan tidak kosong. isOdd terdefinisi true atau false
  F.S. Menghasilkan list L yang telah dihapus seluruh elemen pada urutan ganjil jika
  isOdd=true, atau telah dihapus seluruh elemen pada urutan genap jika isOdd=false }

```

Tuliskan implementasi (a) fungsi isKelipatanX, (b) fungsi sortedMerge dan (c) prosedur deleteAlternate (tidak perlu menulis ulang spesifikasinya). Seluruh fungsi dan prosedur harus diimplementasikan secara rekursif.

Tidak diperkenankan untuk membuat fungsi/prosedur lain, tapi boleh memanfaatkan fungsi/prosedur yang sudah ada.

POHON BINER

2. Kesultanan Andara mengadakan kompetisi sapi berdasarkan beratnya. Data yang diperoleh kemudian disimpan dalam struktur data binary search tree (BST). Berikut adalah definisi kamus ADT Cow BST dengan elemen berupa pasangan Weight dan Count. Weight berupa nilai integer yang unik dan Count yang menyatakan banyaknya kemunculan Weight tersebut. Sesuai aturan BST, semua weight pada subpohon kiri P selalu lebih kecil dari Weight(P), dan weight pada subpohon kanan P selalu lebih besar dari Weight(P). BST direpresentasikan dengan pointer.

```

{ ADT Cow BST dengan elemen unik }
{ Pohon Biner dengan representasi pointer }
constant Nil : ... { terdefinisi }
type Infotype: <Name: string, Weight: integer, Count integer>
type Node : <Info : Infotype, Left : BinTree, Right : BinTree >
type adrNode : pointer to Node
type BinTree : adrNode

{ Selektor : Jika P adalah BinTree:
  Weight(P) untuk mengakses bagian P^.Info.Weight;
  Count(P) untuk mengakses bagian P^.Info.Count;
  Left(P) untuk mengakses bagian P^.Left;
  Right(P) untuk mengakses bagian P^.Right }

{*** Konstruktor ***}
function MakeTree (X : integer) → BinTree
{ Mengirimkan sebuah pohon biner, misalnya P. Weight(P)=X, Count(P)=1,
Left(P)=Right(P)=Nil. }
{ Jika alokasi gagal, menghasilkan pohon kosong }

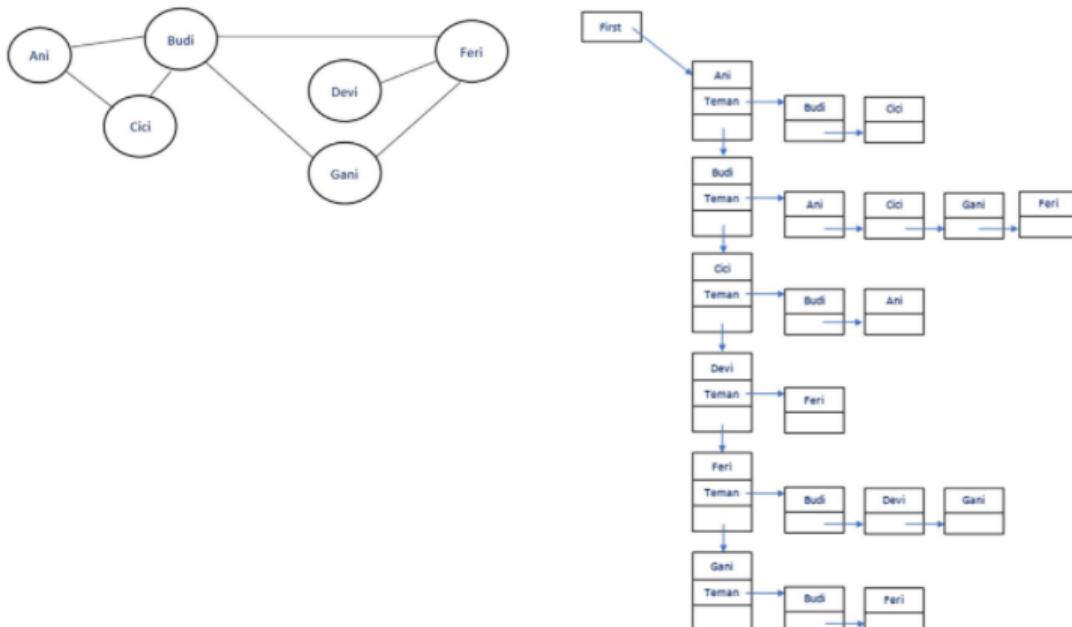
{*** Predikat ***}
function isEmpty (P : BinTree) → boolean
{ Mengirimkan true jika P kosong, false jika tidak }
function isTreeOneElmt (P : BinTree) → boolean
{ Mengirimkan true jika P tidak kosong dan hanya terdiri atas 1 elemen }
function isUnderLeft (P : BinTree) → boolean
{ Mengirimkan true jika P hanya memiliki subpohon kiri. Prekondisi: P tidak kosong }
function isUnderRight (P : BinTree) → boolean
{ Mengirimkan true jika P hanya memiliki subpohon kanan. Prekondisi: P tidak kosong }
function isBiner (P : BinTree) → boolean
{ Mengirimkan true jika P memiliki subpohon kiri dan kanan. Prekondisi: P tidak kosong }
{*** Primitif Lain ***}
function sumAllWeight(P : BinTree) → integer
{ Mengirimkan jumlah total seluruh Weight pada pohon P; jika Count(pt) > 1 maka dihitung
sebesar Weight(pt) dikali Count(pt) }
function maxOccurrence(P : BinTree) → BinTree
{ Mengirimkan alamat node pada pohon P yang Weight-nya muncul paling banyak (Count-nya
terbanyak); jika ada lebih dari satu node yang dengan Count tertinggi, kirimkan alamat
node dengan Weight tertinggi; pohon tidak kosong }

```

Tuliskan implementasi (a) fungsi sumAllWeight dan (b) fungsi maxOccurrence (tidak perlu menulis ulang spesifikasinya, definisi/header fungsi/prosedur tetap harus dituliskan). Seluruh fungsi dan prosedur harus diimplementasikan secara rekursif. Tidak boleh menambahkan prosedur atau fungsi lain, kecuali prosedur atau fungsi yang diperlukan untuk menangani kasus khusus (misalnya pohon kosong pada persoalan basis 1).

GRAF PERTEMANAN

3. Pertemanan sekompok orang dapat digambarkan dalam bentuk graf tidak berarah, dengan simpul merepresentasikan seseorang dan busur merepresentasikan adanya hubungan pertemanan antara dua orang. Graf direpresentasikan dengan adjacency list seperti pada gambar di bawah.



Berikut adalah definisi dan spesifikasi graf pertemanan:

```

constant Nil: ...
type adrOrang: pointer to Orang
type adrTeman: pointer to Teman

type Orang: < Nama : string, { identitas Node }
           Teman : adrTeman,
           Next : adrOrang > { alamat Node berikutnya }
type Teman: < Nama : string, { alamat Node asal & tujuan }
           Next : adrTeman > { alamat Edge berikutnya }
type Graph: < First : adrOrang > { alamat Node pertama }

{ Selektor untuk Graph G, Orang O, dan Teman T:
  - Nama(O), Teman(O) dan Next(O) adalah O.Nama, O.Teman dan O.Next
  - Nama(T) dan Next(T) adalah T.Nama dan T.Next
  - First(G) adalah G.First }

{*** Konstruktor ***}

procedure CreateGraph(input NamaX: string, output G: Graph)
{ I.S. Sembarang;
  F.S. Terbentuk Graph untuk satu orang dengan Nama=NamaX tanpa punya teman }

{*** Manajemen memori ***}
function newOrang (NamaX: string) → adrOrang
procedure dealokOrang (input/output O: adrOrang)
function newTeman (NamaX : string) → adrTeman
procedure dealokTeman (input/output T: adrTeman)

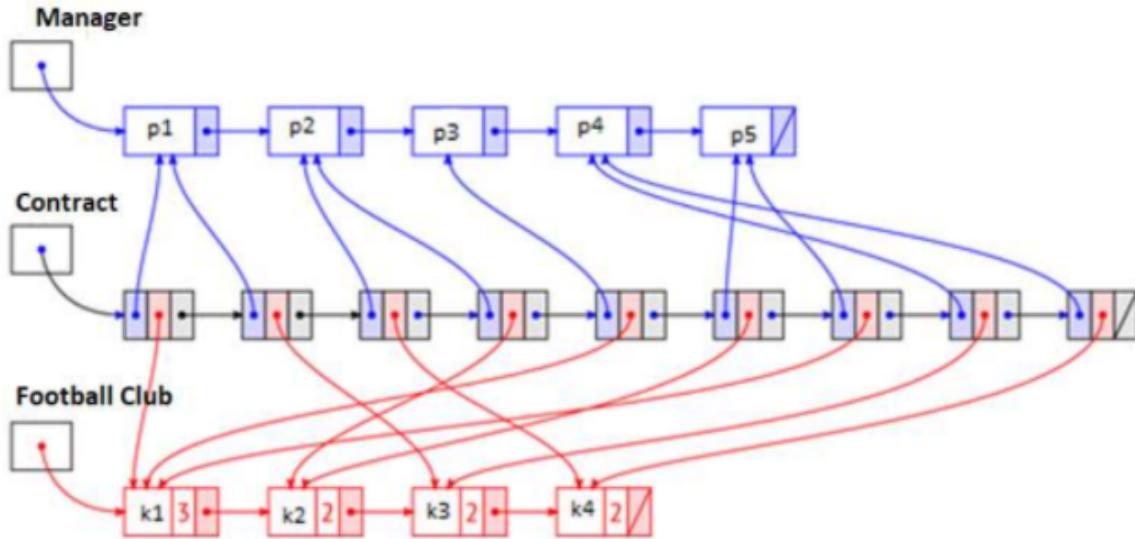
{*** Fungsi/prosedur lain ***}
function cariOrang (G: Graph, siX: string) → adrOrang
{ Mengembalikan alamat orang dengan nama=siX ada pada G, nil jika tidak ada}
procedure tambahOrang (input/output G: Graph,
                       input siX: string,
                       output po: adrOrang)
{ I.S. G, siX terdefinisi, siX belum ada pada G }
{ F.S. Jika alokasi berhasil, siX ditambahkan pada G dengan alamat po }
procedure tambahTeman (input/output G: Graph,
                       input siA, siB: string)
{ I.S. G, siA dan siB terdefinisi; siA dan siB mungkin belum ada pada G
  siA dan siB belum berteman }
{ F.S. Jika alokasi berhasil, ada pertemanan antara siA dan siB pada G
  Jika ada alokasi gagal siA sbg teman siB dan siB sbg teman siA harus dihapus agar
  G tetap konsisten; tapi siA dan/atau siB sbg orang tidak perlu dihapus dari G }
function palingGaul (G: Graph) → string
{ Mengembalikan nama orang yang paling banyak temannya;
  atau "--" jika semua orang tidak punya teman }
function apakahBerteman(G: Graph, siX: string) → boolean
{ Mengembalikan true jika siX ada pada G dan berteman dengan orang paling gaul,
  false jika tidak }

```

Buatlah implementasi (a) prosedur tambahTeman dan (b) fungsi apakahBerteman tanpa membuat/mengasumsikan fungsi/prosedur tambahan. (c) Menurut Anda, apakah ada representasi graf yang lebih tepat untuk kasus ini? Jelaskan alasannya.

STUDI KASUS RELASI M-N

4. Kompetisi sepakbola liga Inggris merupakan kompetisi yang paling diminati warga dunia, salah satunya karena terkenal dengan dinamika pergantian manager klub sepakbola. Setiap manager klub sepakbola tercatat pernah melatih di beberapa klub sepakbola liga Inggris. Setiap klub sepakbola tercatat sudah berapa kali berganti manager. Seorang peneliti hendak mencari tahu bagaimana fenomena pergantian manager klub sepakbola di liga Inggris. Untuk itu, sang peneliti memodelkan riwayat pergantian manager tersebut dalam relasi M-N sebagai berikut:



KAMUS

```

type AdrMgr : { type terdefinisi, alamat sebuah elemen list Manager }
type AdrFC : { type terdefinisi, alamat sebuah elemen list Football Club }
type AdrCt : { type terdefinisi, alamat sebuah elemen list manage }
type Manager : < Nama:string, NextMgr:AdrMgr >
type FootballClub : < NamaFC: string, JumlahMgr:integer, NextFC:AdrFC >
type Contract : < PMgr:AdrMgr, PFC:AdrFC, NextContract:AdrCt >
    { PMgr contract dengan Football Club PFC }

{ Anggap setiap atribut 3 tipe di atas dapat diakses dengan selektor sesuai nama atributnya,
Misalnya Nama(PMgr) untuk atribut Nama dari manager PMgr.
Elemen list dapat diciptakan dengan fungsi bernama Alokasi(),
misal PMgr ← Alokasi("Alex Ferguson") }

type ListMgr : AdrMgr
type ListFC : AdrFC
type ListContract: AdrCt

{*** Fungsi/Prosedur ***}
function alokMgr(nama: string) → AdrMgr
{Mengalokasikan elemen list Manager bertipe AdrMgr, jika gagal akan mengirimkan NIL}
function alokContract (M1: AdrMgr, F1: AdrFC) → AdrCt
{Mengalokasikan elemen list Contract bertipe AdrCt, jika gagal akan mengirimkan NIL}
function alokFC(namaFC: string, jumlah: integer) → AdrFC
{Mengalokasikan elemen list Football Club bertipe AdrFC, jika gagal akan mengirimkan NIL}
procedure addContract (input FirstMgr>ListMgr, input/output FirstFC>ListFC,
    input/output FirstContract>ListContract,
    input PMgr:AdrMgr, input PFC:AdrFC)
{ Menambah contract baru yg menyatakan adanya pergantian manager PMgr mengelola
football Club PFC.
I.S. PMgr dan PFC masing-masing ada di list FirstMgr dan list FirstFC.
F.S. Jika alokasi berhasil, sebuah contract baru yang menyatakan adanya contract antara
PMgr dan PFC ditambahkan; jumlah manager yang pernah menangani PFC disesuaikan.
Jika alokasi tidak berhasil, tidak ada perubahan pada list contract.}

function portofolio (FirstMgr>ListMgr, FirstFC>ListFC, FirstContract>ListContract,
    Mgr: string) → real
{ Menghitung jumlah contract yang pernah dilakukan seorang manager bernama Mgr
dibandingkan dengan total jumlah football club di liga primer Inggris.
Nama Mgr ada pada list FirstMgr.
Misalkan jumlah football club di liga primer Inggris yang pernah ditangani Mgr
sebanyak 4, total jumlah football club di Inggris yang tercatat di list FirstFC sebanyak
64, maka nilai yang dikirimkan adalah 4/64 atau 0.06 }

function isAlumni(FirstMgr>ListMgr, FirstFC>ListFC,
    FirstContract>ListContract, M1: string, M2: string) → boolean
{ Mencari tahu apakah manager bernama M1 dan M2 pernah menangani minimal 1 football club
yang sama }

```

Implementasikan fungsi/prosedur (a) portofolio dan (b) isAlumni yang ada sesuai definisi dan spesifikasi di atas, tanpa menulis ulang spesifikasinya.

IF2120

Matematika Diskrit

UTS MATEMATIKA DISKRIT

IF2153 Matematika Diskrit

UTS SEMESTER I – 2007/2008

Hari/Tanggal: Kamis, 22 Oktober 2007

Dosen: Harlili, M.Sc. (K-01), Ir. Rinaldi Munir, M.T (K-02)

Waktu: 110 menit

Berdo'alah terlebih dahulu sebelum mengerjakan ujian ini.

1. Dari keempat argumen berikut, argumen manakah yang sahih? (15)
 - (i) Jika hari panas, maka Amir mimisan, tetapi hari ini tidak panas, oleh karena itu Amir tidak mimisan.
 - (ii) Jika hari panas, maka Amir mimisan, tetapi Amir tidak mimisan, oleh karena itu hari ini tidak panas.
 - (iii) Jika Amir mimisan maka hari panas, tetapi hari ini tidak panas, oleh karena itu Amir tidak mimisan.
 - (iv) Jika Amir tidak mimisan, maka hari tidak panas, tetapi Amir mimisan, oleh karena itu hari ini tidak panas.
2. Misalkan A , B , dan C adalah himpunan. Buktikan dengan menggunakan aljabar himpunan:

$$(A - B) \cap (A - C) = A - (B \cup C)$$
 (10)
3. Tentukan apakah setiap relasi R pada himpunan semua bilangan bulat adalah refleksif, simetri (setangkup), anti simetri (tolak setangkup), dan/ atau transitif (menghantar). Beri alasannya.

$$\begin{aligned} R &= \{(x, y) \mid x \neq y\} \\ R &= \{(x, y) \mid x \equiv y \pmod{7}\} \\ R &= \{(x, y) \mid x \geq y^2\} \end{aligned} \quad \begin{array}{c} (\\ 2 \\ 0 \\) \end{array}$$
4. Jika A sebuah himpunan pasangan terurut pada bilangan bulat. Sebuah relasi R didefinisikan pada $A \times A$ dengan aturan

$$(a, b)R(c, d) \text{ jika } \frac{a}{c} = \frac{b}{d}$$

 Periksa apakah relasi R merupakan relasi ekivalen. Jelaskan (15)
5. Untuk biaya pos berapa yang dapat menggunakan perangko Rp 4 dan Rp 5? Buktikan jawaban anda dengan induksi matematik. (15)
6. Temukan 5 buah bilangan bulat positif pertama yang balikannya (*invers*) dalam modulo 9 adalah -3 . (10)
7. Buktikan bahwa jika a , b , k , dan m adalah bilangan bulat sedemikian sehingga $k \geq 1$, $m \geq 2$, dan $a \equiv b \pmod{m}$, maka $a^k \equiv b^k \pmod{m}$. (15)

IF2091 Struktur Diskrit
UTS SEMESTER I – 2008/2009
Hari/Tanggal: Senin, 20 Oktober 2008
Dosen: Harlili, M.Sc. (K-01), Ir. Rinaldi Munir, M.T (K-02)
Waktu: 120 menit

Berdo'alah terlebih dahulu sebelum mengerjakan ujian ini.

1. Persoalan logika, diambil dari *WFF'N PROOF, the Game of Logic*, memiliki dua premis:
 1. "Logika sulit atau tidak banyak mahasiswa menyukai logika".
 2. "Jika matematika mudah, maka logika tidak sulit".

Tentukan kesimpulan mana saja di bawah ini yang valid berdasarkan premis di atas?

 - a) Bahwa matematika tidak mudah, jika banyak mahasiswa menyukai logika
 - b) Bahwa logika tidak sulit atau matematika tidak mudah.
 - c) Bahwa jika tidak banyak mahasiswa menyukai logika, maka matematika tidak mudah atau logika tidak sulit.
2. Misalkan A dan B adalah himpunan bagian dari himpunan semesta U . Buktikan dengan hukum aljabar himpunan bahwa $(A \cap B) \cup (A \cap \bar{B}) = A$.
3. Tentukan apakah masing-masing relasi R pada himpunan semua bilangan bulat di bawah ini adalah refleksif, simetri (setangkup), anti simetri (tolak setangkup), dan/ atau transitif (menghantar). Beri alasannya.
 - (i) $R = \{(x, y) \mid x \neq y\}$
 - (ii) $R = \{(x, y) \mid x \equiv y \pmod{7}\}$
 - (iii) $R = \{(x, y) \mid x \geq y^2\}$
4. Diketahui S sebuah himpunan dan himpunan kuasa dari S . Sebuah relasi \subseteq didefinisikan pada himpunan kuasa dari S . Periksa apakah relasi \subseteq merupakan relasi pengurutan parsial. Jelaskan!
5. Buktikan dengan induksi matematik, untuk semua $n \geq 1$ berlaku
$$\frac{xx^{nn+1} - 1}{xx - 1} = 1 + xx + xx^2 + \dots + xx^{nn} \quad dddddd dddddd xx \neq 1$$
6. Diberikan sebuah sistem kekongruenan berikut ini:
 $3x + 2y \equiv 0 \pmod{71}$
 $2x + 2y \equiv 1 \pmod{71}$
Temukan semua solusi untuk x dan y ! (nyatakan x dan y dalam modulus)
7. Carilah semua bilangan ganjil yang bersisa 2 jika dibagi dengan 3 dan bersisa 6 jika dibagi dengan 7.

UTS SEMESTER I – 2009/2010
 IF2091 Struktur Diskrit
 Rabu, 21 Oktober 2009
 Waktu: 2 jam
 Dosen: Rinaldi Munir dan Harlili

(Harlili: 1, 2, 3)

1. Misalkan A dan B adalah himpunan. Pada kondisi manakah pernyataan dibawah ini benar? (15)

$$\begin{aligned}
 A \cup B &= \\
 A \cap B &= \\
 A & \\
 A - B &= A \\
 A - B &= B - \\
 A & \\
 A \cap B &= B \cap A
 \end{aligned}$$

2. Diketahui fungsi rekursif Ackerman dibawah ini. Tentukan $f(2,1) = ?$

$$f(m, n) = \begin{cases} f(0, n) = n + 1 \\ f(m, 0) = f(m - 1, 1) \end{cases} \quad (15)$$

|

$$\boxed{f(m, n) = f(m - 1, f(m, n - 1))}$$

3. Diketahui R suatu relasi pada himpunan bilangan riil sehingga $a R b$ jika dan hanya jika $a - b =$ bilangan bulat. Buktikan R adalah relasi kesetaraan (ekuivalen). (15)

(Rinaldi Munir: 4, 5, 6, 7)

4. Gunakan induksi matematika untuk membuktikan bahwa $\sim(p_1 \vee p_2 \vee \dots \vee p_n) \Leftrightarrow \sim p_1 \wedge \sim p_2 \wedge \dots \wedge \sim p_n$ dimana p_1, p_2, \dots, p_n adalah proposisi. (15)
5. Temukan semua bilangan (x) yang habis dibagi 5 tetapi bersisa 1 jika dibagi 3. Nyatakan solusi anda sebagai $x = a \pmod{b}$ (15)
6. Diketahui kombinasi lanjar $312m + 70n = 2$. Tentukan nilai mn dalam modulus 8. (10)

7. Periksa validitas argumen berikut ini: (15)

Dia dapat menggunakan program *word processor* bilamana dia mempunyai komputer laptop.

Jika dia menggunakan program *word processor* tentu dia seorang penulis berbakat. Tetapi, dia tidak mempunyai komputer laptop, jadi dia bukanlah seorang penulis berbakat.

UTS IF2091 Struktur Diskrit
UTS SEMESTER I – 2010/2011

Hari/Tanggal: Rabu, 13 Oktober 2010, Waktu: 120 menit
Dosen: Harlili, M.Sc. (K-01), Dr. Ir. Rinaldi Munir, M.T (K-02)

Soal dari Dra. Harlili, M.Sc.

- Misalkan A , B , dan C adalah himpunan. Buktikan secara aljabar himpunan bahwa
$$(A - B) - C = (A - C) - (B - C)$$
- Diketahui fungsi rekursif Ackerman dibawah ini. Tentukan $f(2,3) = ?$

$$f(m, n) = \begin{cases} f(0, n) = n + 1 & | \\ f(m, 0) = f(m - 1, 1) & \square \\ | \\ f(m, n) = f(m - 1, f(m, n - 1)) & | \end{cases}$$

- Tentukan apakah setiap relasi R pada himpunan semua bilangan bulat adalah refleksif, simetri (setangkup), anti simetri (tolak setangkup), dan/ atau transitif (menghantar). Beri alasannya.

$$\begin{aligned} R &= \{(x, y) \mid x \neq y\} \\ R &= \{(x, y) \mid x \equiv y \pmod{7}\} \\ R &= \{(x, y) \mid x \geq y^2\} \\ R &= \{(x, y) \mid x = y + 1 \vee x = y - 1\} \\ R &= \{(x, y) \mid xy \geq 1\} \end{aligned}$$

Soal dari Dr. Ir. Rinaldi Munir

- (a) Buktikan dengan hukum-hukum logika (tanpa tabel kebenaran) bahwa
$$(p \wedge q) \rightarrow (p \vee q)$$

merupakan tautologi, jangan lupa tuliskan nama-nama hukum yang digunakan.
(b) Sebaliknya, tunjukkan (tidak perlu pakai hukum-hukum logika) bahwa
$$(p \vee q) \rightarrow (p \wedge q)$$
 bukan merupakan tautologi

- Buktikan (terserah bagaimana caranya) bahwa sekumpulan premis berikut menghasilkan konklusi yang disebutkan dibawahnya:

$$\begin{aligned} p &\rightarrow q \\ r &\rightarrow p \\ \sim p & \\ \therefore \sim r & \end{aligned}$$

- Misalkan $A = \begin{bmatrix} a & 0 \\ 0 & b \end{bmatrix}$

$[0]$

yang dalam hal ini a dan b adalah bilangan riil. Buktikan dengan induksi matematika bahwa A^n

$b]$

$$= \begin{bmatrix} a^n & 0 \\ 0 & b^n \end{bmatrix}. \quad \square$$

- Tentukan dua bilangan asli terkecil yang jika dibagi 3 menyisakan 2, dibagi 4 menyisakan 3, dan dibagi 5 menyisakan 4.

UTS IF2091 Struktur Diskrit
UTS SEMESTER I – 2011/2012

Hari/Tanggal: Rabu, 19 Oktober 2011, Waktu: 100 menit
Dosen: Harlili, M.Sc. (K-02), Dr. Ir. Rinaldi Munir, M.T (K-01)

Berdo'alah terlebih dahulu sebelum mengerjakan ujian ini.

Soal dari Dra. Harlili, M.Sc.

1. Misalkan A dan B adalah himpunan multi set. $A = \{1, 1, 1, 1, 2, 2, 3, 3, 3, 3, 3, 4, 4, 4\}$ dan $B = \{1, 1, 2, 2, 2, 3, 4, 4, 4, 4\}$ Tentukan
 - a) $A \cap B$
 - b) $A \cup B$
 - c) $A - B$
 - d) $A + B$
2. Diketahui fungsi rekursif *Ackerman* dibawah ini. Tentukan $f(2, 1) = ?$

$$f(m, n) = \begin{cases} f(0, n) = n + 1 \\ f(m, 0) = f(m - 1, 1) \\ f(m, n) = f(m - 1, f(m, n - 1)) \end{cases}$$

3. Diketahui R suatu relasi pada himpunan bilangan bulat sehingga $a R b$ jika dan hanya jika a dan b keduanya negatif atau keduanya positif. Buktikan apakah R adalah relasi kesetaraan (ekuivalen).

Soal dari Dr. Ir. Rinaldi Munir

4. Diberikan pernyataan “Untuk mendapatkan satu kupon undian, Anda cukup membeli dua produk senilai Rp. 50.000,-”.
 - a) Nyatakan pernyataan di atas dalam bentuk proposisi “jika p , maka q ”.
 - b) Tentukan ingkaran, konvers, invers, dan kontraposisi dari pernyataan tersebut.
5. Buktikan dengan induksi matematika bahwa untuk setiap n bilangan asli, $dd^3 + 11dd$ selalu habis dibagi 6
6. Misalkan x adalah suatu bilangan ganjil positif yang memenuhi $5x \equiv 7 \pmod{9}$. Tentukan nilai dari $2x \pmod{12}$.
7. Dengan memanfaatkan algoritma Euclid, tentukan PBB(182, 133). Lalu tentukan suatu pasangan bilangan bulat (x, y) yang memenuhi persamaan:
 - a) $182x + 133y = 14$
 - b) $182x + 133y = 10$

Program Studi Teknik Informatika
 Sekolah Teknik Elektro dan Informatika
 Institut Teknologi Bandung

UTS IF2091 Struktur Diskrit

UTS SEMESTER I – 2012/2013

Hari/Tanggal: Rabu, 17 Oktober 2012, Waktu: 100 menit
 Dosen: Harlili, M.Sc. (K-01), Dr. Ir. Rinaldi Munir, M.T (K-02)

Berdo'alah terlebih dahulu sebelum mengerjakan ujian ini.
 Soal dari Dra. Harlili, M.Sc.

- Misalkan A dan B adalah himpunan pada himpunan universal U . Tentukan daftar urutan ini secara membesar berdasarkan banyaknya anggota:

$$|A - B|, |A \cup B|, |\emptyset|, |A \cap B|, |A| + |B|.$$

(15)

- Tentukan fungsi-fungsi dibawah ini yang merupakan fungsi 1-1 pada (*bijection*), f adalah fungsi dari R ke R.
 - $f(x) = 2x + 1$
 - $f(x) = x^2 + 1$
 - $f(x) = x^3$
- Tentukan apakah relasi R pada himpunan semua bilangan bulat adalah refleksif, simetri (setangkup), anti simetri (tolak setangkup), dan/atau transitif (mengantar). Beri alasan.
 - $R = \{(x,y) | xy \geq 1\}$
 - $R = \{(x,y) | x = \text{kelipatan dari } y\}$
 - $R = \{(x,y) | x \geq y^2\}$

(15)

Soal dari Dr. Ir. Rinaldi Munir

- Buktikan validitas argumen berikut dengan cara apapun: "Saya bisa mengerjakan soal ujian bilamana saya sudah belajar atau mencontek saat ujian. Saya mencontek jika saya tidak tahu malu. Tetapi, saya tidak belajar dan tidak tahu malu. Karena itu saya bisa mengerjakan soal ujian" (15)
- Dengan mengguankan algoritma Euclidean, tentukan suatu pasangan bilangan bulat (x,y) yang memenuhi persamaan $56x + 91y = 7$. (15)

- Pecahan sistem kekongruenan lanjar berikut:

$$\boxed{2x \equiv 6 \pmod{10}}$$

(15)

$$| 3x \equiv 11 \pmod{14}$$

- Tunjukkan apa yang salah dengan pembuktian menggunakan prinsip induksi kuat di bawah ini:
 "Teorema: Untuk setiap *integer* non-negatif n , $5n = 0$."
Basis: Untuk $n = 0$, $5 \cdot 0 = 0$

Langkah induksi: Misalkan bahwa $5k = 0$ untuk semua *integer* non-negatif k dengan $0 \leq k \leq n$ (hipotesis induksi). Untuk $n + 1$ tulislah $n + 1 = i + k$ yang dalam hal ini i dan k adalah bilangan asli yang lebih kecil dari $n + 1$. Dengan menggunakan hipotesis induksi, $5(n + 1) = 5(i + k) = 5i + 5k = 0 + 0 = 0$.

(catatan: bilangan asli dimulai dari 1)

(10)

Program Studi Teknik Informatika
 Sekolah Teknik Elektro dan Informatika
 Institut Teknologi Bandung

UTS IF2120 Matematika Diskrit
UTS SEMESTER I – 2013/2014

Hari/Tanggal: Rabu, 23 Oktober 2013, Waktu: 100 menit
 Dosen: Harlili, M.Sc. (K-02), Dr. Ir. Rinaldi Munir, M.T (K-01)

Soal dari Dra. Harlili, M.Sc.

1. Successor dari himpunan A didefinisikan sebagai himpunan $A \cup \{A\}$ ditulis $\{A, \{A\}\}$. Tentukan successor dari :
 - a) $\{1,2,3\}$
 - b) \emptyset
 - c) $\{\emptyset\}$
 - d) $\{\emptyset, \{\emptyset\}\}$
 (Nilai: 10)
2. Misalkan f adalah fungsi dari $X = \{0,1,2,3,4\}$ ke X yang didefinisikan oleh $f(x) = 3x \bmod 5$. Tuliskan f ini sebagai himpunan pasangan terurut. Apakah f merupakan fungsi satu-satu atau pada.
 (Nilai: 10)
3. Matriks-matriks dibawah ini menyatakan suatu pasangan terurut dari relasi R pada $\{1,2,3\}$. Tentukan apakah relasi R adalah refleksif, simetri (setangkup), anti simetri (tolak setangkup), dan/ atau transitif (mengantar). Beri alasannya.
 (Nilai: 15)

	$\begin{bmatrix} 1 \\ \\ 1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 \\ & \\ 1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 \\ \\ 0 \end{bmatrix} \quad \begin{bmatrix} 0 \\ \\ 0 \end{bmatrix}$	
(i)			
	$\begin{bmatrix} 1 \\ \\ 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 \\ & \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 1 \\ \\ 1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 \\ & \\ 0 & 0 \end{bmatrix}$	(ii)

4. Misalkan R adalah suatu relasi pada himpunan dari semua string biner dengan aturan $s R t$ jika dan hanya jika s dan t adalah string-string biner yang memiliki banyak simbol 1 sama. Perlihatkan R adalah relasi ekuivalen.
 (Nilai: 15)

Soal dari Dr. Ir. Rinaldi Munir

5. Buktikan dengan induksi matematika bahwa untuk $f(x) = (\sin x)^n$ maka turunan pertamanya adalah $f'(x) = n(\sin x)^{n-1} \cos x$.
 (Nilai: 10)
6. Selesaikan relasi rekurens berikut: $T(n) = 7T(n-1) - 6T(n-2)$; $T(0) = 2$, $T(1) = 7$ dengan menggunakan metode persamaan karakteristik. (Catatan: T_n ditulis $T(n)$, T_{n-1} ditulis $T(n-1)$, dst).
 (Nilai: 15)
7. Dengan menggunakan Teori Bilangan, buktikan bahwa $8|(7^{2nn+1} + 1)$
 (Nilai: 10)
8. Sebuah buku memiliki ISBN 135-09-2xy9-1. Diketahui pula bahwa $yy^2 \equiv 0 \pmod{5}$. Tentukan semua kemungkinan pasangan (x, y) yang mungkin dari ISBN tersebut.
 (Nilai: 15)

UTS IF2120 Matematika Diskrit
 UTS SEMESTER I – 2014/2015
 Hari/Tanggal: Rabu, 22 Oktober 2014
 Dosen: Harlili, M.Sc. (K-01), Ir. Rinaldi Munir, M.T (K-02)
 Waktu: 120 menit

Setiap soal nilainya = 10.

1. Nyatakan kalimat ini dalam notasi simbolik: *Karena hari Sabtu dua minggu lalu diadakan penutupan PMB 2014, acara kumpul rutin UATM dibatalkan dan rapat ITB Open ditunda hari ini, kecuali ada pemberitahuan lebih lanjut.*
2. Berapa banyak bilangan bulat positif antara 1 dan 1000 habis dibagi 7 atau 11.
3. Tentukan apakah relasi R pada himpunan semua bilangan bulat adalah relasi ekivalen. Beri alasannya dengan menentukan apakah masing-masing relasi bersifat refleksif, simetri (setangkup), anti simetri (tolak setangkup), dan/ atau transitif (mengantar).

$$R = \{(x, y) \mid x \neq y\}$$

$$R = \{(x, y) \mid x \equiv y \pmod{7}\}$$

$$R = \{(x, y) \mid x \geq y^2\}$$

4. John McCarthy salah satu perintis intelektual buatan mendefinisikan fungsi di bawah ini. Tentukan $M(101)$ dan $M(76)$.

$$M(n) = \begin{cases} n - 10, & n > 100 \\ M(M(n + 11)), & n \leq 100 \end{cases}$$

5. Tunjukkan apa yang salah dalam pembuktian dengan induksi matematik berikut ini? Teorema. Untuk setiap integer positif n , jika x dan y adalah integer positif dengan $\max(x, y) = n$, maka $x = y$.

Basis: Misalkan bahwa $n = 1$. Jika $\max(x, y) = 1$ dan x dan y integer positif, maka diperoleh $x = 1$ dan $y = 1$.

Langkah induksi: Misalkan k adalah *integer* positif. Asumsikan bahwa bilamana $\max(x, y) = k$ dan x dan y adalah integer positif, maka $x = y$ (*hipotesis induksi*). Sekarang misalkan $\max(x, y) = k + 1$, yang dalam hal ini x dan y adalah *integer* positif. Maka $\max(x - 1, y - 1) = k$, sehingga dari hipotesis induksi, $x - 1 = y - 1$. Dari sini diperoleh $x = y$.

Karena langkah dan basis dan langkah induksi sudah dibuktikan benar, maka terbukti bahwa Teorema di atas benar.

6. Sebuah buku memiliki kode ISBN 947-2309-97. Jika karakter uji buku tersebut adalah p , maka hitunglah berapa nilai $p^{-1} \pmod{17}$.
7. Hitunglah nilai $2^{2020} \pmod{73}$ dengan bantuan Teorema Fermat.
8. Berapa banyak string biner berbeda yang ditransmisikan dari 1 *byte* (8 bit) jika string harus mulai dengan simbol 1 dan mempunyai paling sedikit dua simbol 0.
9. Berapa banyak solusi bilangan bulat dari $x_1 + x_2 + x_3 + x_4 + x_5 = 21$ jika $0 \leq x_i \leq 10$ dan $x_i \geq 0$ untuk $i \geq 2$.
10. Berapa koefisien $x^{101}y^{99}$ pada $(2x - 3y)^{200}$?

UTS IF2120 Matematika Diskrit (3 SKS)

UTS SEMESTER I – 2015/2016

Dosen: Rinaldi Munir, Harlili

Rabu, 21 Oktober 2015

Waktu: 100 menit

Soal dari Rinaldi Munir

1. Diberikan beberapa pernyataaan di dalam sebuah argumen sebagai berikut:
 - (i) Jika program komputer ini benar, maka ia menghasilkan luaran (*output*) yang benar bilamana ia *di-run* dengan data uji yang diberikan oleh asisten
 - (ii) Program komputer ini menghasilkan luaran (*output*) yang benar bilamana ia *di-run* dengan data uji yang diberikan oleh asisten.
 - (iii) Oleh karena itu, program komputer ini benar

Jawablah pertanyaan berikut:

- (a) Nyatakan argumen di atas dalam notasi simbolik
- (b) Perlihatkan apakah argumen tersebut sahih (valid)?
2. Di kantor pos hanya tersedia perangko senilai Rp2 dan perangko senilai Rp5. Untuk biaya pos berapa saja yang dapat mengguankan perangko-perangko tersebut? Buktikan jawabanmu dengan induksi matematika.
3. Azizah memiliki sejumlah coklat di rumahnya. Jika ia membagi seluruhnya secara merata ke 5 orang, tersisa 4 coklat. Jika ia membagi seluruhnya secara merata ke 8 orang, tersisa 6 coklat. Jika ia membagi seluruhnya secara merata ke 9 orang, tersisa 8 coklat. Berapa jumlah coklat paling sedikit yang Azizah miliki?
4. Hitunglah sisa 5^{2003} dibagi dengan 13 (Petunjuk: Gunakan Teorema Fermat)
5. Berapakah nilai x dan y bilangan bulat yang memenuhi persamaan $1757xx - 1631yy = 483$?

Soal dari Harlili

6. Diantara bilangan bulat 1 sampai 300 (termasuk 1 dan 300 sendiri) berapa banyak bilangan yang TIDAK habis dibagi 3 atau 5?
7. Diketahui f adalah fungsi dari $X = \{0,1,2\}$ ke X didefinisikan oleh $f(x) = (x^2+x+1) \text{ mod } 3$. Tuliskan f ini sebagai himpunan pasangan terurut. Apakah f merupakan fungsi satu-satu atau pada.
8. Sebuah relasi R didefinisikan pada himpunan bilangan riil dengan aturan $a R b$ jika dan hanya jika $a - b =$ bilangan bulat. Periksa apakah relasi R merupakan relasi ekivalen. Jelaskan.
9. Suatu fungsi Boolean 3-variabel pada $\{0,1\}$ didefinisikan sbb: $f(x, y, z) = (x + y + z) \text{ mod } 2$. Sederhanakan fungsi ini sebagai SOP dengan Karnaugh Map, lalu gambarkan rangkaian logikanya.

UTS IF2120 Matematika Diskrit (3 SKS)

UTS SEMESTER I – 2016/2017

Dosen: Rinaldi Munir, Harlili

Jumat, 14 Oktober 2016

Waktu: 100 menit

Soal dari Rinaldi Munir

1. Apakah $p \wedge (p \leftrightarrow q) \wedge \sim q$ merupakan tautologi atau kontradiksi? Tunjukkanlah dengan

hukum-hukum logika tanpa menggunakan tabel kebenaran! Jangan lupa menyebutkan hukum logika yang dipakai!

2. Mesin jaja (*vending machine*) di sebuah Bandara menjual berbagai jenis minuman. Mesin jaja tersebut hanya dapat menerima uang dengan pecahan Rp2000 dan Rp5000 dan tidak dapat memberi kembalian. Minuman seharga berapa saja yang dapat dibayar dengan uang pecahan tersebut Rp2000 dan Rp5000 tersebut? Buktikan jawaban anda dengan induksi matematika.
3. Gunakan Peta Karnaugh untuk membuat rangkaian logika yang menerima masukan kode binerl dari suatu angka desimal dan menghasilkan keluaran 1 jika masukan tersebut TIDAK habis dibagi 3. Fungsi minimasi dalam bentuk POS. Gambarkan rangkain logikanya!

Soal dari Harlili

4. Diantara bilangan bulat 1 sampai 70 (termasuk 1 dan 70 sendiri) berapa banyak bilangan yang TIDAK habis dibagi 2 atau 5 atau 7? Ini berarti mencari berapa banyak bilangan bulat 1 sampai 70 yang relatif prima dengan 70.
5. John McCarthy salah satu perintis intelektual buatan mendefinisikan fungsi ini. Tentukan $M(102)$ dan $M(87)$.

$$M(n) = \begin{cases} n - 10, & n > 100 \\ M(M(n + 11)), & n \leq 100 \end{cases}$$

6. Tentukan apakah relasi R pada himpunan semua bilangan bulat bersifat refleksif, simetri (setangkup), anti simetri (tolak setangkup), dan/ atau transitif (menghantar). Beri alasannya.

$$R = \{(x, y) \mid xy \geq 1\}$$

$$R = \{(x, y) \mid x \equiv y \pmod{7}\}$$

$$R = \{(x, y) \mid x = y^2\}$$

7. Suatu eksperimen bakteri menyatakan banyak bakteri dalam suatu koloni sama dengan 3 kalinya untuk setiap jam.
 - a. Tentukan relasi rekuren a_n yang menyatakan banyak bateri setelah n jam.
 - b. Tentukan solusi relasi rekuren tersebut.
 - c. Jika pada saat awal ada 10 bateri di suatu koloni, tentukan banyak bakteri setelah 5 jam.

UTS IF2120 Matematika Diskrit (3 SKS)
 UTS SEMESTER I – 2017/2018
 Dosen: Rinaldi Munir, Harlili
 Jumat, 13 Oktober 2017
 Waktu: 110 menit

Soal dari Rinaldi Munir

- Dari beberapa pernyataan berikut tentukanlah apakah Andi berhasil lulus maka kuliah Matematika Diskrit (Matdis) dengan menggunakan hukum-hukum logika dan penarikan kesimpulan yang sahih.

Jika Andi lulus ujian Matdis dan lulus kuis Matdis maka ia lulus mata kuliah Matdis. Jika Andi mengikuti tutorial Matdis maka ia lulus ujian Matdis. Faktanya, Andi lulus kuis Matdis atau lulus kuis Logif (Logika Informatika). Diberitahu bahwa jika Andi lulus kuis Logif maka ia senang. Oh iya, Andi mengikuti tutorial Matdis. Namun, Andi tidak senang.

Petunjuk: Gunakanlah proposisi berikut:

p : Andi lulus ujian Matdis; q : Andi lulus kuis Matdis; r : Andi lulus mata kuliah Matdis
 s : Andi mengikuti tutorial Matdis; t : Andi lulus kuis Logif; u : Andi senang

(Nilai: 15)

- Perlihatkan kesamaan ini dengan menggunakan hukum-hukum aljabar himpunan, sebutkan nama hukumnya : $(A - B) \cup (B - A) \cup (A \cap B) = A \cup B$
(Nilai: 10)
- Diketahui f adalah fungsi dari $X = \{0,1,2\}$ ke X , didefinisikan oleh $f(x) = (2x+1) \bmod 3$. Tuliskan f ini sebagai himpunan pasangan terurut. Apakah f merupakan fungsi berkoresponden satu-ke-satu dan pada (bijection). Jika f fungsi berkoresponden satu-ke-satu dan pada, periksa $f \circ f^{-1} = I(x)$, dimana f^{-1} adalah fungsi invers dari f dan $I(x)$ adalah fungsi Identitas yaitu $I(x) = x$.
 (Nilai: 10)
- Tentukan apakah relasi R pada himpunan semua bilangan bulat adalah refleksif, simetri(setangkup), anti simetri (tolak setangkup), dan/atau transitif (menghantar). Beri alasan.
 - $R = \{(x,y) | xy \geq 0\}$
 - $R = \{(x,y) | x = \text{kelipatan dari } y\}$
- Carilah solusi dari relasi rekurens berikut:
 - $a_n = a_{n-1} + 6a_{n-2}$, untuk $n \geq 2$ dengan $a_0 = 3$; $a_1 = 6$
 - $a_n = 2a_{n-1} - a_{n-2}$, untuk $n \geq 2$ dengan $a_0 = 4$; $a_1 = 1$
(Nilai: 15)
- Gunakan Karnaugh-Map untuk meminimumkan bentuk boolean SOP berikut, kemudian gambarkan rangkaian logikanya.
 - $f(x, y, z) = x'yz + x'y'z$
 - $f(x, y, z) = xyz + xy'z + xy'z' + x'yz + x'yz' + x'y'z'$
(Nilai: 15)
- Buktikanlah dengan menggunakan induksi matematika, untuk semua bilangan bulat positif n , bahwa $3^n - 1$ adalah kelipatan 2.
 (Nilai: 10)

8. Sebuah mesin jaja minuman (*vending machine*) menjual hanya 1 jenis minuman yang berharga 8 sen. Mesin dapat menerima pembayaran dengan koin 3 sen dan 5 sen. Jika pembeli memasukkan uang yang cukup (atau lebih dari 8 sen) untuk membeli minuman, maka lampu *vending machine* akan menyala dan minuman akan keluar (satu buah). Sebaliknya, *vending machine* akan menunggu pembeli untuk melunasi pembayaran. Hal yang perlu diperhatikan lainnya adalah mesin tidak memiliki fitur uang kembalian. Rancanglah rangkaian logika pada mesin ini.

(Nilai: 15)

Catatan: Pembayaran dapat dilakukan dengan koin 3 sen saja atau koin 5 sen saja atau kombinasi keduanya.

Definisikan berapa variabel yang akan anda gunakan dan apa yang direpresentasikan oleh variabel tersebut.

UTS IF2120 Matematika Diskrit (3 SKS)

UTS SEMESTER I – 2019/2020

Dosen: Rinaldi Munir, Harlili, Fariska Zakhralatifa

Jumat, 11 Oktober 2019

Waktu: 110 menit

1. Diantara bilangan bulat 1 sampai 300 (termasuk 1 dan 300 sendiri) berapa banyak bilangan yang tidak habis dibagi 3 atau 5, namun tidak keduanya? **(10)**

2. Misalkan n menyatakan bilangan bulat positif. Diketahui sebuah fungsi f yang didefinisikan secara rekursif sebagai berikut:

$$n = 1 : f(n) = 1$$

$$n > 1: f(n) = f(n \text{ div } 5) + 2$$

Tentukan: (a) $f(54)$ dan (b) $f(107)$ **(10)**

3. Tentukan sifat-sifat relasi: refleksif, mengantar, setangkup, dan/atau tolak setangkup, dari relasi R berikut, termasuk alasannya. **(15)**

a. $R = \{(x, y) \mid \frac{1}{2}xy \geq 1\}$; x, y bilangan bulat

b. $M_R =$

$$\begin{matrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{matrix}$$

4. Ingat kembali persoalan pemindahan ibukota ke tempat yang baru, seperti yang diujikan di dalam Kuis 1. Pemerintah berencana membangun infrastruktur jalan untuk menghubungkan antar kabupaten di Provinsi Kalimantan Timur. Terdapat 4 kabupaten yang akan dibangun jalannya, yaitu Kabupaten Paser, Balikpapan, Bontang, dan Samarinda. Semua jalan yang menghubungkan kota A dan B hanya berlaku satu arah, artinya jika ada jalan dari kota A ke kota B belum tentu terdapat jalan sebaliknya. Sampai saat ini hanya terdapat 6 jalan yang terbangun yaitu jalan **Paser-Balikpapan, Bontang – Samarinda, Balikpapan – Bontang, Samarinda – Bontang, Paser – Bontang, dan Samarinda - Paser**. Tentukan ruas jalan yang harus dibangun secara minimal, jika terdapat jalan dari kota A ke kota B dan dari kota B ke kota C, maka terdapat jalan dari kota A ke kota C. Pada kasus ini jika diperlukan memungkinkan bahwa kota A = kota B. **(20)**

(a) Termasuk sifat relasi apakah persoalan ini?

(b) Bagaimana solusinya?

5. Suatu eksperimen bakteri menyatakan banyak bakteri dalam suatu koloni sama dengan 2 kalinya untuk setiap jam.

(15)

a. Tentukan relasi rekuren a_n = banyak bakteri setelah n jam.

b. Tentukan solusi relasi rekuren (a).

Jika pada saat awal ada 10 bakteri di suatu koloni, tentukan banyak bakteri setelah 3 jam.

6. Di negara Oz hanya ada koin 3 sen dan 8 sen. Jika pembayaran hanya dapat dilakukan kedua macam koin tersebut, berapa paling sedikit nilai pembayaran yang menggunakan kedua koin tersebut?
(15)
7. Sederhanakan fungsi Boolean berikut dengan menggunakan metode Peta Karnaugh, kemudian gambarkan rangkaian digitalnya dengan menggunakan:
(15)
- gerbang dasar (AND, OR, NOT)
 $f(w, x, y, z) = \sum(0, 2, 3, 5, 7, 11, 12)$
 $d(w, x, y, z) = \sum(9, 13, 14, 15)$
 - gerbang NAND saja
 $g(x,y,z) = (x' + y')(x + z')(y' + z')$

UTS Matdis 2020/2021**16 Oktober 2020****HIMPUNAN**

1. Dalam seleksi penerima beasiswa prestasi, setiap mahasiswa harus lulus tes TPA (Tes Potensi Akademik) dan Bahasa Inggris. Dari 180 peserta terdapat 103 orang dinyatakan lulus tes TPA dan 142 orang lulus tes Bahasa. Banyak siswa yang dinyatakan lulus sebagai penerima beasiswa prestasi adalah . .Q sama dengan inverse(BA)
 - a. 38 orang
 - b. 45 orang
 - c. 65 orang
 - d. 77 orang
 - e. 58 orang
2. Misalkan A dan B adalah himpunan pada himpunan universal U. Daftar urutan banyaknya anggota secara membesar, mulai dari himpunan banyak anggota sedikit sampai himpunan banyak anggota banyak yang benar adalah
 - a. $|\emptyset|, |A - B|, |A \cup B|, |A \cap B|, |A| + |B|$
 - b. $|\emptyset|, |A \cap B|, |A - B|, |A \cup B|, |A| + |B|$
 - c. $|\emptyset|, |A \cap B|, |A \cup B|, |A - B|, |A| + |B|$
 - d. $|\emptyset|, |A \cap B|, |A - B|, |A| + |B|, |A \cup B|$
 - e. $|\emptyset|, |A - B|, |A \cap B|, |A| + |B|, |A \cup B|$
3. Operasi beda setangkup dari himpunan A dan B dinyatakan oleh $A \oplus B$ adalah himpunan yang memuat anggota A atau B tetapi tidak di keduanya A dan B. Pernyataan yang benar dari kesamaan operasi beda setangkup adalah
 - a. $(A \oplus B) \oplus B = A$
 - b. $(A \oplus A) \oplus B = A$
 - c. $(A \oplus B) \oplus B = B$
 - d. $(B \oplus A) \oplus B = A$
 - e. $(B \oplus A) \oplus A = B$
4. $P(A)$ adalah himpunan kuasa (power set). Jika a dan b adalah berbeda merupakan anggota himpunan, tentukan berapa banyak anggota (elemen) dari power set $P(\{\emptyset, a, \{a\}, \{\{a\}\}\})$.
 - a. 4
 - b. 8
 - c. 6
 - d. 16
 - e. 12
5. Himpunan ganda (Multi set) adalah himpunan yang anggotanya boleh berulang. A dan B adalah himpunan ganda, $A = \{a, a, a, b, b, c\}$ $B = \{a, a, b, b, b, d, d, d, d\}$. Pernyataan yang benar dari operasi dibawah adalah
 - a. $A \cap B = \{a, a, b, b, c, d\}$
 - b. $A - B = \{a, b, b, c\}$
 - c. $A + B = \{a, a, a, a, b, b, b, c, d\}$
 - d. $A + B = \{a, a, a, b, b, b, b, c, d\}$
 - e. $A \cap B = \{a, a, b, b\}$

RELASI DAN FUNGSI

6. Manakah diantara relasi berikut yang bersifat: refleksif, setangkup, tidak menghantar:
 - a. Himpunan $A = \{1, 2, 3, 4\}$, $R = \{(1,1), (1,3), (2, 2), (2, 3), (2, 4), (3, 3), (4, 1), (4, 3), (4, 4)\}$

- b. Himpunan $A = \{1, 2, 3, 4\}$, $R = \{(1, 1), (1, 2), (1, 4), (2, 1), (2, 2), (2, 3), (3, 2), (3, 3), (4, 1), (4, 4)\}$
- c. Himpunan $A = \{1, 2, 3\}$, $R = \{(1, 1), (1, 2), (1, 3), (2, 1), (2, 2), (2, 3), (3, 1), (3, 2), (3, 3)\}$
- d. Himpunan $A = \{1, 2, 3, 4\}$, $R = \{(1, 1), (1, 3), (2, 2), (2, 3), (2, 4), (3, 3), (3, 4), (4, 3), (4, 4)\}$
- e. Himpunan $A = \{1, 2, 3\}$, $R = \{(1, 1), (1, 2), (2, 1), (2, 2), (2, 3), (3, 1), (3, 2), (3, 3)\}$
- f. Semua jawaban tidak ada yang benar
7. Diketahui relasi yang didefinisikan pada himpunan $A = \{a, b, c, d\}$. Relasi direpresentasikan dalam graf berarah berikut ini:
-
- Relasi tersebut tidak menghantar. Agar relasi bersifat menghantar, maka busur yang perlu ditambahkan adalah:
- a. $(b, d), (d, b), (d, d)$
- b. $(b, d), (d, d)$
- c. $(a, a), (d, b)$
- d. $(a, a), (b, d), (d, d)$
- e. $(a, a), (b, d), (d, b), (d, d)$
- f. Semua jawaban di atas tidak ada yang benar
8. Tentukan sifat relasi pada himpunan semua bilangan bulat berikut:
 $(x, y) \in R$ jika dan hanya jika $xy \geq 1$
apakah memiliki sifat refleksif, menghantar, setangkup atau tolak setangkup:
- a. Tidak refleksif, tidak menghantar, setangkup, tolak setangkup
- b. Refleksif, tidak menghantar, setangkup, tidak tolak setangkup
- c. Tidak refleksif, menghantar, setangkup, tolak setangkup
- d. Refleksif, menghantar, setangkup, tidak tolak setangkup
- e. Tidak refleksif, menghantar, setangkup, tidak tolak setangkup
- f. Semua jawaban di atas tidak ada yang benar
9. Diberikan fungsi bilangan riil sebagai berikut: $f(x) = (x + 1)/(x + 2)$. Fungsi tersebut bersifat:
- a. Injektif, surjektif, bijektif
- b. Tidak injektif, surjektif, tidak bijektif
- c. Injektif, tidak surjektif, tidak bijektif
- d. Tidak injektif, tidak surjektif, tidak bijektif
- e. Bukan fungsi
- f. Tidak dapat ditentukan sifatnya karena bukan fungsi integer
10. Fungsi $f(n) = \text{floor}(n/5)$ untuk n bilangan bulat memiliki sifat:
- a. Injektif, surjektif, bijektif
- b. Tidak injektif, surjektif, tidak bijektif
- c. Injektif, tidak surjektif, tidak bijektif

- d. Tidak injektif, tidak surjektif, tidak bijektif
e. Bukan fungsi
f. Semua jawaban di atas salah
11. Fungsi $f(x) = (3x) \text{ mod } 5$ adalah fungsi:
a. Satu-ke-satu (one-to-one)
b. Pada (onto)
c. Satu-ke-satu, tetapi tidak pada
d. Bukan fungsi satu-ke-satu, tetapi fungsi pada
e. Bukan fungsi satu-ke-satu dan bukan fungsi pada
f. Bukan fungsi
12. Diberikan empat buah relasi sebagai berikut:
(a) $\{(a, b) \mid \text{jika } a \text{ adalah pernyataan yang benar, maka pernyataan } b \text{ bernilai benar}\}$
(b) $\{(a,b) \mid \text{jarak kota } a \text{ ke kota Bandung sama dengan jarak kota } b \text{ ke kota Jakarta}\}$
(c) $\{(a,b) \mid \text{jarak kota } a \text{ ke kota } b \text{ kurang dari } 100 \text{ km }\}$
(d) $\{(a,b) \mid a \text{ dan } b \text{ adalah dua himpunan tidak kosong yang irisananya bukan himpunan kosong}\}$
Dari keempat relasi di atas, yang bukan relasi kesetaraan adalah:
a. (a) dan (b) saja
b. (a) dan (c) saja
c. (b) saja
d. (d) saja
e. (b) dan (c) saja
f. Tidak ada jawaban yang benar
- ### INDUKSI MATEMATIKA
13. Terdapat perangko dengan nilai 5 sen dan 7 sen. Dari dua jenis perangko tersebut, maka tentukan pernyataan berikut ini benar atau salah.
- Biaya pos terkecil yang bisa digunakan sebagai basis untuk membuktikan bahwa hanya dengan dua perangko tersebut bisa untuk mengirimkan surat adalah 14 sen. (**nilai: 1**) **Salah**
 - Biaya pos terkecil yang bisa digunakan sebagai basis untuk membuktikan bahwa hanya dengan dua perangko tersebut bisa untuk mengirimkan surat adalah 24 sen. (**nilai: 1**) **Benar**
 - Biaya pos terkecil yang bisa digunakan sebagai basis untuk membuktikan bahwa hanya dengan dua perangko tersebut bisa untuk mengirimkan surat adalah 12 sen. (**nilai: 1**) **Salah**
 - Pada langkah induksi, kita asumsikan biaya pos untuk senilai n sen menggunakan dua buah perangko 7 sen, maka biaya $n+1$ sen dapat diperoleh dengan menggunakan 3 buah perangko senilai 5 sen. (**nilai: 1**) **Benar**
14. Tentukan apakah pembuktian proposisi bilangan bulat berulat berikut ini memerlukan induksi kuat atau tidak.
- Setiap bilangan bulat n ($n \geq 2$) dapat dinyatakan sebagai perkalian dari (satu atau lebih) bilangan prima. (**nilai: 1**) **Perlu**
 - Penjumlahan sejumlah n bilangan positif adalah $P(n) = n(n+1)/2$. (**nilai: 1**) **Tidak**
 - Permainan dua orang, di mana pada setiap gilirannya, seorang pemain mengambil sejumlah korek (bilangan positif) dari satu tumpukan korek (terdapat 2 tumpukan korek). Pemenang permainan adalah pemain yang mengambil korek terakhir. Buktikan jika dua tumpukan mengandung jumlah korek yang sama (setiap tumpukan berisi n korek), maka pemain yang mendapat giliran ke-2 pasti menang. (**nilai: 1**) **Perlu**
 - Untuk n bilangan bulat positif, maka $n^5 - n$ habis dibagi 5. (**nilai: 1**) **Tidak**
15. Ingin dibuktikan bahwa $a^n = 1$ untuk semua bilangan bulat tak-negatif n bilamana a adalah bilangan riil tidak-nol, dengan menggunakan induksi kuat. Tentukan pernyataan berikut ini benar atau salah.
- Basis pembuktian untuk $n=0$ bernilai benar. (**nilai: 1**) **Benar**

- b. Karena ingin dibuktikan dengan induksi kuat, maka perlu dilakukan pembuktian untuk nilai j di mana $1 \leq j \leq k$, sebagai asumsi/ hipotesis untuk membuktikan $n=k+1$ benar. **(nilai: 1) Benar**
- c. Induksi kuat menunjukkan bahwa $a_0 = 1, a_1 = 1, a_2 = 1, \dots, a_j = 1$. **(nilai: 1) Salah**
- d. Untuk pembuktian pada langkah induksi dengan $n=k+1$, maka dapat digunakan formula sebagai berikut, dengan memanfaatkan hipotesis induksi pada langkah (c):

$$a^{k+1} = \frac{a^k \cdot a^k}{a^{k-1}} = \frac{1 \cdot 1}{1} = 1.$$

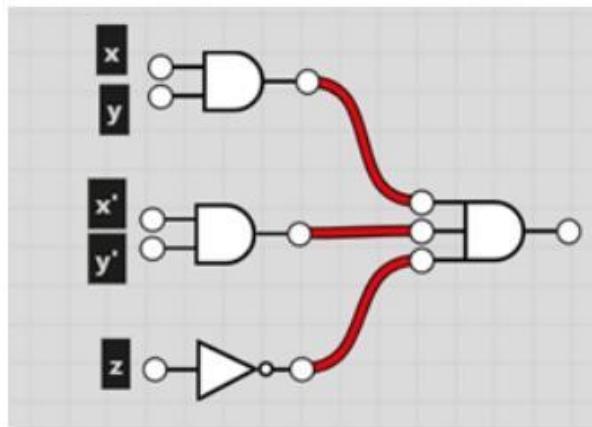
(nilai: 1) Benar

RELASI REKURENS

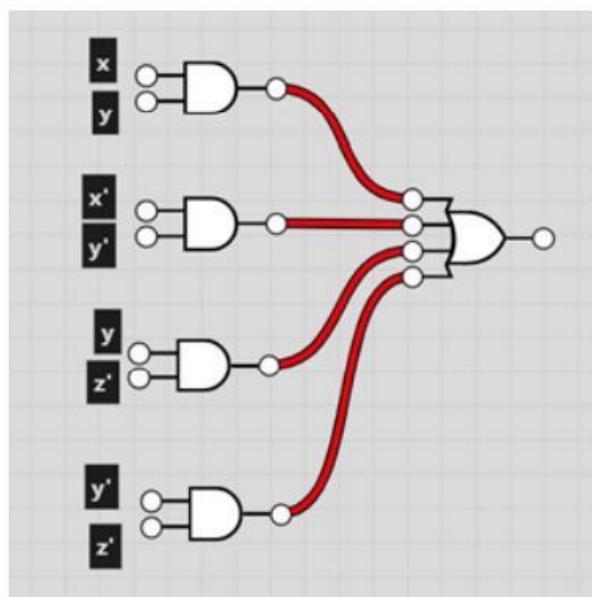
16. Pada relasi rekurens di bawah ini, tentukan apakah relasi rekurens tersebut homogen lanjor atau tidak.
- a. $a_n = (1.02)a_{n-1}$ **(nilai: 1) Homogen Lanjar**
 - b. $a_n = a_{n-1} \cdot a_{n-2}$ **(nilai: 1) Bukan homogen lanjar**
 - c. $a_n = 7a_{n/2} + a_{n-2}$ **(nilai: 1) Bukan homogen lanjar**
 - d. $a_n = a_{n-6}$ **(nilai: 1) Homogen Lanjar**
17. Dengan melakukan substitusi, periksalah apakah solusi dari relasi rekurens berikut ini sudah benar atau belum.
- a. $a_n = 2^n + 1$ adalah solusi dari relasi rekurens $a_n = 2a_{n-1} - 1$ dengan $a_1 = 3$ **(nilai: 1) Benar**
 - b. $a_n = 3^n + 2$ adalah solusi dari relasi rekurens $a_n = 3a_{n-1} + 2$ dengan $a_0 = 1$ **(nilai: 1) Salah**
 - c. $a_n = (1,045)^n \cdot a_0$ adalah solusi dari relasi rekurens $a_n = a_{n-1} + 0,045 \cdot a_{n-1}$ **(nilai: 1) Benar**
 - d. $a_n = 4^n$ adalah solusi dari relasi rekurens $a_n = 3a_{n-1} + 4a_{n-2}$ **(nilai: 1) Benar**
18. Terdapat relasi rekurens $a_n = -a_{n-2} + 2a_{n-1}$ dengan $a_0 = 1$ dan $a_1 = 2$.
- a. Persamaan karakteristik dari relasi tersebut adalah $r^2 - r - 2 = 0$. **(nilai: 1) Salah**
 - b. Akar-akar karakteristik dari relasi tersebut adalah $r_1 = 2$ dan $r_2 = -1$. **(nilai: 1) Salah**
 - c. Akar-akar karakteristik dari relasi tersebut adalah $r_0 = 1$ **(nilai: 1) Benar**
 - d. Solusi dari relasi rekurens tersebut adalah $a_n = 1 + n$. **(nilai: 1) Benar**

ALJABAR BOOLEAN

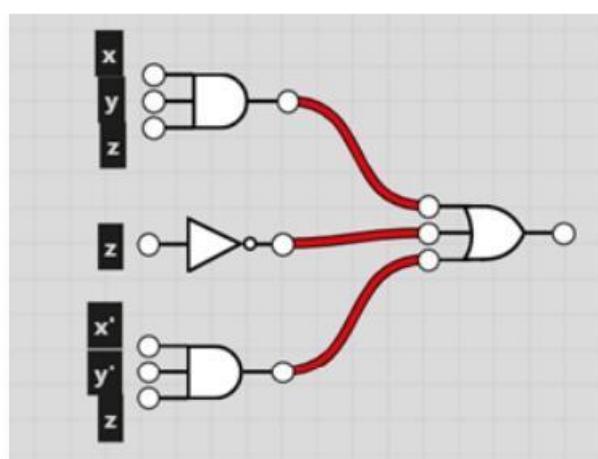
19. Diketahui fungsi boolean $f(w,x,y,z) = xz + wz' + xy' + w'y'z + x'yz'$. Fungsi tersebut adalah fungsi yang menerima masukan kode biner dari suatu digit desimal yang bernilai 1 jika dan hanya jika inputnya berupa:
- a. Bilangan ganjil
 - b. Bilangan prima
 - c. **Bilangan yang tidak habis dibagi 3**
 - d. Bilangan kelipatan 5
20. Bentuk POS fungsi boolean $F(x, y, z) = z' + xy'$ adalah:
- a. $\prod(0, 2, 4, 5, 6)$
 - b. **$\prod(1, 3, 7)$**
 - c. $\prod(0, 2, 3, 4, 6)$
 - d. $\prod(1, 3, 5, 7)$
21. Diketahui $f(x, y, z)$ yang menghasilkan nilai 1 jika inputnya bukan 3 atau 5 (desimal). Hasil penyederhanaan f dalam bentuk rangkaian logika adalah:



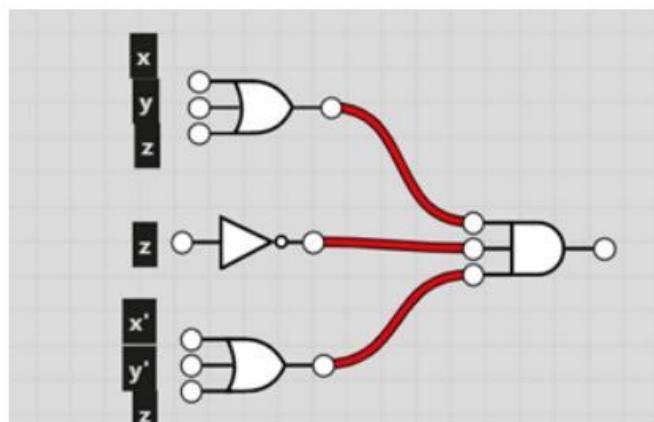
a.



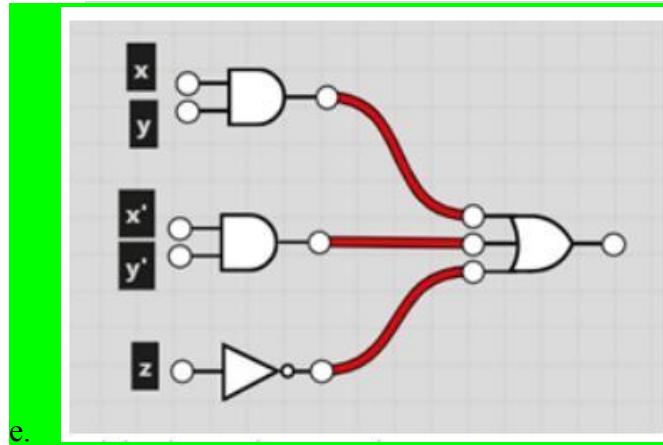
b.



c.



d.



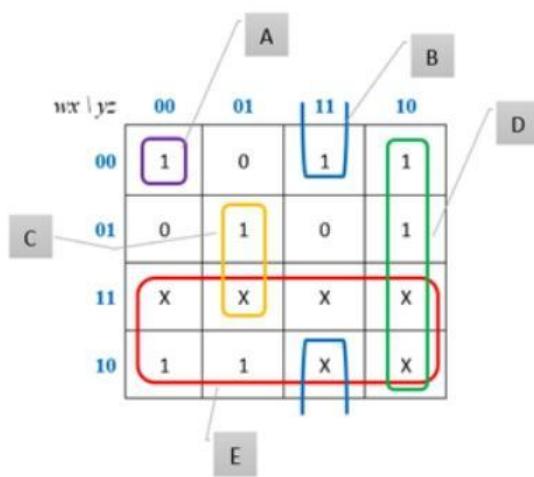
e.

- f. Tidak ada jawaban yang benar

22. Hasil penyederhanaan dari fungsi Boolean $f(w,x,y,z) = wxyz' + wxy'z' + wx'yz + wx'yz' + wx'y'z + w'xyz + w'xyz' + w'xy'z' + w'xy'z$ adalah

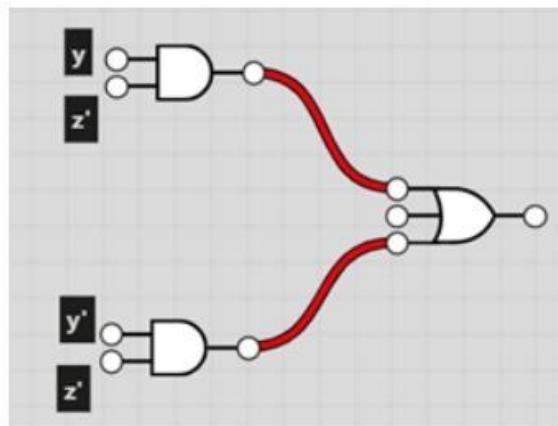
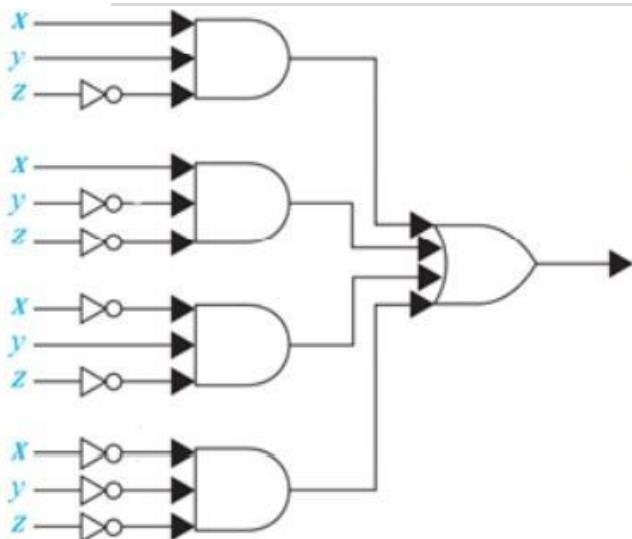
- a. $yz' + y'z' + wx'y + w'x$
- b. $wz' + w'z' + wx'yz + w'xy'z$
- c. $z' + wx'yz + w'xy'z$
- d. $z' + w'x + wx'y$**
- e. Tidak ada yang benar

23. Pengelompokan minterm yang kurang tepat adalah pada:

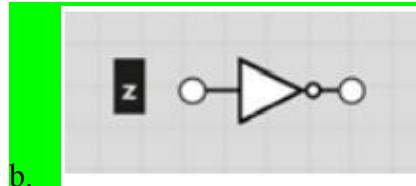


- a. A
- b. B
- c. C
- d. D
- e. A, C
- f. A, D
- g. A, B**
- h. C, D
- i. Tidak ada

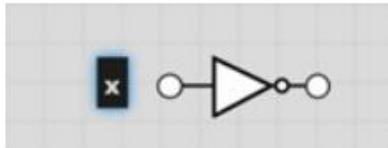
24. Rangkaian logika berikut dapat disederhanakan menjadi:



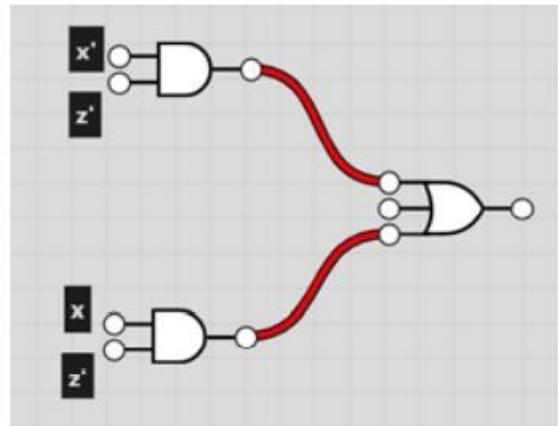
a.



b.



c.



d.

e. Tidak ada yang benar

25. Komplemen dari fungsi boolean $f(x,y,z) = x(y'z' + yz)$ adalah

- a. $\prod(0, 1, 2, 3, 5, 6)$
 b. $\sum(0, 1, 2, 3, 5, 6)$

- c. $\sum(4, 7)$
- d. $\prod(4, 6)$
- e. Tidak ada jawaban yang benar

UTS MATDIS 2021/2022

HIMPUNAN

1. Anggota suatu himpunan adalah objek di dalam himpunan tersebut. Untuk setiap himpunan-himpunan ini, tentukan himpunan sehingga 2 merupakan anggota dari himpunan.

- {x elemen R | x adalah bilangan bulat lebih besar 1 }
- {x elemen R | x adalah bilangan bulat kwadrat }
- {{2},{2, {2}}}
- {{{2}}}
- Jawaban tidak ada yang benar

2. Operasi selisih 2 himpunan A dan B diberi notasi A-B. Pernyataan yang benar dari operasi selisih adalah

- $\{A - B\} \cap A = \emptyset$
- $\{ A - B \} \cap B= \emptyset$
- $\{ A - B \} \cup A= B$
- $\{ A - B \} = \{ B - A \}$
- Tidak ada jawaban yang benar

3. Diantara bilangan bulat 1 sampai 70 (termasuk 1 dan 70 sendiri) berapa banyak bilangan yang TIDAK habis dibagi 2 atau 5 atau 7? Ini berarti mencari berapa banyak bilangan bulat 1 sampai 70 yang relatif prima dengan 70.

- 46
- 35
- 24
- 45
- Tidak ada jawaban yang benar

4. Misalkan A merupakan himpunan spesies yang bersayap dan B merupakan himpunan spesies endemik Indonesia. Dari 60 spesies yang diteliti, diketahui bahwa banyaknya spesies endemik Indonesia yang bersayap adalah sepertiga dari banyaknya spesies endemik Indonesia. Jika semua spesies yang diteliti berhasil dikelompokkan ke paling sedikit satu kelompok dan banyaknya spesies yang bersayap adalah 28, maka tentukanlah banyaknya spesies endemik Indonesia.

- 60
- 20
- 30
- 36
- 48
- Tidak ada jawaban yang benar

5. Dari soal di atas, berapa banyak spesies endemik Indonesia yang bersayap

- 16
 20
 28
 32
 30

Semua jawaban di atas tidak ada yang benar

RELASI DAN FUNGSI

6. Tentukan sifat relasi pada himpunan semua bilangan bulat berikut, $R = \{ (x,y) \mid x \geq y^2 \}$, apakah memiliki sifat refleksif, setangkup, menghantar, tolak setangkup

- Tidak refleksif, setangkup, tidak menghantar, tolak setangkup.
 Refleksif, tidak setangkup, menghantar, tidak tolak setangkup.
 Refleksif, setangkup, tidak menghantar, tolak setangkup.
 Tidak refleksif, tidak setangkup, menghantar, tolak setangkup
 Tidak ada jawaban yang benar

7. Matriks di bawah ini menyatakan relasi R pada $\{1, 2, 3\}$. Tentukan sifat relasi R apakah memiliki sifat refleksif, setangkup, menghantar, tolak setangkup

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

- Tidak refleksif, setangkup, tidak menghantar, tolak stangkup
 Refleksif, setangkup, menghantar, tidak tolak-setangkup
 Refleksif, setangkup, tidak menghantar, tolak setangkup
 Tidak refleksif, tidaksetangkup, menghantar, tolak setangkup
 Tidak ada jawaban yang benar

8. Diberikan beberapa relasi berikut : (i) $R_1 = \{ (a, b) \mid \text{jarak dari kota } a \text{ ke kota Bandung sama dengan jarak dari kota Bandung ke kota } b \}$ (ii) $R_2 = \{(a, b) \mid \text{jarak dari kota } a \text{ ke kota } b \text{ kurang dari } 100 \text{ km} \}$, dan (iii) $R_3 = \{(a, b) \mid a \text{ dan } b \text{ adalah himpunan tidak kosong yang irisannya bukan himpunan kosong } \}$. Tentukan yang manakah dari relasi-relasi tersebut merupakan relasi kesetaraan.

- R1 saja
 R1 dan R2
 R1, R2, dan R3
 R2 saja
 R3 saja
 R1 dan R3
 Semua jawaban salah

9. Diberikan tiga buah fungsi f, g, dan h yang masing-masing memetakan dari A ke A, dalam hal ini $A = \{1, 2, 3, 4\}$. Fungsi $f = \{(1, 2), (2, 1), (3, 1), (4, 4)\}$, $g = \{(1, 2), (2, 4), (3, 1), (4, 3)\}$, dan $h = \{(1, 1), (2, 3), (3, 1), (4, 3)\}$. Manakah dari ketiga fungsi tersebut memiliki balikan (invers)?

- f
- g
- h
- f dan g
- g dan h
- f dan h

10. Lanjutan soal sebelumnya. Tentukan hasil $f \circ h \circ g$

- $\{(1, 1), (2, 1), (3, 3), (4, 2)\}$
- $\{(1, 1), (2, 1), (3, 2), (4, 3)\}$
- $\{(1, 1), (2, 1), (3, 2), (4, 2)\}$
- $\{(1, 2), (2, 1), (3, 2), (4, 2)\}$
- $\{(1, 2), (2, 2), (3, 2), (4, 2)\}$
- Semua jawaban tidak ada yang benar

11. Fungsi $h : R \rightarrow Z$, dalam hal ini $h(x) = \lfloor x \rfloor + \lceil x \rceil$. Sifat fungsi h adalah:

- injektif, surjektif, bijektif
- injektif, tidak surjektif, tidak bijektif
- tidak injektif, surjektif, tidak bijektif
- tidak injektif, tidak surjektif, tidak bijektif
- Semua jawaban tidak ada yang benar

INDUKSI MATEMATIKA

12. Sebuah ATM hanya menyediakan uang pecahan Rp 20 ribuan dan Rp 50 ribuan. Kelipatan uang berapakah yang dapat dikeluarkan oleh ATM yang dapat digunakan sebagai basis pada teori induksi matematika

- ATM dapat mengeluarkan kelipatan (20 ribu) $n, n \geq 1$ yang dapat digunakan sebagai basis pada teori induksi matematika.
- ATM dapat mengeluarkan kelipatan (50 ribu) $n, n \geq 1$ yang dapat digunakan sebagai basis pada teori induksi matematika.
- ATM dapat mengeluarkan kelipatan (10 ribu) $n, n \geq 4$ yang dapat digunakan sebagai basis pada teori induksi matematika.
- ATM dapat mengeluarkan kelipatan (10 ribu) $n, n \geq 2$ yang dapat digunakan sebagai basis pada teori induksi matematika.
- Tidak ada jawaban yang benar

13. 2. Terdapat proposisi $p(n)$ yang menyatakan $1^2 + 2^2 + \dots + n^2 = n(n+1)(2n+1)/6$, untuk n anggota himpunan bilangan bulat positif.

Benar Salah

- a) Basis induksi dibuktikan untuk $n = 0$.

- b) Pada langkah induksi tahap awal, akan dibuktikan dengan mengubah bentuk persamaan proposisi menjadi:
- $$1^2 + 2^2 + \dots + n^2 + (n+1)^2 = n(n + 1)(2n + 1)/6 + (n+1)^2.$$

- c) Salah satu tahapan pembuktian pada langkah induksi menghasilkan bentuk:
- $$1^2 + 2^2 + \dots + n^2 + (n+1)^2 = (n+1)(2n^2 + 7n + 6).$$

- d) Dengan induksi matematika ditunjukkan bahwa proposisi $p(n)$ yang menyatakan $1^2 + 2^2 + \dots + n^2 = n(n + 1)(2n + 1)/6$, untuk n anggota himpunan bilangan bulat positif tidak terbukti.

14. Ingin dibuktikan sebuah proposisi $p(n)$ bahwa $2^n = 2$, untuk n anggota himpunan bilangan alami (natural number). Basis induksi untuk $n = 1$, maka $2^1 = 2$ adalah benar. Pada langkah induksi, terdapat asumsi bahwa untuk semua bilangan alami $n \leq k$ berlaku $2^n = 2$. Akan ditunjukkan bahwa untuk $n = k + 1$ juga berlaku $2^{k+1} = 2$, dengan langkah sebagai berikut.

- (i) $2^{k+1} = 2^k \cdot 2$
- (ii) $= 2^k \cdot (2^k / 2^{k-1})$
- (iii) $= 2 \cdot (2 / 2)$
- (iv) $= 2$

Benar Salah

- a) Contoh di atas adalah pembuktian dengan prinsip induksi kuat

- b) Basis induksi seharusnya dimulai dari $n = 0$.

- c) Pada pembuktian dengan induksi matematika, hipotesis atau asumsi yang ditarik pada langkah induksi pasti benar.

- d) Pembuktian proposisi melalui

induksi matematika harus membuktikan bahwa proposisi benar untuk basis induksi, dan proposisi juga benar pada langkah induksi.

15. Pada pembuktian proposisi di soal nomor 3, terdapat kesalahan pada:

Tandai satu oval saja.

a) langkah (i)

b) langkah (ii)

c) langkah (iii)

d) langkah (iv)

e) Tidak ada jawaban yang benar (tidak terdapat kesalahan langkah pembuktian).

REKURSIFITAS DAN RELASI REKURENS

16. Suatu relasi rekuren $a(n) = a(n-1) + 3$ dengan kondisi awal $a(1) = 2$, untuk $n = 1, 2, 3, \dots$. Dengan cara iterasi relasi rekuren diatas mulai dari $a(2) = 2 + 3$ kemudian $a(3) = (2+3) + 3$ dan seterusnya maka tentukan solusi relasi rekuren $a(n)$ dan tentukan nilai $a(n)$ untuk $n=10$ adalah

- Solusi relasi rekuren $a(n) = 2 + 3(n-1)$ dan nilai $a(10) = 29$.
- Solusi relasi rekuren $a(n) = 2 + 3(n-2)$ dan nilai $a(10) = 29$.
- Solusi relasi rekuren $a(n) = 2 + 3(n-1)$ dan nilai $a(10) = 32$.
- Solusi relasi rekuren $a(n) = 2 + 3(n-2)$ dan nilai $a(10) = 27$.
- Tidak ada jawaban yang benar

17. Tentukan solusi dari relasi rekurens berikut ini $a(n) = 8a(n-1) + 12a(n-2)$ untuk $n \geq 2$; $a(0) = 1$; $a(1) = 5$

- $a(n) = 6^n + 2^n$ (Ket: \wedge menyatakan operasi pangkat)
- $a(n) = 3^n + 5^n$ (Ket: \wedge menyatakan operasi pangkat)
- $a(n) = 6^n + 3^n$ (Ket: \wedge menyatakan operasi pangkat)
- $a(n) = 3^n + 2^n$ (Ket: \wedge menyatakan operasi pangkat)
- $a(n) = 5^n + 2^n$ (Ket: \wedge menyatakan operasi pangkat)

Semua jawaban tidak ada yang benar

18. Tentukan apakah relasi rekurens berikut ini adalah relasi rekurens homogen lanjor atau bukan

Ya Bukan

a) $a(n) = 2.55a(n-1)$

b) $a(n) = a(n-1) + 2n$

c) $a(n) = 3a(n-3) + 2a(n-2)$

d) $a(n) = 5a(n-1) - 6a(n-2) + 7^n$

19. Terdapat relasi rekurens sebagai berikut: $a(n) = 2a(n-1) + 2^n$.

- a) $a(n) = 2^n$ adalah solusi dari relasi rekurens tersebut.
- b) $a(n) = n2^n$ adalah solusi dari relasi rekurens tersebut.
- c) $a(n) = n^2$ adalah solusi dari relasi rekurens tersebut.
- d) $a(n) = 2(n^2)$ adalah solusi dari relasi rekurens tersebut.
- e) Solusi dari relasi rekurens tersebut tidak dapat ditentukan.

20. Prediksi banyaknya ikan yang ditangkap pada suatu tahun adalah ratarata banyaknya ikan yang ditangkap pada tahun sebelumnya dan dua tahun sebelumnya. Jika ikan yang ditangkap pada tahun pertama adalah 744 ton dan pada tahun kedua adalah 1128 ton, maka:

Benar Salah

a) prediksi
banyaknya
ikan yang
ditangkap
pada suatu
tahun dapat
dimodelkan
dengan relasi
rekurens
homogen
lanjut derajat k
= 1.



b) Solusi dari
persoalan
prediksi
banyaknya
ikan yang
ditangkap
pada suatu
tahun tidak
dapat dicari
secara
sistematis
dengan
memanfaatkan
persamaan
karakteristik.



c) Prediksi
banyaknya
ikan yang
ditangkap
pada tahun
ke-9 adalah
999.



d) Untuk

mencari solusi
relasi rekurens
homogen
lanjut dengan
derajat k = 2
dengan akar
kembar, kita
cukup
mengetahui
relasi
rekurensnya
saja.

ALJABAR BOOLEAN

21. Tentukan bentuk kanonik Sum-of-Product (SOP) dari fungsi boolean $f(x, y, z) = (x + z)y$

- Sigma(1, 4, 7)
- Sigma(1, 3, 7)
- Sigma(3, 4, 7)
- Sigma(3, 6, 7)
- Tidakada jawaban yang benar

22. Diberikan peta Karnaugh dari fungsi boolean $f(w, x, y, z)$ dengan nilai X adalah don't care. Hasil penyederhanaan fungsi dalam bentuk baku SOP adalah:

wx \ yz	00	01	11	10
00	x	0	1	x
01	1	1	1	1
11	x	0	x	0
10	x	0	1	x

- A) $f(w, x, y, z) = w'x + yz + x'y$
- B) $f(w, x, y, z) = w'x + yz$
- C) $f(w, x, y, z) = w'x + x'y$
- D) Semua jawaban di atas benar
- E) Jawaban B dan C benar
- Semua jawaban salah

23. Diberikan peta Karnaugh dari fungsi boolean $f(w, x, y, z)$ dengan nilai X adalah don't care. Hasil penyederhanaan fungsi dalam bentuk baku POS adalah:

wx \ yz	00	01	11	10
00	0	x	1	0
01	x	x	x	1
11	0	0	0	0
10	0	x	1	x

- f(w, x, y, z) = $y'(w + x)(x' + z')$
- f(w, x, y, z) = $y(w' + x')(w + x + y + z')$
- f(w, x, y, z) = $y(w' + x')(x + z)$
- f(w, x, y, z) = $y'(w' + x')(w' + x' + y' + z)$
- Semua jawaban di atas tidak ada yang benar

24. Terdapat instruksi dalam sebuah program yang menerima 2 variabel A dan B, dan masing-masing panjangnya 2 bit

(misalkan $a[1]a[2]$ dan $b[1]b[2]$). Jika bilangan desimal yang direpresentasikan oleh variabel A dijumlahkan dengan bilangan desimal yang direpresentasikan oleh variabel B, rangkaian logika akan menghasilkan keluaran 1 jika hasil penjumlahan adalah bilangan ganjil, dan menghasilkan keluaran 0 jika tidak. Tentukan apakah pernyataan berikut ini benar atau salah.

Benar Salah

a) Jika input variabel A adalah 01 dan input variabel B adalah 11, maka rangkaian logika akan memberikan hasil 1.

b) Jika input variabel A adalah 10 dan input variabel B adalah 10, maka rangkaian logika akan memberikan hasil 1.

c) Terdapat 8 sel pada Peta Karnaugh yang memiliki nilai 1.

d) Rangkaian logika dari fungsi yang sudah disederhanakan untuk persoalan ini

bisa dibentuk hanya dengan gerbang AND dan gerbang OR saja.

25. Pada persoalan yang sudah disebutkan di soal nomor (4), fungsi yang sudah disederhanakan adalah:

- a) $f(a[1],a[2],b[1],b[2]) = (a[2]' + b[2]) \cdot (a[2] + b[2]')$
- b) $f(a[1],a[2],b[1],b[2]) = (a[2]b[2]') + (a[2]'b[2])$
- c) $f(a[1],a[2],b[1],b[2]) = (a[2]'b[2]) + (a[2]b[2]')$
- d) $f(a[1],a[2],b[1],b[2]) = (a[1]a[2]'b[2]) + (a[1]'a[2]b[2]) + (a[2]b[1]'b[2]') + (a[2]b[1]b[2])$
- Tidak ada jawaban yang benar.

UAS MATEMATIKA DISKRIT

UAS IF2153 Matematika Diskrit

UAS SEMESTER I – 2007/2008

Hari/Tanggal: Kamis, 4 Januari 2008

Dosen: Harlili, M.Sc. (K-01), Ir. Rinaldi Munir, M.T (K-02)

Waktu: 120 menit

Soal dari Rinaldi Munir

1. Sebuah algoritma rekursif memproses sebuah tabel yang berukuran n (asumsi: n merupakan perpangkatan dari 2, yaitu $n = 2^k$). Tabel dibagi 2 secara rekursif dan setiap bagian diproses lagi dengan algoritma yang sama. Begitu seterusnya sampai ukuran tabel = 1. Kompleksitas waktu algoritma tersebut adalah: (10)

$$T(n) = \begin{cases} 1 & , n = 1 \\ 2T(n/2) + n & , n > 1 \end{cases}$$

\square

Nyatakan kebutuhan waktu algoritma tersebut dalam notasi asimptotik (O -besar).

2. Berapa kali instruksi $x \leftarrow x + 1$ pada potongan algoritam di bawah ini dikerjakan?

Nyatakan kompleksitas waktunya dalam notasi O -Besar, Ω -Besar, dan Θ -besar.

```
i ← n
while i ≥ 1 do
    x ← x + 1
    i ← ⌊i/2⌋
end
```

(15)

3. Ada 8 zat kimia berupa (A, B, ..., H), beberapa diantaranya tidak dapat disimpan bersama-sama di dalam satu ruangan karena campuran uapnya dapat menimbulkan reaksi kimia yang eksplosif, yaitu:

Zat kimia A tidak dapat disimpan bersama-sama dengan C, E, dan H

Zat kimia B tidak dapat disimpan bersama-sama dengan A, D, dan F

Zat kimia C tidak dapat disimpan bersama-sama dengan A, G, dan H

Zat kimia D tidak dapat disimpan bersama-sama dengan B, E, dan H

Zat kimia E tidak dapat disimpan bersama-sama dengan D, A, dan H

Zat kimia F tidak dapat disimpan bersama-sama dengan B, G, dan H

Zat kimia G tidak dapat disimpan bersama-sama dengan C, dan F

Zat kimia H tidak dapat disimpan bersama-sama dengan A, C, D, E, dan F

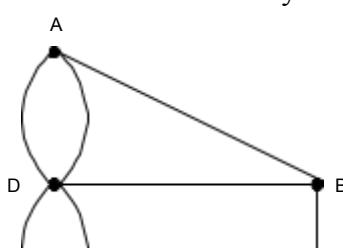
- (a) Gambarkan graf yang merepresentasikan kondisi di atas, jelaskan arti setiap simpul dan sisi.

(5)

- (b) Berapa minimal jumlah ruangan yang dibutuhkan untuk menyimpan zat kimia tersebut? Tuliskan zat-zat kimia apa saja yang bersama-sama ditempatkan pada setiap ruangan.

(10)

4. Graf yang mewakili masalah jembatan Konigsberg adalah seperti di bawah ini, yang dalam hal ini simpul menyatakan daratan dan sisi menyatakan ruas jembatan.



C

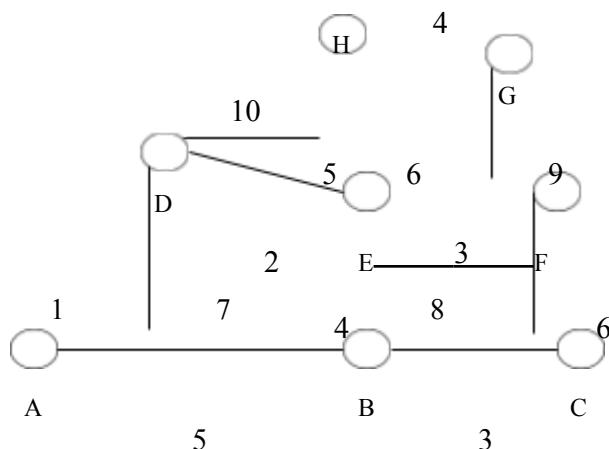
Berapa paling sedikit ruas jembatan baru yang perlu ditambahkan agar setiap jembatan dapat dilalui tepat sekali? Gambarkan graf yang baru setelah penambahan jembatan baru tersebut. (10)

Soal dari Harlili

5. Seorang profesor membuat 20 soal pilihan ganda Matematika Diskrit, setiap soal mempunyai pilihan jawaban a, b, c, dan, d. Jika soal yang mempunyai jawab a ada 8 soal, mempunyai jawab b ada 3 soal, mempunyai jawab c ada 4 soal, dan mempunyai jawab d ada 5 soal, berapa banyak kunci jawaban yang mungkin, jika urutan soal tidak diperhatikan. (10)
6. Tentukan berapa banyak solusi bilangan bulat nonnegatif dari $x_1 + x_2 + x_3 + x_4 + x_5 = 21$, jika $0 \leq x_i \leq 10$ (10)
7. Carilah pohon merentang minimum dari graf di bawah dengan menggunakan Algoritma Kruskal. (15)

(Jawaban harus menunjukkan langkah-langkah pembuatannya dengan mengisi tabel ini)

Langkah	Sisi	Bobot	Pohon Merentang
1



8. Tentukan kode Huffman untuk enkoding setiap simbol yang frekuensi setiap simbol diberikan berikut ini;

A: 0,08 B: 0,10 C: 0,12 D: 0,15 E: 0,20 F: 0,35.

Gambarkan pula pohon Huffman yang terbentuk. Berapa rata-rata banyaknya bit yang digunakan untuk enkoding satu karakter ini? (15)

Bonus:

9. Apa perkiraan nilai anda untuk mata kuliah ini? (A/B/C/D). D = E = Tidak lulus. (2)

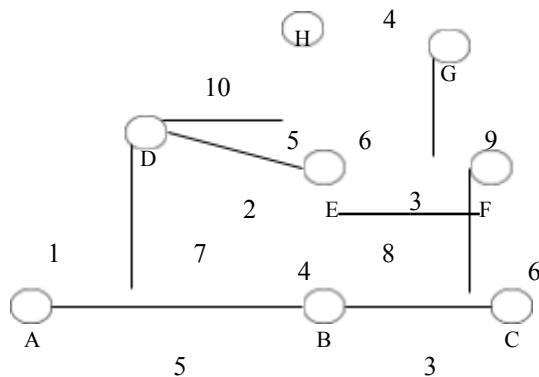
SELAMAT BERPIKIR dan BEKERJA

UAS SEMESTER I – 2009/2010
IF2091 Struktur Diskrit
Dosen: Harlili (K1) dan Rinaldi Munir (K2)
Senin, 21 Desember 2009
Waktu: 100 menit

Soal dari Harlili

1. Sebuah peubah (variabel) di dalam bahasa pemrograman harus berupa sebuah huruf atau sebuah huruf diikuti dengan sebuah angka. Berapa banyak nama peubah yang dapat dibuat jika:
 - a. Huruf kapital dan huruf kecil dibedakan ?
 - b. Huruf kapital dan huruf kecil tidak dibedakan ?(10)
2. Ujian struktur diskrit terdiri dari 10 soal. Berapa banyak cara memberi bobot nilai pada setiap soal jika jumlah semua 10 soal benar adalah 100 dan setiap soal berbobot nilai paling sedikit 5? (15)
3. Carilah pohon merentang dari graf di bawah dengan menggunakan Algoritma Prim.
 (Jawaban harus menunjukkan langkah-langkah pembuatannya dengan mengisi tabel ini) (12,5)

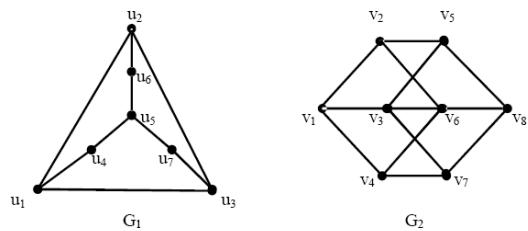
Langkah	Sisi	Bobot	Pohon Merentang
1



4. Tentukan kode Huffman untuk enkoding setiap simbol yang frekuensi setiap simbol diberikan berikut ini; A: 0,20 B: 0,10 C: 0,15 D: 0,25 E: 0,30
 Gambarkan pula pohon Huffman yang terbentuk. Berapa rata-rata banyaknya bit yang digunakan untuk enkoding satu karakter ini? (12,5)

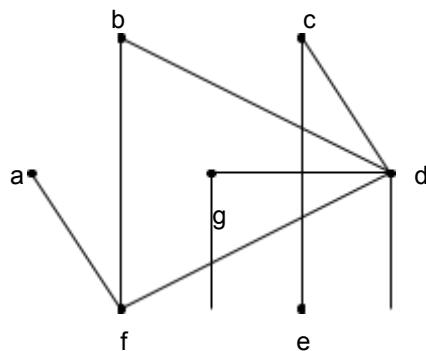
Soal Rinaldi Munir

5. Mana dari kedua graf di samping ini yang merupakan graf *bipartite*? Tentukan mana himpunan simpul V_1 dan mana simpul V_2 dari graf *bipartite* tersebut, lalu gambarkan graf *bipartite* alternatif dengan himpunan V_1 di bagian atas dan himpunan V_2 di bagian bawah. (10)



6. Gunakan Teorema Kuratowski untuk menentukan apakah graf di samping ini planar.

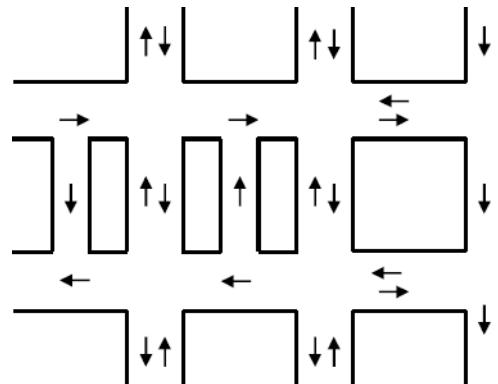
(15)



7. Di samping ini adalah gambar jaringan lalu lintas. Arah panah menunjukkan bahwa jalan dapat dilalui sesuai arah tersebut.

- (a) Jika dimodelkan sebagai graf, simpul menyatakan apa, busur menyatakan apa, kemudian gambarkan graf yang merepresentasikan jaringan jalan tersebut
- (b) Apakah setiap arah jalan dapat dilalui sekali saja? Mengapa?

(10)



8. Algoritma Prim adalah algoritma untuk mencari pohon merentang minimum (*minimum spanning tree*) T di dalam sebuah graf berbobot. Algoritma ini bekerja secara iteratif. Pada setiap iterasi, pilih sisi e dari graf $G(V, E)$ yang mempunyai bobot terkecil dan bersisian dengan simpul-simpul di T asalkan e tidak membentuk sirkuit di T . Secara garis besar algoritma Prim adalah sebagai berikut:

```

function Prim(input E : himpunan_sisi) → himpunan_sisi
{ menghasilkan pohon merentang minimum}
Deklarasi
    T : himpunan_sisi
    e : sisi
Algoritma
    T ← e {sisi yang mempunyai bobot terkecil di dalam E}
    for i ← 1 to n-1 do
        e ← sisi yang belum terpilih dan mempunyai bobot terkecil di dalam E
        dan bersisian dengan simpul di T
        T ← T ∪ {e} {masukkan e ke dalam pohon merentang T}
    endfor
    return T
  
```

- (a) Berapa kompleksitas (dalam notasi O) untuk pernyataan:

$e \leftarrow$ sisi yang belum terpilih dan mempunyai bobot terkecil di dalam E
dan bersisian dengan simpul di T }

- (b) Berapa kompleksitas (dalam notasi O) untuk algoritma Prim secara keseluruhan. (15)

9. Perkiraan anda untuk nilai kuliah ini
(A/AB/B/BC/C/D/E) (2)

UAS SEMESTER I – 2011/2012

IF2091 Struktur Diskrit

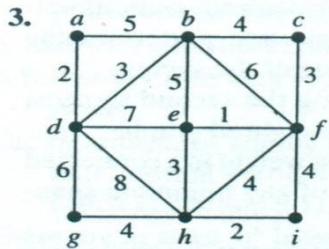
Hari/Tanggal: Senin, 12 Desember 2011, Waktu: 120 menit
Dosen: Harlili, M.Sc. (K-02), Dr. Ir. Rinaldi Munir, M.T (K-01)*Berdo'alah terlebih dahulu sebelum mengerjakan ujian ini. Selalu bersikap jujur dalam ujian.*

Soal dari Dra. Harlili, M.Sc.

1. Berapa banyak solusi bilangan bulat dari $x_1 + x_2 + x_3 + x_4 + x_5 = 21$ jika $0 \leq x_i \leq 10$? (10)
 2. Tentukan kode Huffman untuk enkoding setiap simbol yang frekuensi setiap simbol diberikan dibawah ini.
 $A: 0,20 \quad B: 0,10 \quad C: 0,15 \quad D: 0,25 \quad E: 0,30$

Gambarkan pula pohon Huffman yang terbentuk. Berapa rata-rata banyaknya bit yang digunakan untuk enkoding satu karakter ini? (15)

3. Carilah pohon merentang minimum dari graf (3) dibawah ini dengan menggunakan Algoritma Kruskal, serta tuliskan setiap langkahnya. (15)

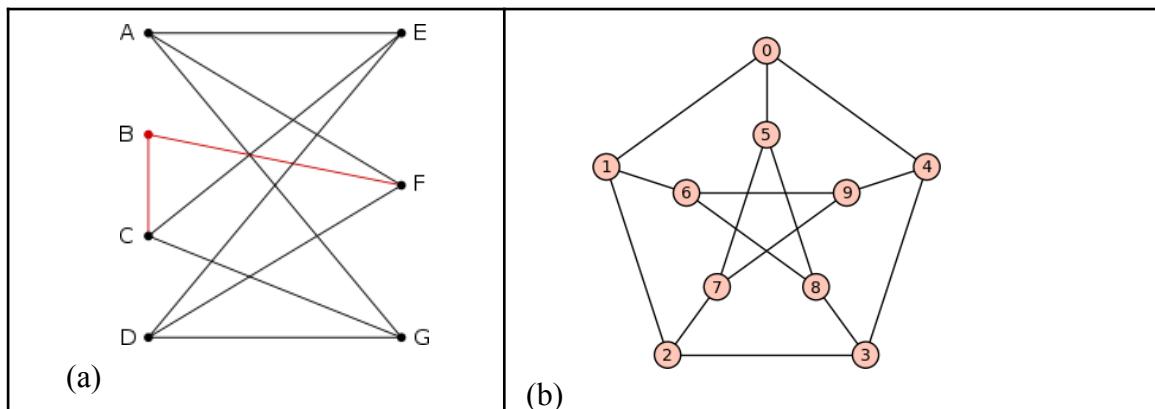


4. Buatlah suatu pohon berakar terurut berdasarkan hasil penelusuran *preorder* simbol dibawah ini:

$a, b, f, c, g, h, i, d, e, j, k, l$
 dengan ketentuan a mempunyai 4-anak, c mempunyai 3-anak, j mempunyai 2-anak, b dan e masing-masing mempunyai 1-anak, dan simpul lainnya adalah daun. (10)

Soal dari Dr. Ir. Rinaldi Munir

5. Gunakan Teorema Kuratowski untuk memperlihatkan apakah graf di bawah ini planar atau tidak planar.
 (15)



6. Dinas Komunikasi dan Informasi (Diskominfo) sebuah kota memonitor semua stasiun radio untuk memastikan bahwa sinyal radio mereka tidak mengganggu satu sama lain. Diskominfo mencegah interferensi dengan menetapkan frekuensi yang tepat untuk masing-masing stasiun. Anda akan diminta menentukan berapa minimal jumlah frekuensi yang berbeda yang dibutuhkan untuk enam stasiun yang terletak pada jarak yang ditunjukkan dalam tabel, jika dua stasiun tidak dapat menggunakan saluran yang sama ketika mereka berada dalam 150 mil satu sama lain.

	KQAA	KQBB	KQCC	KQDD	KQEE	KQFF
KQAA	-	25	202	77	375	106
KQBB	25	-	175	51	148	222
KQCC	202	175	-	111	365	411
KQDD	77	51	111	-	78	297
KQEE	375	148	365	78	-	227
KQFF	106	222	411	297	227	-

- (a) Gambarkan graf yang merepresentasikan persoalan ini. Simpul merepresentasikan apa, sisi merepresntasikan apa? (5)
- (b) Termasuk jenis persoalan apakah ini? Berapa minimal jumlah frekuensi yang berbeda yang dibutuhkan untuk enam stasiun yang terletak pada jarak yang ditunjukkan di dalam tabel, jika dua stasiun tidak dapat menggunakan saluran yang sama ketika mereka berada dalam 150 mil satu sama lain? (5)
7. (a) Berapakah nilai kompleksitas waktu asimptotik dalam notasi *O*-Besar untuk
- $$T(n) = \begin{cases} k, & n = 1 \\ k + k^2 T(n-1), & n > 1 \end{cases}$$
- +
1
;
- (b) Berikan estimasi Big-*O* untuk $T(n) = (2n + 5) \log(n!) + 3n^2$ (5)
8. Diberikan potongan sebuah algoritma dalam notasi Pascal-like sbb:
- ```
i := 2
w
h
i
l
e
i
≥
1
d
o
b
e
g
i
n
f
o
r
j
:
=
1
t
o
n
d
o
x
:
=
x
```
- {  
e  
n  
d  
f  
o  
r  
}  
i  
:  
=
- (a) Hitung berapa kali pernyataan  $x := x + 1$  dieksekusi (7,5)
- (b) Nyatakan hasil jawaban a di atas dalam notasi *O*-besar, omega-besar, dan theta-besar. (5)

9. Apa perkiraan nilai anda untuk mata kuliah ini?

(A/AB/B/BC/C/D/E)

(2)

(7,5)

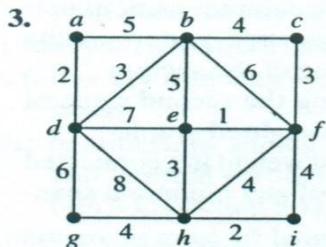
Program Studi Teknik Informatika  
 Sekolah Teknik Elektro dan Informatika  
 Institut Teknologi Bandung

UAS SEMESTER I – 2012/2013  
*IF2091 Struktur Diskrit*  
 Hari/Tanggal: Rabu, 19 Desember 2012, Waktu: 120 menit  
 Dosen: Harlili, M.Sc. dan Dr. Ir. Rinaldi Munir, M.T

*Berdo'alah terlebih dahulu sebelum mengerjakan ujian ini.*

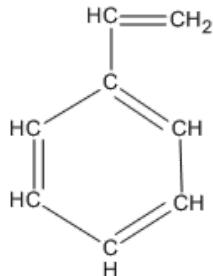
Soal dari Dra. Harlili, M.Sc.

1. Berapa banyak cara memilih suatu tim kurikulum pada suatu prodi ITB, jika tim kurikulum tersebut terdiri atas 3 anggota Kelompok Keahlian A (KK-A) dan 4 anggota KK-B, jika KK-A mempunyai 9 anggota dan KK-B mempunyai 11 anggota. (10)
2. Tentukan berapa banyak bilangan bulat positif lebih kecil dari 1.000.000 yang mempunyai tepat satu digit sama dengan 9 dan jumlah dari digit-digitnya sama dengan 19. (10)
3. Tentukan kode Huffman untuk enkoding setiap simbol dari string informaaai dan gambarkan pula pohon Huffman yang terbentuk. Berapa rata-rata banyaknya bit yang digunakan untuk enkoding satu karakter ini? (20)
4. Carilah pohon merentang minimum dari graf (3) dibawah ini dengan menggunakan Algoritma Prim, serta tuliskan setiap langkahnya. (10)



Soal dari Dr. Ir. Rinaldi Munir

5. Senyawa aromatik (siklik) pada hidrokarbon memiliki karakteristik *bipartite graph*. Gambarkan graf yang merepresentasikan senyawa *Styrene* di bawah ini (simpul menyatakan apa dan sisi menyatakan apa), kemudian perlihatkan (dengan menggambar ulang grafnya) bahwa senyawa tersebut adalah graf bipartit! (15)



6. Pada suatu hari, Martin si penyayang hewan membeli 6 jenis hewan peliharaan yang berbeda. Hewan tersebut adalah anjing, kucing, burung, ular, tikus, dan ikan. Beberapa hewan tidak bisa ditempatkan di dalam satu ruangan yang sama, atau mereka akan saling memakan atau dimakan, yang dapat dilihat pada tabel berikut :

| Nama Hewan | Tidak bisa ditempatkan dengan |
|------------|-------------------------------|
| Anjing     | Kucing, burung, ular          |
| Kucing     | Anjing, burung, tikus, ikan   |
| Burung     | Anjing, kucing, ular, ikan    |
| Ular       | Anjing, burung, tikus         |

|       |                       |
|-------|-----------------------|
| Tikus | Kucing, ular, ikan    |
| Ikan  | Kucing, burung, tikus |

Namun untuk menghemat biaya, Martin diharapkan dapat menempatkan enam binatang tersebut ke dalam ruangan sesedikit mungkin. Modelkan dan jelaskan permasalahan ini ke dalam bentuk graf, dan tentukan ruangan minimum yang dibutuhkan oleh Martin. (15)

7. Urutkan kompleksitas waktu asimptotik berikut mulai dari yang paling mangkus:  
 $O(n)$ ,  $O(9^n)$ ,  $O(n^9)$ ,  $O(n!)$ ,  $O(1)$ ,  $O(9 \log n)$  (5)
8. Hitunglah kompleksitas algoritma waktu di bawah ini (dalam bentuk  $T(n)$ ) jika melihat banyaknya operasi  $x = x + 2$ , lalu nyatakan hasilnya dalam notasi O-besar. Selanjutnya hitunglah nilai  $x$  setelah algoritma selesai.  
(15)

```
int x = 0;
int n =
2010
for (int i = 1; i <= n; i++)
{
for (int j = 1; j <= i; j++)
{
for (int k = j; k <= n; k++)
{
x = x + 2;
}
}
}
return x;
```

9. Apa perkiraan nilaimu untuk mata kuliah ini? (A/AB/B/BC/C/D/E) (2)

Program Studi Teknik Informatika  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung

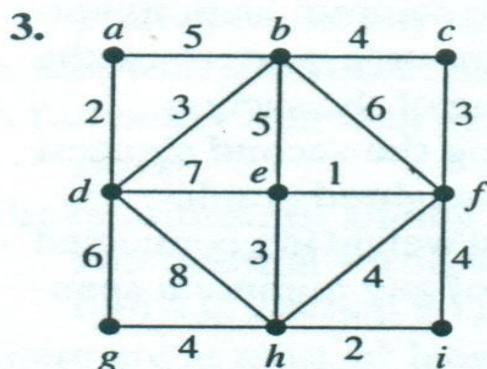
**UAS IF2120 Matematika Diskrit**

**UAS SEMESTER I – 2013/2014**

Hari/Tanggal: Selasa, 17 Desember 2013, Waktu: 120 menit  
Dosen: Harlili, M.Sc. (K-02), Dr. Ir. Rinaldi Munir, M.T (K-01)

Soal dari Dra. Harlili, M.Sc.

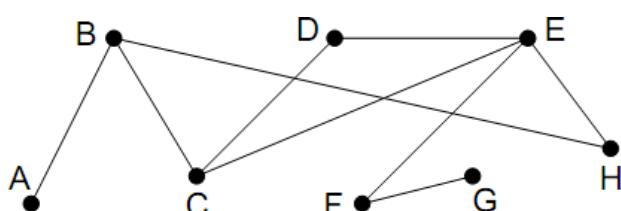
1. Berapa banyak bilangan bulat positif lebih kecil dari 1.000.000 mempunyai tepat satu digit = 9 dan jumlah semua digit-digitnya = 19?  
(Nilai: 10)
2. Tentukan kode Huffman untuk enkoding setiap simbol dari: mahasiswa informatika. Gambarkan pula pohon Huffman yang terbentuk. Berapa rata-rata banyaknya bit yang digunakan untuk enkoding satu karakter ini?  
(Nilai: 15)
3. Carilah pohon merentang minimum dari graf (3) dibawah ini dengan menggunakan Algoritma Prim, serta tuliskan setiap langkahnya.  
(Nilai: 10)



4. Tentukan nilai dari ekspresi *prefix* sbb: + - \*  
 $235 / \uparrow 234$  dan ekspresi *postfix* sbb:  
 $723 * -4 \uparrow 93 / + .$   
(Nilai: 10)

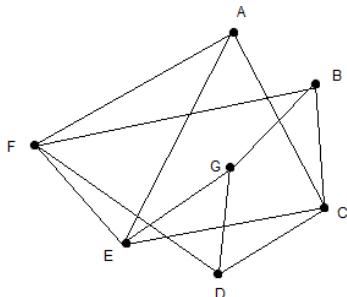
Soal dari Dr. Ir. Rinaldi Munir

5. Gambarkan sebuah graf sederhana tak-terhubung G yang memiliki 6 simpul dan 10 sisi.  
(Nilai: 10)
6. Graf di bawah ini untuk soal a dan b



- (a) Berapa bilangan kromatis graf tersebut?  
 (b) *Jarak* antara dua buah simpul di dalam graf didefinisikan sebagai panjang lintasan dengan jumlah sisi paling sedikit di antara kedua simpul tersebut. Misalnya jarak antara simpul A dan E adalah 3. *Diameter* sebuah graf adalah jarak antara dua buah simpul yang paling jauh. Tentukan diameter graf di atas, antara simpul manakah itu?  
 (Nilai: 5 + 5)

7. Diketahui graf sebagai berikut, buktikan dengan teorema Kuratowski bahwa graf tersebut tidak planar



(Nilai: 10)

8. Tentukan notasi-notasi big-O,  $\Omega$ , dan  $\Theta$  untuk :  $T(n) = 7n^3 + 5n^2 \log n$  (Nilai: 5)

9. Tentukan nilai kompleksitas waktu asimptotik dalam notasi O-Besar untuk fungsi berikut:

$$T(n) = (5n+1) \log (n!) + 3n^2 \quad (\text{Nilai: 5})$$

10. Terdapat sebuah algoritma dalam notasi algoritmik seperti dibawah ini:

```

procedure X(input/output s : array [1..n] of integer)
Deklarasi
 i, j, imaks, temp : integer
Algoritma:
 for i from 1 to n do imaks := 1
 for j from 2 to i do
 if s[j] > s[imaks] then
 imaks := j
 endif
 endfor
 temp := s[i]
 s[i] := s[imaks]
 s[imaks] := temp
 endfor

```

- (a) Algoritma apakah itu? (Nilai: 5)  
 (b) Tentukan  $T(n)$  dan kompleksitas algoritma dalam notasi Big-O. (Nilai: 10)

11. Apa perkiraan nilaimu untuk mata kuliah ini? (A/AB/B/BC/C/D/E) (Nilai: 2)

## UAS IF2120 Matematika Diskrit (3 SKS)

UAS SEMESTER I – 2014/2015

Dosen: Rinaldi Munir, Harlili

Kamis, 11 Desember 2014

Waktu: 120 menit

Setiap soal nilainya = 10, kecuali nomor 6 nilainya = 2.

Soal dari Rinaldi Munir

1. (Graf) Buktikan dengan induksi matematik bahwa sebuah graf lengkap dengan  $n$  simpul memiliki  $n(n - 1)/2$  buah sisi.

- (a) Apakah graf tersebut graf Euler?
- (b) Jika bukan graf Euler, tambahkan beberapa buah sisi ganda sehingga menjadi graf Euler
- (c) Berapa minimal sisi ganda yang dibutuhkan untuk membuatnya menjadi graf Euler?

Relasi rekurens) Barisan Fibonacci memiliki rumus  $F(n) = F(n - 1) + F(n - 2)$ ;  $F(0) = 0$ ,  $F(1) = 1$ . yatakan rumus Fibonacci dalam rumus yang tidak mengandung relasi rekurens, lalu hitunglah engan rumus tersebut barisan Fibonacci ke-20.

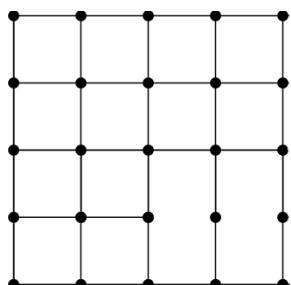
Kompleksitas algoritma) Di bawah ini adalah algoritma untuk menghitung polinom  $p(x) = a_nx^n + a_{n-1}x^{n-1} + \dots + a_1x + a_0$  pada  $x = t$ . Hitung  $T(n)$ , yaitu banyaknya operasi perkalian (\*) yang dilakukan di dalam algoritma, lalu nyatakan kompleksitas waktu asimptotik  $T(n)$  dalam notasi Big- $h$ .

```

 $p \leftarrow 0$
for $i \leftarrow n$ downto 0 do
 $pangkat \leftarrow 1$
 for $j \leftarrow 1$ to i do {hitung x^i }
 $pangkat \leftarrow pangkat * t$
 endfor
 $p \leftarrow p + a[i] * pangkat$
endfor
return p

```

2. (Graf) Perhatikan graf di bawah ini:



3. ( )  
N  
d

4. ( )  
a  
d  
 $O$

5. (Kompleksitas Algoritma) Diberikan enam buah fungsi kompleksitas waktu:

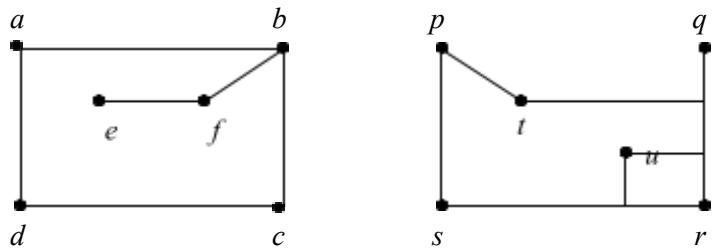
$$\begin{array}{lll} g_1(n) = \log \log n; & g_2(n) = \log(n!); & g_3(n) = n!; \\ g_4(n) = 2^{\log n}; & g_5(n) = n^{4/3}; & g_6(n) = n \log n; \end{array}$$

Urutkan fungsi tersebut dalam urutan yang menaik berdasarkan laju pertumbuhannya.

6. Apa prediksi nilaimu untuk kuliah ini? (A/AB/B/BC/C/D/E)

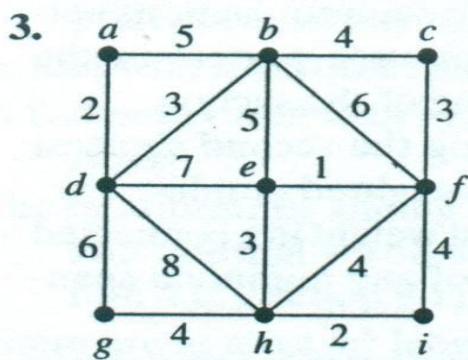
Soal dari Harlili

7. Tentukan apakah kedua graf ini isomorfik. Jawaban harus ada matriks ketetanggaan (*adjacent*) dari kedua graf tersebut.



8. Diberikan masukan berupa rangkaian karakter dengan urutan sebagai berikut: P, T, B, F, H, K, N, S, A, U, M, I, D, C, W, O
- Gambarkan pohon pencarian biner yang terbentuk.
  - Berapa perbandingan yang diperlukan untuk mencari karakter A.
9. Gunakan algoritma Prim untuk mencari pohon rentang minimum. Jawaban dengan mengisi tabel sbb.

| Langkah | Sisi | Bobot | Pohon Merentang |
|---------|------|-------|-----------------|
| 1       | ...  | ...   | ...             |



10. Tentukan kode Huffman untuk enkoding setiap simbol yang frekuensi setiap simbol diberikan dibawah ini. Gambarkan pula pohon Huffman yang terbentuk. Berapa rata-rata banyaknya bit yang digunakan untuk enkoding satu karakter ini?
- A: 0,10   B: 0,25   C: 0,05   D: 0,15   E: 0,30   F: 0,07   G: 0,08
11. Berapa banyak daun pada pohon 3-ary penuh dengan 100 simpul?

## UAS IF2120 Matematika Diskrit (3 SKS)

UAS SEMESTER I – 2016/2017

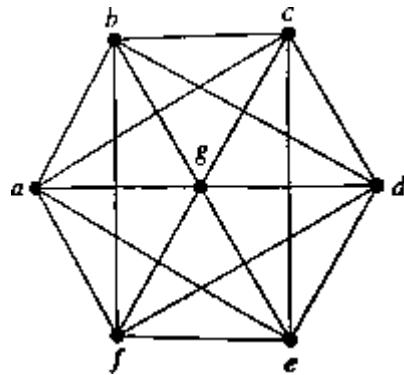
Dosen: Rinaldi Munir, Harlili

Kamis, 5 Desember 2016

Waktu: 100 menit

**Setiap soal nilainya = 12,5, kecuali nomor 9 nilainya = 2.****Soal dari Rinaldi Munir**

1. (a) Gunakan Teorema Fermat untuk menghitung  $3^{302} \bmod 5$ ,  $3^{302} \bmod 7$ , dan  $3^{302} \bmod 11$   
 (b) Gunakan hasil dari (a) dan *Chinese Remainder Theorem* untuk menghitung nilai  $3^{302} \bmod 385$   
 (catatan:  $385 = 5 \cdot 7 \cdot 11$ )
2. Gunakan Teorema Kuratowski untuk menentukan apakah graf di bawah ini planar.



3. Sepotong algoritma disajikan di bawah ini:

```

j ← n
sum ← 0
while j ≥ 1 do
 for i ← 1 to j do
 sum ← sum + a[i]
 endfor
 j ← j div 2
endwhile

```

- (a) Jika  $T(n)$  dihitung dari operasi penjumlahan pada pernyataan  $sum ← sum + a[i]$ , tentukan  $T(n)$ .
- (b) Nyatakan  $T(n)$  dalam notasi  $O$ -besar,  $\Omega$ -besar, dan  $\Theta$ -besar.
4. Sebuah kelas terdiri dari 7 orang laki-laki dan 5 orang perempuan. Berapa banyak cara memilih panitia beranggotakan 5 orang dari kelas tersebut jika panitiannya paling sedikit satu laki-laki dan paling sedikit satu perempuan.

**Soal dari Harlili**

5. Berapa banyak susunan string kata yang terdiri dari huruf A, E, I, O, U, X, X, X, X, X, X, X ( 5 vokal, 8 konsonan huruf mati X) jika tidak ada 2 vokal berdekatan? Jawaban minimal harus ada bentuk mencari solusi persamaan bilangan bulatnya apa.
6. Ada 6 jenis zat kimia yang perlu disimpan di dalam gudang. Beberapa pasang dari zat itu tidak dapat disimpan di dalam ruangan yang sama, karena campuran gasnya bersifat eksplosif (mudah meledak). Untuk zat yang semacam itu perlu dibangun ruang-ruang terpisah yang dilengkapi ventilasi dan penyedot udara keluar yang berlainan. Jika lebih banyak ruang yang dibutuhkan, berarti lebih banyak ongkos yang harus dikeluarkan. Karena itu perlu diketahui berapa banyak minimum ruangan yang diperlukan untuk dapat menyimpan semua zat kimia dengan aman. Berikut ini adalah daftar pasangan zat kimia yang tidak dapat disimpan di dalam ruangan yang sama:

| Zat kimia | Tidak dapat disimpan bersama zat kimia |
|-----------|----------------------------------------|
| A         | B, D                                   |
| B         | A, D, E, F, G                          |
| C         | E, G                                   |
| D         | A, F, B                                |
| E         | B, C, G                                |
| F         | B, D                                   |
| G         | C, E, B                                |

Gambarkan graf yang menyatakan persoalan (jelaskan arti simpul dan sisi yang menghubungkan dua buah simpul). Pikirkan termasuk jenis manakah persoalan ini). Kemudian tentukan jumlah minimum ruangan yang dibutuhkan untuk menyimpan semua zat kimia di atas.

7. Diberikan masukan berupa rangkaian karakter dengan urutan sebagai berikut:  
B, T, P, F, H, K, M, S, A, U, N, I, D, O, W, C
- Gambarkan pohon pencarian biner yang terbentuk.
  - Berapa perbandingan yang diperlukan untuk mencari karakter A.
8. Diberikan rangkaian string : M A A A A A A T H D D D E E C R E E T E (ada 21 karakter).
- Tentukan kode Huffman untuk enkoding setiap simbol dari rangkaian string diatas.
  - Gambarkan pula pohon Huffman yang terbentuk. Berapa rata-rata banyaknya bit yang digunakan untuk enkoding satu karakter ini?

**Soal bersama**

9. Apa prediksi nilai anda untuk kuliah ini? (A/AB/B/BC/C/D/E) (Nilai = 2)

## UAS IF2120 Matematika Diskrit (3 SKS)

UAS SEMESTER I – 2017/2018

Dosen: Rinaldi Munir, Harlili

Senin, 4 Desember 2016

Waktu: 110 menit

**Setiap soal nilainya = 11, kecuali nomor 1 nilainya = 2. Total nilai = 101****Soal dari Rinaldi Munir**

1. Apa prediksi nilai anda untuk kuliah ini? (A/AB/B/BC/C/D/E)
2. (a) Hitunglah  $51^{-1} \pmod{1008}$   
 (b) Gunakan hasil jawaban a di atas untuk menemukan semua solusi bilangan bulat  $x$  yang memenuhi kongruensi  $51x \equiv 177 \pmod{1008}$
3. Sebuah buku memiliki kode ISBN **0-1p026-690-q**. Tentukan nilai  $(p+q) \pmod{3}$  dari nomor ISBN tersebut jika diketahui  $6p \equiv 3 \pmod{7}$
4. Perhatikan algoritma *Bubble Sort* di bawah ini.

```

procedure BubbleSort (input/output s : TabelInt, input n : integer)
{ Mengurutkan tabel s[1..N] sehingga terurut menaik dengan metode }

Deklarasi
 i : integer { pencacah untuk jumlah langkah }
 k : integer { pencacah, untuk pengapungan pada setiap langkah }
 temp : integer { peubah bantu untuk pertukaran }

Algoritma:
 for i 0..n-1 downto 1 do
 for k 0..1 to i do
 if s[k+1] < s[k] then
 swap(s[k], s[k+1]) {pertukarkan s[k] dengan s[k+1]}
 endif
 endfor
 endfor

```

- (a) Berapa kali jumlah operasi perbandingan elemen-elemen larik (*array*) dilakukan pada kasus terbaik dan kasus terburuk?
- (b) Berapa kali operasi pertukaran elemen-elemen larik dilakukan pada kasus terbaik dan terburuk?
- (c) Bandingkan hasil-hasil a dan b dengan algoritma *selection sort*, mana yang lebih baik?

**Soal dari Harlili**

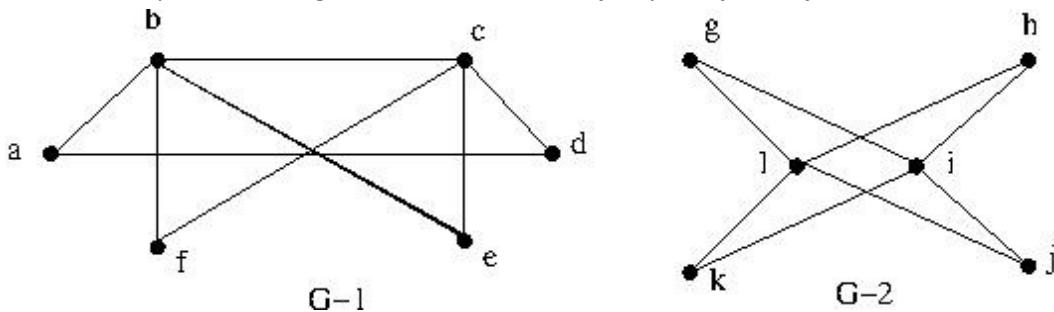
5. Suatu string biner disusun oleh 10 bit. Berapa banyak jumlah string biner yang memuat paling sedikit 3 simbol 1 dan paling sedikit 3 simbol 0?

6. Prodi Teknik Informatika STEI, disingkat TIFSTEI. Berapa banyak permutasi TFSTEI yang memuat TIF dan ST?
7. Berapa banyak solusi bilangan bulat nonnegative dari persamaan dibawah jika  $0 \leq x_1 \leq 3$ ,  $1 \leq x_2 \leq 4$ , dan  $x_3 \geq 15$ ?

$$x_1 + x_2 + x_3 + x_4 + x_5 = 21$$

**Soal dari Judhi Santoso**

8. (a). Periksa apakah graf G-1 dan G-2 bipartite, jika ya gambarkan bentuk lain yang menunjukkan bipartite.  
 (b). Selidiki apakah kedua graf tersebut isomorf, jika ya tunjukkan jika tidak sebut alasannya.



9. (a). Gambarkan pohon pencarian dari barisan bilangan berikut: 23 45 67 3 20 50 25 15 10 1.  
 (b). Diberikan notasi infix : A\*B+C-D+E\*F\*G. Gambarkan pohon ekspresi tersebut.  
 (c). Tuliskan notasi prefix dan postfix dari soal (2b).
10. Periksa apakah graf G-1 dan G-2 planar. Jawaban harus brisi langkah2nya. Jika graf tersebut planar, gambarkan graf planarnya.

## UAS IF2120 Matematika Diskrit (3 SKS)

UAS SEMESTER I – 2019/2020

Dosen: Rinaldi Munir, Harlili, Fariska Zr

Jumat, 6 Desember 2019

Waktu: 120 menit

1.

- (a) Algoritma Euclidian dapat diperluas untuk mencari PBB lebih dari dua bilangan bulat. Gunakan algoritma Euclidian untuk menghitung PBB (378, 336, 490).
- (b) Tentukan nilai x dan y bilangan bulat yang memenuhi persamaan  $336x + 490y = 14$ .
- (c) Hitung  $3^{64} \pmod{67}$  dengan menggunakan Fermat's Little Theorem

(15)

2. Diberikan potongan *pseudo-code* sebagai berikut:

```

X 0
for i 1 to n do
 for j 1 to i do
 for k 1 to j do
 x = x + A[j] * A[j] * A[k]
 end
 end
end

```

- (a) Hitunglah  $T(n) = \text{total jumlah operasi perkalian dan operasi penjumlahan di dalam algoritma tersebut}$
- (b) Tentukan notasi kompleksitas asimptotiknya dalam notasi  $O$ ,  $\Omega$ , dan  $\Theta$

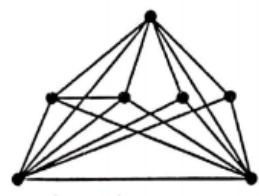
3. Diketahui fungsi kompleksitas beberapa algoritma :

- (i)  $f(n) = 7^n + n^2$
- (ii)  $g(n) = 2n^2 + n \log n$
- (iii)  $h(n) = n + 5$

Hitunglah notasi  $O$  (big-oh) dari hasil operasi berikut : (10)

- a.  $f(n) + g(n) + h(n)$
- b.  $f(n) \times g(n) \times h(n)$

4. Periksa graf G1 berikut dan berikan penjelasan/bukti untuk jawaban anda tersebut.



ned with G1

(10)

- Apakah graf G1 termasuk Graf Hamilton, Graf Euler, Graf Semi-Hamilton, dan/atau Graf Semi-Euler?
  - Apakah graf G1 termasuk graf planar?
5. Sebuah mesin multifungsi dapat digunakan untuk menjalankan 5 jenis pekerjaan berbeda (pekerjaan A, B, C, D, E). Ada biaya untuk setiap pergantian dari satu pekerjaan ke pekerjaan lain (*setup cost*). Parameternya adalah:

$$C_{ij} = \text{biaya pergantian saat pekerjaan } j \text{ dijalankan segera setelah pekerjaan } i$$

Ketentuan untuk runtunan pekerjaan dimulai dari pekerjaan I adalah:

- Awal runtunan penggeraan, biaya awal adalah  $C_{0i}$ .
- Akhir runtunan, mesin harus di-reset ke kondisi awal, memerlukan biaya akhir  $C_{i0}$ .

Untuk setiap runtunan, 1 jenis pekerjaan hanya dilakukan 1 kali saja.

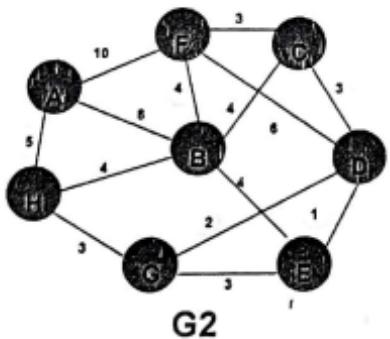
Biaya pergantian dapat dilihat pada tabel berikut.

(15)

|   | A        | B        | C        | D        | E        |
|---|----------|----------|----------|----------|----------|
| A | $\infty$ | 8        | 8        | 6        | 7        |
| B | 8        | $\infty$ | 12       | 10       | 11       |
| C | 8        | 12       | $\infty$ | 5        | 9        |
| D | 6        | 10       | 5        | $\infty$ | 15       |
| E | 7        | 11       | 9        | 15       | $\infty$ |

Biaya awal untuk setiap pekerjaan sama dengan biaya akhir, yaitu 20. Tentukan runtunan pekerjaan agar semua pekerjaan dapat dijalankan, dengan biaya total seminimal mungkin.

6. Tentukan pohon merentang minimum dari graf di G2 bawah ini. Cantumkan nama algoritma yang Anda gunakan dan tahapan pembentukan pohon berdasarkan algoritma tersebut.



(10)

1. Suatu bank nasional membuat nomor rekening terdiri dari 10 angka. Berapa banyak nomor rekening yang diperoleh jika 3 angkat pertama menyatakan kode identitas bank, angka ke 4 tidak boleh angka nol, angka ke 4 sd ke 10 harus berbeda. **(10)**
2. Berapa banyak bilangan integer positif lebih kecil dari 1.000.000 yang mempunyai satu digit = angka 9 dan mempunyai angka-angka = 19. **(10)**
3. Gunakan kode Huffman untuk *encode* simbol di dalam pesan IF\_RUNNER. Berapa rata-rata banyak bit yang digunakan *encode* satu karakter? **(10)**

Apa nilai perkiraan anda untuk mata kuliah ini? (A/AB/B/BC/C/D/E) **(2)**

## UAS Matdis 2020/2021

### 11 Desember 2020

#### TEORI BILANGAN

1. Diberikan persamaan  $621m + 483n = k$ , yang dalam hal ini  $k$  adalah PBB dari 621 dan 483,  $m$  dan  $n$  adalah bilangan bulat. Maka dapat disimpulkan

*Mark only one oval.*

- (A)  $k = 61$
- (B)  $m = -3$
- (C)  $n = 5$
- (D) B dan C benar
- (E) A dan B benar
- (F) A, B, dan C benar
- (G) semua jawaban di atas salah

2. Sebuah bilangan bulat bersisa 1 jika dibagi 5, bersisa 2 jika dibagi 6, dan bersisa 3 jika dibagi 7. Dengan menggunakan Chinese Remainder Theorem, maka

*Mark only one oval.*

- (A) Solusinya di dalam modulus 18
- (B) Bilangan bulat positif terkecil yang memenuhi adalah 206
- (C) Bilangan bulat lain yang memenuhi adalah 416
- (D) Bilangan bulat negatif terbesar yang memenuhi adalah -4
- (E) Semua jawaban benar
- (F) Hanya B dan C benar
- (G) Hanya jawaban B, C, dan D yang benar
- (H) Semua jawaban salah

3. Sebuah kekongruenan berbentuk seperti pada gambar. Misalkan  $m = 7$  dan  $n = 26$ ,  $b = 10$ , maka

$$P \equiv m^{-1} (C - b) \pmod{n}$$

*Mark only one oval.*

- Nilai  $m^{-1}$  adalah 15
- Untuk  $C = 2$ , maka  $P = 10$
- Untuk  $C = 14$ , maka  $P = 8$
- Untuk  $C = 4$ , maka  $P = 14$
- Semua jawaban di atas benar
- Semua jawaban di atas tidak ada yang benar

4. Bilangan Acak yang mungkin dibangkitkan oleh  $x_n = (3x_{n-1} + 1) \pmod{7}$  dengan umpan  $x_0 = 7$  adalah sebagai berikut:

Choose Benar/Salah for each row

| Pernyataan | Jawaban     |
|------------|-------------|
| a. 2       | Benar/Salah |
| b. 3       | Benar/Salah |
| c. 4       | Benar/Salah |
| d. 5       | Benar/Salah |

5. Terdapat sekumpulan data dengan urutan penyimpanan kunci sebagai berikut: 714, 631, 26, 373, 775, 906, 509. Jika ditempatkan dalam memori dengan indeks memori yang tersedia adalah indeks 0 hingga indeks 16, maka tentukan pernyataan berikut salah atau benar.

Choose Benar/Salah for each row

| Pernyataan                                                             | Jawaban     |
|------------------------------------------------------------------------|-------------|
| a. Jika ingin dicari data 509, maka lokasi ditemukannya pada indeks 16 | Benar/Salah |
| b. Lokasi penyimpanan data dengan kunci 906 adalah pada indeks 10      | Benar/Salah |
| c. Data dengan kunci 26 tidak dapat ditemukan pada indeks 10           | Benar/Salah |
| d. Tidak ada data yang disimpan pada indeks 13                         | Benar/Salah |

## KOMBINATORIAL

6. Suatu Perjalanan dari kota A ke B ada 4 jalan pilihan dan dari kota B ke C ada 3 jalan pilihan. Supir suatu travel melakukan perjalanan dari kota A ke C melalui kota B pulang pergi. Berapa banyak jalan pilihan bagi supir jika supir itu tidak ingin menggunakan jalan yang sama lebih dari satu kali?

Mark only one oval.

- 144 jalan
- 72 jalan
- 36 jalan
- 81 jalan
- 16 jalan
- Semua jawaban di atas salah

7. Pada suatu acara wisuda ada sesi foto bersama Rektor dan Dekan oleh fotografer. Suatu keluarga wisudawan A terdiri dari 4 orang. Berapa banyak cara bagi fotografer mengatur susunan keluarga A foto bersama Rektor dan Dekan dalam satu baris jika Rektor dan Dekan harus berdampingan?

Mark only one oval.

- 240 cara
- 120 cara
- 720 cara
- 360 cara
- 600 cara
- Semua jawaban di atas salah

8. Berapa paling sedikit jumlah mahasiswa yang harus ikut UAS pada ujian matematika diskrit sehingga terdapat 10 mahasiswa yang mendapat nilai yang sama, jika index mahasiswa ada 7 yaitu A, AB, B, BC, C, D, dan E?

Mark only one oval.

- 70 mahasiswa
  - 63 mahasiswa
  - 64 mahasiswa
  - 71 mahasiswa
  - 69 mahasiswa
  - Semua jawaban di atas salah
9. Berapa banyak solusi bilangan bulat dari  $x_1 + x_2 + x_3 + x_4 + x_5 = 21$  jika  $1 \leq x_2 \leq 4$ ,  $x_3 \geq 15$  ?

*Mark only one oval.*

- 126 solusi
- 106 solusi
- 210 solusi
- 141 solusi
- 111 solusi
- Semua jawaban di atas salah

10. Tentukan koefisien dari suku yang memuat  $x^{11}$  dan  $y^4$  dari ekspansi  $(2x^2 - 3xy^2 + z^2)^6$

*Mark only one oval.*

- 4820
- 3600
- 2160
- 4320
- 1440
- Semua jawaban di atas salah

**Soal ini dibatalkan karena ada typo soal, seharusnya  $2x^3$  tetapi tertulis  $2x^2$**

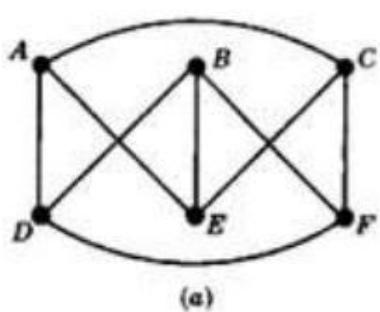
## **GRAF**

11. Derajat setiap simpul pada sebuah graf sederhana adalah 5. Berapakah banyaknya simpul yang mungkin?

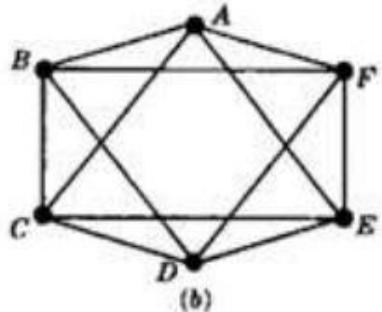
*Mark only one oval.*

- 1, 3, 5, 7
- 2, 4, 5, 8
- 2, 3, 4, 5
- 15, 16, 17
- 10, 12, 14
- Tidak ada jawaban yang benar

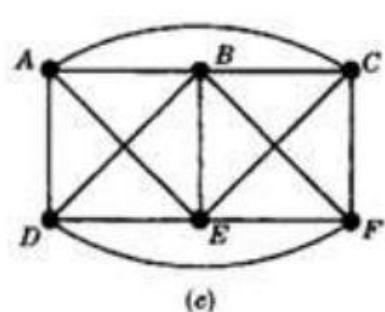
12. Pernyataan yang benar terkait keplanaran graf-graf berikut adalah:



(a)



(b)



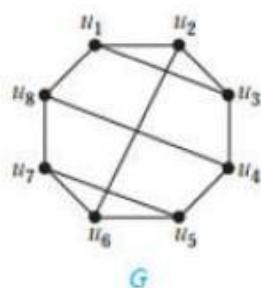
(c)

Mark only one oval.

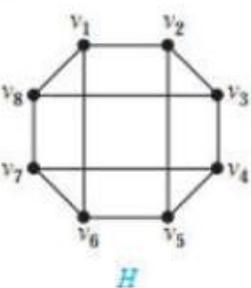
- Ketiga graf tersebut adalah graf planar.
- Hanya graf (b) saja yang planar.
- Graf (a) tidak planar, karena tidak memenuhi ketidaksamaan Euler kedua.
- Graf (b) tidak planar, karena mengandung upagraf yang isomorfik dengan graf Kuratowski kedua
- Graf (c) tidak planar, karena mengandung upagraf yang isomorfik dengan graf Kuratowski kedua
- Tidak ada jawaban yang benar

13. Pernyataan yang benar terkait kedua pasangan graf berikut adalah:

A

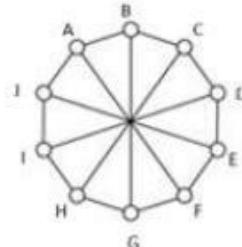
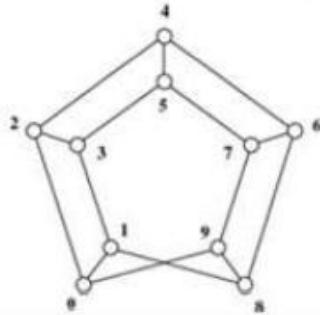


G



H

B



Mark only one oval.

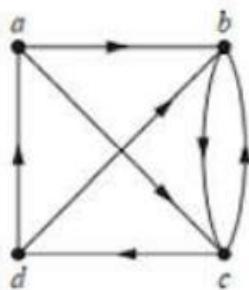
- Kedua pasangan graf tersebut bersifat isomorfik
- Kedua pasangan graf tersebut non-isomorfik
- Pasangan B saja yang isomorfik
- Pasangan A saja yang isomorfik
- Tidak ada jawaban yang benar

14. Perhatikan graf-graf dan pernyataan berikut!

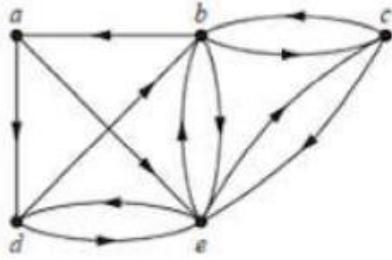
- (1) Ketiga graf tersebut merupakan graf Euler
- (2) Graf A merupakan graf Euler sekaligus juga graf Hamilton
- (3) Graf A dan C memiliki lintasan Euler
- (4) Banyaknya sirkuit Hamilton pada graf A adalah 12 buah
- (5) Graf B merupakan graf Hamilton sekaligus juga graf semi-Euler.

Pernyataan yang benar terkait graf tsb adalah:

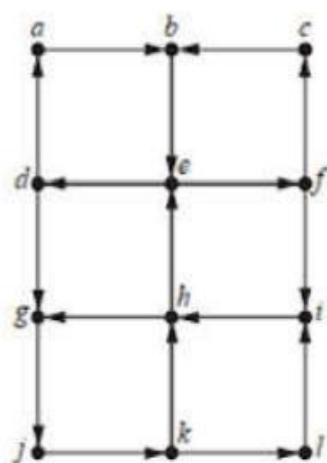
G1



G2



G3



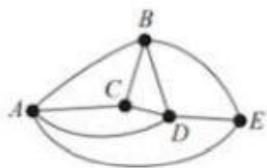
Mark only one oval

- 1, 2, 3, 4, 5
- 1, 2, 3
- 2, 3, 4
- 3, 4, 5
- 1, 3, 5
- 1, 2, 4
- 2
- 4
- Tidak ada jawaban yang benar

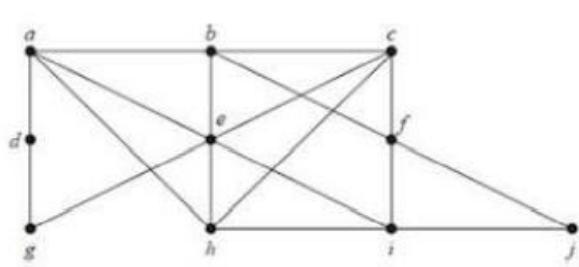
Ada kesalahan pencantuman gambar graf,maka soal tsb dianulir semua mhs dibetulkan semua (mendapat nilai yang sama)

15. Pernyataan yang benar adalah:

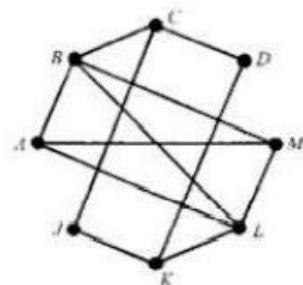
A



B



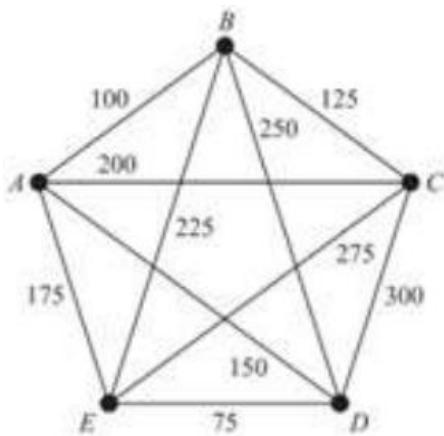
C



Mark only one oval.

- Ketiga graf tersebut memiliki bilangan kromatik yang sama
- Graf C memiliki bilangan kromatik terbesar
- Tidak ada yang memiliki bilangan kromatik lebih dari 3
- Graf B memiliki bilangan kromatik terkecil
- Ada sebuah graf yang lebih besar bilangan kromatiknya dari kedua graf yang lain
- Tidak ada jawaban yang benar

16. Graf berikut adalah peta jarak dari 5 buah kota. Tur terpendek yang harus dilalui oleh seorang pedagang bila pedagang itu berangkat dari salah satu kota asal dan menyinggahi setiap kota asal dan menyinggahi setiap kota tepat satu kali dan kembali lagi ke kota asal keberangkatan adalah



*Mark only one oval.*

- A. 625
- B. 725
- C. 775
- D. 1100
- E. Tidak ada jawaban yang benar

## POHON

17. Berikut ini yang bukan contoh formula dengan notasi postfix untuk symbol (x, y, z) dan operator  $\{+, \times, {}^\circ\}$  adalah

- (1)  $xy+zx{}^\circ+x{}^\circ$
- (2)  $xyz++yx++$
- (3)  $xyxy{}^\circ{}^\circ xy{}^\circ{}^\circ z{}^\circ+$
- (4)  $\times{}^\circ xz \times \times xy$
- (5)  $xz \times$
- (6)  $zz{}^\circ$
- (7)  $yyyy{}^\circ{}^\circ{}^\circ$
- (8)  $zx+yz+{}^\circ$

*Mark only one oval.*

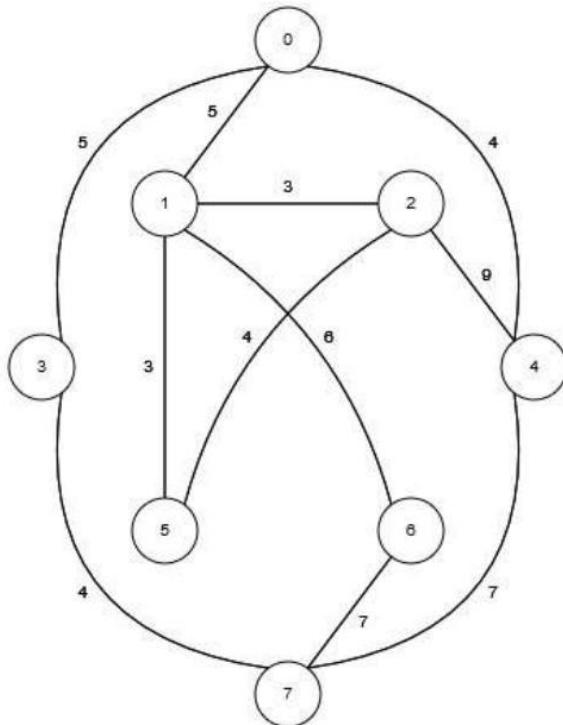
- 1, 2, 3
- 4, 5, 6
- 6
- 5, 7, 8
- 3, 4, 5
- 4, 5
- 5
- 5, 6
- Tidak ada jawaban yang benar

18. Berikut ini yang tidak mungkin menjadi bagian dari kode Huffman untuk string “LA LA LAKOBUN” (karakter spasi termasuk) adalah:

*Mark only one oval.*

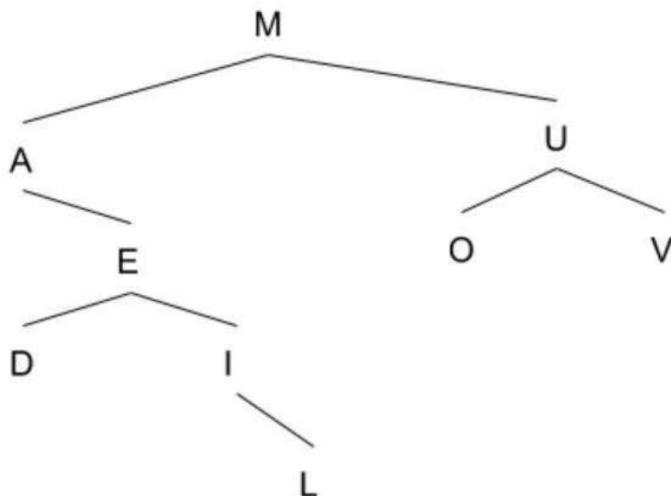
- A: 01
- B: 101
- C: 100
- D: 00
- E: 1100
- F: 101
- G: 1111
- H: Tidak ada yang benar

19. Diberikan sebuah graf sebagai berikut yang menyatakan rencana pembangunan jaringan jalan raya di sebuah pulau. Simpul menyatakan kota, sisi menyatakan ruas jalan. Bobot pada setiap sisi menyatakan biaya pembangunan ruas jalan yang menghubungkan dua kota. Karena anggaran yang terbatas, maka tidak semua ruas jalan tersebut dibangun. Jika diinginkan antara setiap pasang kota sembarang di pulau dapat dihubungkan dengan satu lintasan tunggal dan kita ingin total biaya pembangunan jalan semurah mungkin, maka yang dapat disimpulkan dari rencana pembangunan jalan tersebut adalah:



Mark only one oval.

- (A) Cukup dibangun 7 ruas jalan saja
  - (B) Biaya minimum pembangunan jalan adalah 27
  - (C) Cukup dibangun 6 ruas jalan saja
  - (D) Biaya minimum pembangunan jalan adalah 30
  - (E) Jawaban C dan D benar
  - (F) Jawaban A dan D benar
  - (G) Jawaban A dan B benar
  - (H) Jawaban B dan C benar
  - I Tidak ada jawaban yang benar
20. Jika pohon pencarian biner yang dihasilkan adalah sebagai berikut, tentukan kemungkinan urutan pencarian yang mungkin dilakukan.



Choose Benar/Salah for each row

| Pernyataan                   | Jawaban     |
|------------------------------|-------------|
| a. M, U, A, E, I, D, V, L, O | Benar/Salah |
| b. M, U, A, V, D, O, I, E, L | Benar/Salah |
| c. M, A, U, O, D, E, L, I, V | Benar/Salah |
| d. M, A, U, V, E, I, O, D, L | Benar/Salah |

21. Sebuah surat berantai dimulai ketika seseorang mengirim sebuah surat kepada 3 (tiga) orang lainnya. Tiap orang yang menerima surat mengirimkan surat tersebut kepada 3 (tiga) orang lain yang belum pernah menerima surat tersebut (catatan: surat yang dikirimkan pasti sampai/diterima di tempat tujuan). Misalkan ada 1.000 orang yang mengirimkan surat tersebut sebelum rantai berakhir; tentukan apakah pernyataan berikut salah atau benar.

Choose Benar/Salah for each row

| Pernyataan                                                                                                                                               | Jawaban     |
|----------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| a. Persoalan tersebut dapat dimodelkan dengan pohon biner                                                                                                | Benar/Salah |
| b. Saat 1.000 orang sudah mengirimkan surat berantai, maka terdapat 3.000 orang yang sudah menerima surat berantai tersebut.                             | Benar/Salah |
| c. Saat 1.000 orang sudah mengirimkan surat berantai, maka terdapat 2.000 orang yang sudah menerima surat dan belum mengirimkan surat berantai tersebut. | Benar/Salah |
| d. Jumlah orang yang sudah mengirimkan surat dapat dimodelkan dengan Deret Geometri.                                                                     | Benar/Salah |

## KOMPLEKSITAS ALGORITMA

22. Diberikan potongan algoritma dalam notasi pseudo-code sebagai berikut. Asumsikan n adalah perpangkatan dari 2. Maka, dapat disimpulkan:

**int**  $k, t, n$ **int**  $a[n+1]$ **k := 1****t := 0****while**  $k \leq n$  **do****t :=  $t + a[k]$** **k :=  $2k$** **end |while***Mark only one oval.*

- (A) Algoritma tersebut membutuhkan waktu dalam  $O(\log n)$
- (B) Banyaknya operasi penjumlahan ( $t + a[k]$ ) adalah  $T(n) = \log n$
- (C) Kompleksitas asimptotik algoritma tersebut adalah  $O(2^n)$
- (D) Banyaknya operasi penjumlahan ( $t + a[k]$ ) adalah  $T(n) = 2^n + 1$
- (E) A dan B benar
- (F) C dan D benar
- (G) Semua jawaban salah

23. Dari notasi asimptotik untuk setiap  $T(n)$  berikut, manakah yang SALAH?

- a.  $n \log n + n/2 = O(n)$
- b.  $\sqrt{n} + \log n = \Omega(1)$
- c.  $10\sqrt{n} + \log n = O(n)$
- d.  $\sqrt{n} + \log n = \Omega(\log n)$

*Mark only one oval.*

- a saja
- c dan d
- a saja
- a dan b
- semua salah
- a, b, c, dan d benar, tidak ada yang salah

24. Tinjau potongan algoritma berikut. Kompleksitas algoritma dihitung dari banyaknya operasi penjumlahan dan perkalian di dalam  $x := (a[k] + b[j]) * 2$  dilakukan. Maka dapat disimpulkan bahwa:**for**  $i := 1$  **to**  $n$ **for**  $j := 1$  **to**  $i+1$  $x := (a[k] + b[j]) * 2$ **end for****end for**

Mark only one oval.

- (A) Jumlah operasi perkalian adalah  $T(n) = n^2 + 3n$
- (B) Jumlah operasi penjumlahan adalah  $T(n) = n^2 + 3n$
- (C)  $T(n) = O(n^2)$  karena  $n^2 + 3n \leq 4n^2$
- (D)  $T(n) = \Omega(n^2)$  karena  $n^2 + 3n \geq n^2$
- (E) Jawaban A, B, C, dan D benar
- (F) Jawaban C dan D benar
- (G) Semua jawaban salah

25. Dari pernyataan terkait kompleksitas waktu berikut ini, tentukan pernyataan tersebut benar atau salah.

- a.  $T(n) = 4(n^{2^n}) - 1 + 3(n^{n^2}) - 3 \log n$ ;  $T(n) = O(n^{2^n})$  dengan  $C = 5$  dan  $n_0 = 1$ .
- b.  $T(n) = (n^2 + 1) \log(n^2 + 1) + n!$ ;  $T(n) = O(n!)$  dengan  $C = 2$  dan  $n_0 = 1$ .
- c.  $T(n) = (3n + n^2 \log n)(4 \log n - 2)$ ;  $T(n) = O(n^2(\log n)^2)$  dengan  $C = 7$  dan  $n_0 = 1$ .
- d.  $T(n) = (1+3n-2n^2)(5-7n)$ ;  $T(n) = O(n^2)$  dengan  $C = 15$  dengan  $n_0 = 1$ .

Choose Benar/Salah for each row

| Pernyataan     | Jawaban     |
|----------------|-------------|
| Pernyataan (a) | Benar/Salah |
| Pernyataan (b) | Benar/Salah |
| Pernyataan (c) | Benar/Salah |
| Pernyataan (d) | Benar/Salah |

**UAS MATDIS 2021/2022****TEORI BILANGAN**

1. Pembagi Bersama Terbesar disingkat PBB. PBB (a,b) dapat dinyatakan sebagai kombinasi lanjar dari a dan b yaitu PBB (a,b) = ma+nb. Nyatakan PBB (36,48) sebagai kombinasi lanjar dari 36 dan 48.
- a.  $24 = (-2) 36 + 48$   
 b.  $12 = (-3) 36 + (2) 48$   
 c.  $12 = (-1) 36 + (1) 48$   
 d.  $24 = (3) 36 + (-2) 48$   
 e. Jawaban tidak ada.
2. Berapakah  $13/8 \pmod{11}$ ?
- 3  
 4  
 5  
 6  
 7  
 Tidak dapat dihitung  
 Semua jawaban di atas tidak ada yang benar
3. Sebuah buku memiliki kode ISBN sebagai berikut 0-30X5-4561-Y dan memenuhi  $3X \pmod{11} = 1$ , serta Y adalah karakter uji. Maka dapat disimpulkan bahwa:
- X + Y = 12  
 X \* Y = 28  
 Y - X = 5  
 Y mod X = 2  
 2X + Y = 15  
 Semua jawaban di atas tidak ada yang benar
4. Bilangan acak yang dibangkitkan oleh  $X_n = (3 X_{n-1} + 2) \pmod{13}$  dengan umpan (seed)  $X_0 = 1$  adalah bilangan-bilangan berikut.
- a. 1, 2, 3  
 b. 1, 4, 5  
 c. 2, 5, 6  
 d. 2, 4, 6  
 e. Jawaban tidak ada.
5. Tentukan nilai-nilai x yang merupakan balikan modulo dari 12 dalam modulo 23:  $x = 12^{-1} \pmod{23}$ .

|         | Benar                            | Salah                            |
|---------|----------------------------------|----------------------------------|
| a. 25   | <input checked="" type="radio"/> | <input type="radio"/>            |
| b. - 10 | <input type="radio"/>            | <input checked="" type="radio"/> |
| c. - 21 | <input checked="" type="radio"/> | <input type="radio"/>            |
| d. 14   | <input type="radio"/>            | <input checked="" type="radio"/> |

**KOMBINATORIAL**

6. Huruf abjad di dalam Bahasa Indonesia terdiri dari 26 huruf, terdiri dari 5 huruf vokal dan 21 huruf konsonan. Tentukan jumlah kata yang panjangnya 5 huruf yang memuat 3 huruf konsonan berbeda dan 2 huruf vokal berbeda, namun kata-katanya harus memuat huruf B.
- C(21, 3) x C(5, 2)  
 C(21, 3) x C(5, 2) x 5!  
 C(20, 2) x C(5, 2)  
 C(20, 2) x C(5, 2) x 5!  
 C(20, 2) x C(5, 2) x 5  
 Semua jawaban di atas tidak ada yang benar
7. Berapa banyak string biner dengan panjang 10 yang memuat paling sedikit 3 simbol 1 dan paling sedikit 3 simbol 0.
- a. 912  
 b. 848  
 c. 683  
 d. 1024  
 e. Jawaban tidak ada.
8. Berapa banyak string bilangan-bilangan (0, 1, 2) dengan panjang 10 yang memuat 2 simbol 0, 3 simbol 1, 5 simbol 2.
- a. 3600  
 b. 2860  
 c. 2520  
 d. 6300  
 e. Jawaban tidak ada.
9. Berapa banyak solusi bilangan bulat nonnegative dari persamaan berikut jika  $0 \leq x_i \leq 10$  dimana  $i=1, 2, 3, 4, \text{ dan } 5$ ?  
 Persamaan:  $x_1 + x_2 + x_3 + x_4 + x_5 = 21$
- a. 12.650  
 b. 11.649  
 c. 7.645  
 d. 10.240  
 e. Jawaban tidak ada.
10. Berapakah koefisien  $x^9$  di dalam  $(2 - x)^{19}$ ?

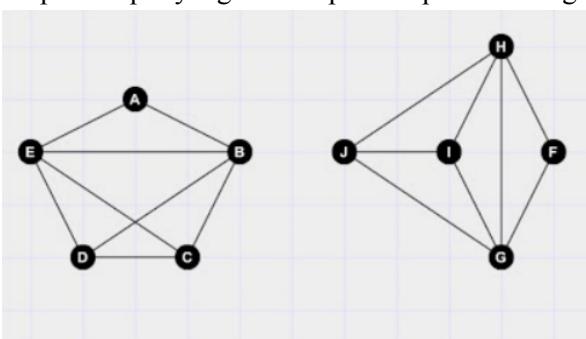
- $2^{10} \times C(19, 9)$
- $2^{10} \times C(19, 9)$
- $-2^9 \times C(19, 9)$
- $2^9 \times C(19, 9)$
- $(-2)^9 \times C(19, 9)$
- Semua jawaban di atas tidak ada yang benar

11. Berapa paling sedikit jumlah mahasiswa ikut UAS kelas Matematika Diskrit sehingga terdapat 6 mahasiswa mendapat nilai index yang sama jika terdapat 5 nilai index yaitu A, B, C, D, dan E

- a. 25
- b. 26
- c. 24
- d. 30
- e. Jawaban tidak ada.

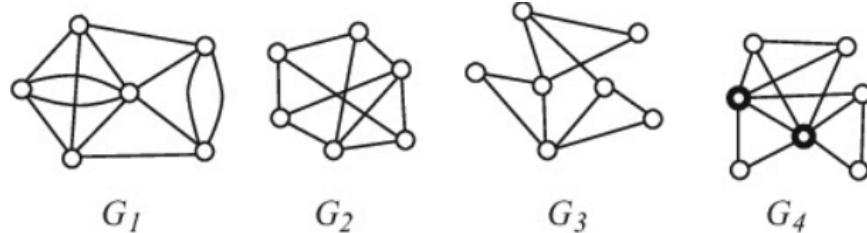
### GRAF

12. Diberikan dua buah graf seperti gambar berikut. Apakah kedua graf di bawah ini isomorfik! Jika ya, tuliskan simpul-simpul yang berkoreponden pada kedua graf tersebut.



- Tidak isomorfik
- Isomorfik; koresponden: A-F, B-I, C-J, D-H, E-G
- Isomorfik; koresponden: A-F, B-H, C-I, D-J, E-G
- Isomorfik; koresponden: A-F, B-J, C-G, D-I, E-H
- Isomorfik; koresponden: A-F, B-G, C-H, D-I, E-J
- Tidak ada jawaban yang benar

13. Perhatikan empat buah graf pada gambar di bawah ini. Graf mana yang setiap sisinya dapat dilalui tepat satu kali?



Tandai satu oval saja.

- G1
- G2
- G3
- G1 dan G3
- G1, G2, dan G3
- G1 dan G4
- G3 dan G4
- Semua jawaban di atas tidak ada yang benar

14. Diketahui suatu graf tak berarah dinyatakan dalam matriks ketetanggaan (adjacent) dibawah ini . Pernyataan yang benar sebagai berikut.

$$\begin{bmatrix} 1 & 2 & 0 & 1 \\ 2 & 0 & 3 & 0 \\ 0 & 3 & 1 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

Tandai satu oval saja.

- a. Graf mempunyai 4 simpul, 9 sisi, terdapat 2 sisi gelang.
- b. Graf mempunyai 4 simpul, 9 sisi ,semuanya sisi ganda.
- c. Graf mempunyai 4 simpul, 9 sisi, terdapat simpul terpencil.
- d. Graf mempunyai 4 simpul, 9 sisi, terdapat simpul berderajat satu.
- e. Jawaban tidak ada.

15. Terdapat sebuah persoalan teka-teki aritmetika sebagai berikut. Setiap huruf harus diganti dengan sebuah angka. Huruf yang sama harus merepresentasikan angka yang sama, huruf yang berbeda harus merepresentasikan angka yang berbeda, sedemikian sehingga rangkaian angka tersebut merepresentasikan penjumlahan yang benar. Huruf awal setiap kata tidak boleh merepresentasikan angka nol. Persoalan ini akan diselesaikan dengan pendekatan pewarnaan graf

$$\begin{array}{r}
 \text{S E N D} \\
 \text{M O R E} \\
 \hline
 \text{M O N E Y}
 \end{array} +$$

| Benar | Salah |
|-------|-------|
|-------|-------|

- a. Simpul merepresentasikan angka.
- |                       |                                  |
|-----------------------|----------------------------------|
| <input type="radio"/> | <input checked="" type="radio"/> |
|-----------------------|----------------------------------|

- b. Sisi graf berupa sisi berarah.
- |                       |                                  |
|-----------------------|----------------------------------|
| <input type="radio"/> | <input checked="" type="radio"/> |
|-----------------------|----------------------------------|

- c. Dua buah simpul yang dihubungkan dengan sebuah sisi menyatakan bahwa kedua simpul tersebut harus diberi nilai yang berbeda.
- |                                  |                       |
|----------------------------------|-----------------------|
| <input checked="" type="radio"/> | <input type="radio"/> |
|----------------------------------|-----------------------|

- d. Terdapat simpul anting-anting pada graf yang terbentuk untuk mepresentasikan persoalan tersebut.
- |                       |                                  |
|-----------------------|----------------------------------|
| <input type="radio"/> | <input checked="" type="radio"/> |
|-----------------------|----------------------------------|

16. Jumlah simpul yang mungkin, jika suatu graf sederhana memiliki 20 buah sisi dan tiap simpul memiliki derajat yang sama

| Benar | Salah |
|-------|-------|
|-------|-------|

- |       |                                  |                                  |
|-------|----------------------------------|----------------------------------|
| a. 5  | <input type="radio"/>            | <input checked="" type="radio"/> |
| b. 8  | <input checked="" type="radio"/> | <input type="radio"/>            |
| c. 20 | <input checked="" type="radio"/> | <input type="radio"/>            |
| d. 40 | <input checked="" type="radio"/> | <input type="radio"/>            |

## POHON

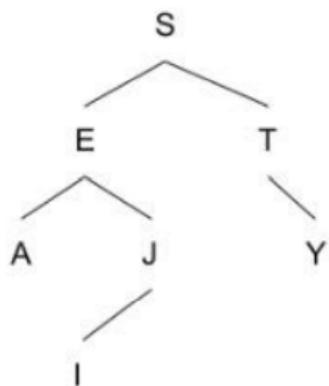
17. Diberikan frekuensi setiap simbol dari suatu string dibawah ini. String dikompresi menggunakan kode Huffman. Hasil Kode Huffman untuk enkoding sebagai berikut. A: 0,20; B: 0,10; C: 0,15; D: 0,25; E: 0,30.

- a. Simbol A mempunyai banyak bit paling minimal.
- b. Simbol D dan E mempunyai banyak bit = 2 bit.
- c. Simbol A, B, dan C mempunyai banyak bit = 2 bit.
- d. Simbol B mempunyai frekuensi paling kecil maka banyak bit encoding paling minimal.
- e. Jawaban tidak ada.

18. Bentuklah pohon ekspresi dari notasi postfix berikut: {Postfix: A B C ^ - D / E G H \* F - + \* }. Maka, dapat disimpulkan bahwa

- A) Notasi prefix: \* / - A ^ B C D + E - \* G H F
- B) Notasi postfix: A - B ^ C / D \* E + F \* G - H
- C) Notasi prefix: \* - / A ^ C B D + E - \* H F G
- D) Notasi postfix: A ^ B - C / D + E \* G - H \* F
- A dan B benar
- C dan D benar
- A dan D benar
- Tidak ada satupun jawaban di atas yang benar

19. Terdapat pohon pencarian sebagai berikut. Data masukan yang mungkin untuk pohon pencarian tersebut adalah urutan sebagai berikut



Benar      Salah

- a. S, E,  
J, A, I,         
Y, T.

- b. S, T,  
E, Y, J,         
I, A.

- c. S, T,  
E, A, I,         
Y, J.

- d. S, E,  
T, Y, A,         
J, I.

20. Terdapat persoalan missionaries and cannibals sebagai berikut. Tiga missionaries and tiga cannibals akan menyeberang sungai dan tersedia sebuah perahu di tempat missionaries and cannibals berada. Perahu hanya bisa diisi maksimal dua orang, dan harus ada minimal satu orang untuk memindahkan perahu dari satu sisi sungai ke sisi yang lain. Jika pada suatu sisi sungai jumlah cannibals lebih banyak daripada jumlah missionaries, maka missionaries akan dimakan. Tujuannya adalah semua missionaries and cannibals tiba di seberang sungai dalam keadaan semua selamat. Proses pencarian solusi dari persoalan tersebut memanfaatkan pohon pencarian.

---

Benar      Salah

---

a. Sisi

merepresentasikan kemungkinan aksi dalam hal ini perpindahan perahu dan isi perahu.

b. Simpul

merepresentasikan kondisi pada tiap sisi sungai dan posisi perahu di suatu saat.

c. Lintasan dari akar hingga ke daun adalah solusi dari persoalan, jika daun adalah goal atau tujuan yang ingin dicapai.

d. Pohon

pencarian yang terbentuk adalah pohon biner.

21. Terdapat sebuah pohon dengan ketinggian  $h$ , setiap simpul memiliki maksimum  $m$  anak, sejumlah  $d$  simpul daun, dan sejumlah  $i$  simpul dalam. Tentukan tiap pernyataan berikut ini benar atau salah jika pohon tersebut berupa pohon  $m$ -ary penuh dan  $n$  menyatakan seluruh simpul dalam pohon.

---

Benar      Salah

---

a.

Banyaknya simpul dalam adalah:  $i = n / m$ .

b.

Banyaknya simpul adalah:  $n = (m^{(h+1)} - 1) / (m - 1)$ .

- c.  
Banyaknya simpul daun adalah: d  
 $= n - i$ .



- d.  
Banyaknya simpul daun adalah: d  
 $= \lfloor (m - 1) / n \rfloor + 1$



## KOMPLEKSITAS ALGORITMA

22. Notasi Big-O untuk  $T(n) = [n \log(n) + 1]^2 + [\log(n) + 1](n^2 + 1)$  adalah

- $O(n \log n)$
- $O(n^2 \log n)$
- $O(n^2 (\log n)^2)$
- $O(n \log n)^2$
- $O(n^2 \log n^2)$
- Semua jawaban di atas tidak ada yang benar

23. Diberikan potongan program dalam Bahasa C sebagai berikut. Kompleksitas waktu algoritma,  $T(n)$ , dihitung dari jumlah operasi tambah (+) dan operasi kali (\*). Nilai  $T(n)$  dan notasi Big-O adalah sebagai berikut

```
for (i=1; i<= n; i++) {
 j = n;
 temp = 0;
 while (i <= j) {
 for (k=1; k<=n; k++) {
 temp = temp + A[i] * B[j][k];
 }
 j = j - 1;
 res[i] = temp;
 }
}
return res;
```

Tandai satu oval saja.

- $T(n) = n^2 + n = O(n^2)$
- $T(n) = 1/2(n^2 + n) = O(n^2)$
- $T(n) = n^3 + n^2 = O(n^3)$
- $T(n) = n^3 + 2n^2 = O(n^3)$
- $T(n) = 1/2 (n^3 + n^2) = O(n^3)$
- Tidak ada jawaban yang benar

24. Dari pernyataan kompleksitas algoritma berikut ini, tentukan pernyataan tersebut benar atau salah.

---

Benar      Salah

a.  $T(n) = (4$

$\log(n) - 2$

$n^2)(1-7n);$

$T(n) =$

$O(n^2)$

dengan C =

15 dan  $n \geq$ 

1.

b.  $T(n) = n$

$\log(n^4 -$

$8)) + (n^3)$

$\log n; T(n)$

$= O((n^3)$

$\log n)$

dengan C =

2 dan  $n \geq$ 

10.

c.  $T(n) = 3$

$n^{(2^n)} + 6$

$n^{(n^2)}$

$T(n) =$

$O(n^{(2^n)})$

dengan C =

8 dan  $n \geq$ 

1.

d.  $T(n) =$

$(n+1)(n+3)n$

$/ (n+2);$

$T(n) = O(n)$

dengan C =

3 dan  $n \geq$ 

1.

25. Terdapat sejumlah n koin (dengan n adalah bilangan hasil perpangkatan dua), dan satu diantaranya palsu (memiliki bobot yang lebih ringan dibandingkan koin yang lain). Timbangan digunakan untuk menemukan koin yang palsu dengan cara membagi dua sama banyak untuk kemudian ditimbang, bagian yang lebih ringan akan ditimbang lagi tiap setengah bagiannya untuk menemukan yang lebih ringan, begitu seterusnya hingga tersisa 1 koin di tiap sisi timbangan dan ditemukan koin yang palsu.

---

Benar      Salah

a. Proses ini

dapat didekati

dengan

memanfaatkan

rekursi.

b. Perubahan ukuran n pada tiap kali penimbangan hingga akhir: n,  $(n - 2)$ ,  $((n - 2) - 2)$ ,  $((((n - 2) - 2) - 2)$ , ..., 2.

c. Banyaknya penimbangan yang dilakukan adalah:  $T(n) = 1 + T(n/2)$ .

d. Dengan asumsi  $n = 2^k$ , maka  $T(n) = k + T(1)$  dan  $T(n) = O(\log n)$ .

IF2121

Logika Komputasional

## UTS LOGIKA KOMPUTASIONAL

UTS SEMESTER II – 2004/2005

IF-2251 Informatical Logics

Day&Date: Wednesday, March 24, 2004

Allocated Time: 120 minutes

### CLOSED BOOK

Notation : we use  $\sim$  for negation,  $\&$  for conjunction,  $|$  for disjunction,  $\Rightarrow$  for implication,  $\Leftarrow$  for reduction, and  $\Leftrightarrow$  for equivalence. Please use the same notation in your solution.

1. Translate the following English sentences of propositional logic. Use only the logical constants we provided for you. For instance, the sentence “It is either raining or snowing” with the given logical constants:

raining = “It is raining”

snowing = “It is snowing”

*should be answered by:*

raining  $|$  snowing

Here is the problem :

“you can go to the fair if you are not sick, but otherwise you must stay home and rest.”

f = you can go to the fair

s = you are sick

h = you stay at

home r = you rest

(SCORE : 20)

2. Given  $(p \Rightarrow q) \Rightarrow (q \Rightarrow \sim r)$ ,  $(q \Rightarrow r)$ , and  $(p \Rightarrow q)$ , give a formal proof showing  $\sim p$  (i.e. use only the Standard axiom schemata and Modus Ponens). (SCORE : 20)
3. Convert this sentence into clausal form.  $((\sim(p \& q) \mid r) \Leftrightarrow \sim t) \& (p \mid r)$  (SCORE : 20)
4. Prove by resolution (i.e. negate, put into clausal form, and prove the empty clause) that the following sentence is valid. (SCORE : 20)
5. Translate each of the following English sentences into an equivalent sentences in predicate logic. Use only the relation and object constants given in each part and you can introduce your own variables if necessary. (SCORE : 20)

Kings eat bread and honey on Sundays.

Object constants : breadAndHoney

Relational constant with arity of 1 : isSundays, king

Relational constant with arity of 3 :

eats(x,y,z) means x eats y on day z

IF2252 – Logika Informatika

UTS SEMESTER II – 2005/2006

90 menit

Kerjakan di lembar soal ini dan sebaliknya. Tidak ada tambahan kertas lain

1. Bukti dengan menggunakan kaidah inferensi bahwa “mahasiswa tidak malas belajar” dapat diturunkan dari premis berikut ini :  
Kalau mahasiswa malas belajar dan sering mbolos kuliah, mahasiswa tidak lulus ujian.  
Kalau mahasiswa tidak lulus ujian, orang tuanya akan marah.  
Mahasiswa sering mbolos kuliah tetapi orang tuanya tidak marah.
2. Dengan menggunakan resolusi proposisi, buktikan bahwa :  
 $\{(q \wedge (p \wedge s)), (\sim s \wedge \sim p), ((p \wedge s) \rightarrow t)\} \models (q \rightarrow t)$
3. Jika diberikan premis  $(p \rightarrow q) \wedge (q \rightarrow \sim r)$ ,  $(q \rightarrow r)$ , dan  $(p \rightarrow q)$ , dengan menggunakan hanya standard axiom schemata dan modus ponens saja, buktikan bahwa  $\sim p$ .
4. Jika diketahui ada
  - Suatu relasi dengan aritas 1 : orang(x) artinya x adalah orang.
  - Suatu relasi dengan aritas 2 : takut(x,y) artinya x takut pada y.
  - Suatu relasi dengan aritas 2 : ibu(x,y) artinya y adalah ibu dari x.

Tuliskan ekspresi logikanya untuk menyatakan bahwa ada orang yang tidak takut pada siapapun kecuali pada ibunya. (Catatan : anda tidak boleh membuat relasi, fungsi, dan konstanta lain kecuali yang sudah disebutkan diatas)

## UTS SEMESTER II – 2006/2007

Matakuliah : IF2252 Logika Informatika  
Kelas : K1  
Dosen : Dr. Ir. Rila Mandala, M.Eng  
Waktu : 120 menit  
Sifat Ujian : Closed-book

1. Jika diketahui predikat anak(X,Y) yang berarti bahwa X adalah anak dari Y, buatlah rule untuk predikat sepupu(X,Y) yang berarti X adalah sepupu dari Y. Catatan : Tidak boleh ada duplikasi. (NILAI : 10)

2. Tuliskan apa output program di bawah ini (NILAI :

15): DOMAINS

list = integer\*

PREDICATES

add1(list,list)

CLAUSES

add1([],[]).

add1([Head | Head2[Tail]], [Head2 | Tail1]):-

add1(Tail,Tail1).

GOAL

add1([1,2,3,4,5,6,7,8], NewList).

3. Tuliskan apa output program di bawah ini (NILAI :

15) : DOMAINS

integerlist = integer\*

PREDICATES

append(integerlist,integerlist,integerlist)

CLAUSES

append([],List,List).

append([H | L1], List2, [H | L3]):-

append(L1,List2,L3).

GOAL

append(L1, L2, [1,2,4]).

## UTS SEMESTER II - 2006/2007

Matakuliah : IF2252 Logika Informatika  
 Kelas : K1  
 Dosen : Dr. Ir. Rila Mandala, M.Eng  
 Waktu : 90 menit  
 Sifat Ujian :

## Closed-book Standar Axiom

## Schemata

$$\begin{aligned}
 \text{II: } & \varphi \Rightarrow (\psi \Rightarrow \varphi) \\
 \text{ID: } & (\varphi \Rightarrow (\psi \Rightarrow \chi)) \Rightarrow ((\varphi \Rightarrow \psi) \Rightarrow (\varphi \Rightarrow \chi)) \\
 \text{CR: } & (\neg\psi \Rightarrow \varphi) \Rightarrow ((\neg\psi \Rightarrow \neg\varphi) \Rightarrow \psi) \\
 & (\psi \Rightarrow \varphi) \Rightarrow ((\psi \Rightarrow \neg\varphi) \Rightarrow \neg\psi) \\
 \text{EQ: } & (\varphi \Leftrightarrow \psi) \Rightarrow (\varphi \Rightarrow \psi) \\
 & (\varphi \Leftrightarrow \psi) \Rightarrow (\psi \Rightarrow \varphi) \\
 & (\varphi \Rightarrow \psi) \Rightarrow ((\psi \Rightarrow \varphi) \Rightarrow (\varphi \Leftrightarrow \psi)) \\
 \text{OQ: } & \underline{(\varphi \Leftarrow \psi) \Leftrightarrow (\psi \Rightarrow \varphi)} \\
 & (\varphi \vee \psi) \Leftrightarrow (\neg\varphi \Rightarrow \psi) \\
 & (\varphi \wedge \psi) \Leftrightarrow \neg(\neg\varphi \vee \neg\psi)
 \end{aligned}$$

1. Jika diketahui premis  
 $(p \Rightarrow q) \Rightarrow (q \Rightarrow \neg r)$ ,  
 $(q \Rightarrow r)$ ,  
 $(p \Rightarrow q)$ ,  
Buktikan bahwa kesimpulannya adalah  $\neg p$  dengan menggunakan Standard axiom schemata dan Modus Ponens saja.
2. Ubah ekspresi logika di bawah ini dalam bentuk klausal :  
 $((\neg p \wedge q \mid r) \Leftrightarrow \neg t) \wedge (p \vee r)$
3. Buktikan bahwa ekspresi logika ini valid dengan menggunakan propositional resolution :  
 $(\neg r \Rightarrow ((\neg q \vee r) \Rightarrow (p \wedge \neg q \wedge \neg r))) \vee (\neg p \wedge \neg q \wedge \neg r)$   
Jika diberikan premis sebagai berikut :  
 $p \wedge q \Leftrightarrow s$   
 $\neg p \wedge \neg q \Leftrightarrow t$   
 $r \Leftrightarrow s \vee t$   
Buktikan bahwa kesimpulan di bawah ini adalah benar dengan menggunakan propositional resolution  
 $((\neg p \vee \neg q) \Rightarrow \neg(p \vee q)) \Rightarrow r$
4. Tuliskan apa output program di bawah ini (NILAI : 15)? Analisislah mengerjakan apa program tersebut di bawah ini (NILAI :15)

DOMAINS

list = integer\*

PREDICATES

secrets(list,list)

CLAUSES

secrets([],[]).

secrets([H | T], ProcessedTail):-

H &lt; 0,

!,

secrets(T, ProcessedTail).

secre

t

s

(

[

H

GOAL

|  
T  
]  
,

[  
H  
|  
P  
r  
o  
c  
e  
s  
s  
e  
d  
T  
a  
i  
1  
]  
)  
:  
-

S  
e  
c  
r  
e  
t  
s  
(  
T

,  
P  
r  
o  
c  
e  
s  
s  
e  
d  
T  
a  
i  
1  
)  
.

secrets([2,-45,3,-46], X).

5. Tuliskan apa output program di bawah ini (NILAI : 15)? Analisislah mengerjakan apa program tersebut di bawah ini (NILAI :15)

DOMAINS

list = integer\*

PREDICATES

secrets(list,list)

CLAUSES

secrets([],[]).

secrets([H | T], [H, H | DoubledTail]):-

    secrets(T, DoubledTail).

GOAL

secrets([1,2,3,4], X).

## UTS SEMESTER II – 2009/2010

Mata Kuliah : Logika Informatika

Dosen : Dr. Ir. Rila Mandala, M.Eng

Waktu : 90 menit

Sifat Ujian : Tutup Buku

*Notasi: Dalam UTS ini digunakan ~ untuk negasi, & untuk conjunction, | untuk disjunction, => untuk implication, <= untuk reduction, <=> untuk equivalence, A untuk universal quantifier dan E untuk existential quantifier.*

1. Buktiakan dengan menggunakan prinsip resolusi bahwa ekspresi logika di bawah ini adalah valid/tautology (NILAI : 10).
2. Diketahui sebagai berikut :
 

variables:  $x, y, z$   
           object constants: casper, Scooby, shaggy, homework, scoobysnacks  
           fungsi: mother, son, plus  
           relasi/predikat dengan aritas satu : ghost, blue, friendly, happy, cousin  
           relasi/predikat dengan aritas dua : scareOf, likes, does, friend

Tentukan apakah ekspresi logika di bawah ini adalah benar atau tidak secara sintaks dan berikan alasannya (Masing-masing bernilai 10, jadi total : 30).

  - a.  $\neg\text{happy}(z) \& \text{scareOf}(\text{shaggy}, \text{son}(\text{cousin}(\text{casper})))$
  - b.  $\text{blue}(\text{mother}(\text{son}(\text{casper}))) \mid \text{friendly}(\text{son}(\text{mother}(\text{shaggy}))) \Leftrightarrow \text{happy}(\text{scoby})$
  - c.  $\text{likes}(\text{Scooby}, \text{scoobysnacks}) \& \text{likes}(\text{shaggy}, \text{scoobysnacks}) \Rightarrow \text{friend}(\text{shaggy})$
3. Tentukan apakah ekspresi logika di bawah ini benar atau salah secara arti/makna (Masing-masing bernilai 10, jadi total : 50)
  - a.  $\exists x \exists y \exists z (x + y = z)$
  - b.  $\exists x \exists z \exists y (x + y = z)$
  - c.  $\forall y \forall z \forall x (x + y = z)$
  - d.  $\forall y \forall z \exists x (x + y = z)$
  - e.  $\exists x (\forall y (x + 0 = x) \& (x + 0 = 1))$
4. Jika diketahui
 

Relasi/predikat dengan aritas 1 : manusia  
           Relasi/predikat dengan aritas 2 :  
             takut( $x, y$ ) artinya  $x$  takut pada  $y$   
             ayah( $x, y$ ) artinya  $y$  adalah ayahnya  $x$ .

Terjemahkan dalam ekspresi logika fakta sebagai berikut (NILAI : 10) :

Ada orang yang takut hanya sama ayahnya saja.

SELAMAT BEKERJA

## UTS SEMESTER II – 2010/2011

IF 2050

90 menit

1. Ubah kalimat di bawah ini ke dalam representasi proposisional logic atau relational logic.  
Untuk representasi relational logic, gunakan proposisi dan konstanta pada tabel berikut:

| Proposisi                                | Relation constants                         |
|------------------------------------------|--------------------------------------------|
| c: Christine is singing soul             | kolamrenang(x): x adalah kolam renang      |
| j: Justin is singing pop                 | ramai(x): x tempat yang ramai              |
| k: Kelly is singing rock                 | penuhuap(x): x penuh uap                   |
| b: beyonce will sing rap                 | tidakmenyenangkan(x): x tidak menyenangkan |
| s: shakira will sing rap                 | mata(x): x adalah sebuah mata              |
| a: Atari is releasing a stalker game     | memiliki(x,y): x memiliki y                |
| n: Nintendo is releasing a stalker game  | manusia(x): x adalah manusia               |
| g: Sega is releasing a molester game     |                                            |
| m: Commodore is releasing a stalker game |                                            |

- a. Semua manusia memiliki dua mata (ubah ke relational logic)
  - b. Semua tempat yang ramai dan penuh uap selain kolam renang, tidak menyenangkan.  
(ubah ke relational logic)
  - c. If Christina is singing soul and Justin is singing pop are sufficient and necessary conditions for Kelly is singing rock, then neither Beyonce nor Shakira will sing rap.  
(ubah ke propositional logic)
  - d. It is not the case that both Atari's releasing a stalker game implies that Nintendo does and Sega's releasing a molester game implies that Commodore does. (ubah ke propositional logic)
2. Terdapat kumpulan premis dan kesimpulan dalam propositional logic sebagai berikut.
- a. Seorang bayi bukan manusia yang logis
  - b. Seseorang yang bisa menaklukkan buaya tidak dipandang rendah
  - c. Seseorang tidak logis dipandang rendah
- Kesimpulan: Seseorang yang bisa menaklukkan buaya bukan seorang bayi.  
Buktikan kesimpulan dapat diturunkan dari kumpulan premis dengan:
- a. Rule of inference dan/atau Axiom Schemata
  - b. Propositional
- Resolution Gunakan:
- |                                           |                    |
|-------------------------------------------|--------------------|
| b: seorang bayi                           | l: seseorang tidak |
| logis m: seseorang bisa menaklukkan buaya | d: dipandang       |
| rendah                                    |                    |
3. Diketahui fakta sebagai berikut:  
Penjual software bajakan adalah seorang kriminal  
Ketua kelas mempunyai beberapa software bajakan dan semuanya dibeli dari Gayus  
Buktikan bahwa Gayus adalah seorang kriminal dengan menggunakan relational proof tanpa memakai standard axioms.
4. Ubahlah ekspresi relasional logika ini menjadi bentuk klausal:  
 $\exists w. \forall x (\exists y. (\sim p(x, y) \wedge r(y)) \leftarrow \exists z. (q(w, z)))$

## UTS SEMESTER II – 2011/2012

Mata kuliah : IF2050 Logika Informatika

Kelas : K1 dan K2

Waktu : 100 menit

Sifat Ujian : Tutup buku

1. Buktikan dengan menggunakan prinsip resolusi bahwa ekspresi logika di bawah ini adalah valid/tautologi (NILAI : 10)

$$(\sim r \quad ((\sim q \vee r) \quad (p \wedge \sim q \wedge \sim r))) \vee (\sim p \wedge \sim q \wedge \sim r)$$

2. Diketahui sebagai berikut :

*variables:*  $x, y, z$ *object constant:* patrick, joe, kevinKW, PR, cemilan*function constant:* mother, anak, plus*relational constants dengan aritas satu:* hantu, biru, ramah, senang sepupu*relational constants dengan aritas dua:* takutpada, suka, mengerjakan, teman

Tentukanlah apakah ekspresi logika di bawah ini adalah benar atau tidak secara sintaks (bukan semantik), dan kemukakan alasannya mengapa? (NILAI: 15)

- (a)  $\sim \text{suka}(\text{joe}, \text{PR}) \vee \text{mengerjakan}(\text{joe}, \text{PR})$
- (b)  $\text{plus}(\text{joe}, z) \quad \text{suka}(\text{mother}(\text{kevinKW}), \text{joe})$
- (c)  $\text{hantu}(x) \wedge \text{ramah}(x) \quad \text{patrick}(x)$
- (d)  $\sim \text{senang}(z) \wedge \text{takutpada}(\text{kevinKW}, \text{anak}(\text{sepupu}(\text{patrick})))$
- (e)  $\text{biru}(\text{mother}(\text{anak}(\text{patrick}))) \vee \text{ramah}(\text{anak}(\text{mother}(\text{kevinKW}))) \Leftrightarrow \text{senang}(\text{joe})$
- (f)  $\text{suka}(\text{joe}, \text{cemilan}) \wedge \text{suka}(\text{kevinKW}, \text{cemilan}) \Rightarrow \text{teman}(\text{kwvinKW})$
- (g)  $\text{takutpada}(\text{joe}, \text{sepupu}(\text{patrick}))$
- (h)  $\text{teman}(\text{patrick}, \text{joe}) \wedge \sim \text{takutpada}(\text{kevinKW}, \text{hantu}(\text{patrick}))$
- (i)  $\text{suka}(\text{kevinKW}, \text{mother}(x)) \wedge \text{teman}(\text{kevinKW}, x) \Rightarrow \text{hantu}(x) \wedge \text{biru}(x)$
- (j)  $\text{plus}(\text{mother}(\text{anak}(z)), \text{anak}(\text{anak}(\text{cemilan}))) \wedge \text{plus}(\text{joe}, \text{kevinKW})$

3. Tentukan apakah ekspresi logika di bawah ini benar atau salah secara arti/makna (Masing- masing bernilai 5, jadi total NILAI : 25)

- a.  $\exists x. \exists y. \exists z. (x + y = z)$
- b.  $\exists x. \forall z. \exists y. (x + y = z)$
- c.  $\forall y. \forall z. \forall x. (x + y = z)$
- d.  $\forall y. \forall z. \exists x. (x + y = z)$

4. Jika diketahui

*Relational constants dengan aritas 1:* manusia*Relational constants dengan aritas 2:**takut(x, y)* artinya x takut pada y*ayah(x, y)* artinya y adalah ayahnya x.

Terjemahkan dalam ekspresi logika relasional fakta sebagai berikut (NILAI : 10) :

Ada orang yang takut hanya sama ayahnya saja

5. Diketahui fakta sebagai berikut:

Sesuatu di laut yang tidak diperhatikan, bukan putri duyung.

Sesuatu di laut yang dicatat pada log, berarti layak untuk diingat.

Sesuatu yang saya lihat di laut, tidak ada yang layak untuk diingat.

Sesuatu yang saya perhatikan di laut, adalah sesuatu yang saya catat di log.

Buktikan bahwa kesimpulan: "Sesuatu yang saya lihat di laut, bukanlah putri duyung; , dapat diturunkan dari kesimpulan fakta tersebut dengan memanfaatkan kaidah inferensi saja.

(NILAI: 15)

Gunakan proposisi sebagai berikut:

n: sesuatu di laut yang saya (di) perhatikan;

m: putri duyung;

l: sesuatu di laut yang dicatat di log;

r: sesuatu di laut yang layak untuk diingat;

i: sesuatu yang saya lihat di laut.

6. Diketahui fakta sebagai berikut:

$$\forall x.(p(x) \Rightarrow q(x)) \Rightarrow \exists x.(r(x) \wedge s(x))$$

$$\forall x.(p(x) \Rightarrow s(x)) \wedge \forall x.(s(x) \Rightarrow q(x))$$

Buktikan bahwa kesimpulan:  $\exists x.s(x)$  dapat ditarik dari kumpulan fakta tersebut dengan memanfaatkan kaidah inferensi dan/atau *axiom schema*. (NILAI: 15)

7. Buktiakan bahwa kesimpulan  $(\sim r \Rightarrow (\sim q \wedge \sim p)) \Rightarrow ((p \Rightarrow \sim r) \Rightarrow \sim p)$  dapat ditarik dari kumpulan fakta  $\{p \Rightarrow q, q \Rightarrow r\}$  dengan memanfaatkan *axiom schema* dan *modus ponens* saja (NILAI: 10)

Program Studi Teknik Informatika  
 Institusi Teknologi Bandung

---

UTS SEMESTER II – 2012/2013

Mata kuliah : IF2050 Logika Informatika  
 Kelas : K1 dan K2  
 Waktu : 100 menit  
 Ujian : Closed book

1. Tentukan apakah kalimat dalam logika proposisi berikut ini legal atau tidak legal (tidak perlu diberi alasan). (NILAI 10)
  - a. This  $V$  is  $\sim$  correct.
  - b.  $P \vee \sim q \wedge \sim p \vee \sim q \Rightarrow p \vee q$
  - c.  $\sim(q \vee r) \sim q \Rightarrow \sim \sim p$
  - d. Logic\_is\_fun <= get\_good\_grade  $\wedge$  not\_too\_much\_work
  - e. ((charlesIsWilliamFather  $\wedge$  dianaIsWilliamMother)  $\Rightarrow$  charlesLovesDiana)
2. Tentukan pernyataan terkait logika proposisi ini benar atau salah, jika salah tuliskan pernyataan yang benar. (NILAI 10)
  - a. Dalam metode pembuktian *truth tabel* (menggunakan dua tabel), kesimpulan dapat diturunkan dari kumpulan premis jika baris yang bernilai benar pada tabel kesimpulan merupakan himpunan bagian dari baris yang bernilai benar pada tabel premis.
  - b. Dalam metode pembuktian *validity checking*, kumpulan premis (jika terdapat lebih dari satu kalimat premis) dihubungkan dengan operator implikasi ( $\Rightarrow$ ).
  - c. Pada metode pembuktian resolusi, yang diubah menjadi klausa adalah seluruh kalimat premis dan konklusi (kesimpulan).
  - d. *Unsatifiability checking* memeriksa apakah kumpulan premis implikasi negasi konklusi untuk semua kemungkinan interpretasi, bernilai *false*.
  - e. Jika tidak terdapat premis-premis untuk diterapkan kaidah inferensi, maka *axiom schema* dapat diterapkan pada premis untuk membuktikan suatu kesimpulan.
3. Terdapat teka-teki sebagai berikut. (NILAI 10)  
 Terdapat tiga buah kotak, hanya satu kotak yang berisi emas, dan dua kotak lainnya kosong. Terdapat kertas yang ditempelkan pada setiap kotak yang berisi suatu petunjuk. Petunjuk di kotak 1 tertulis: "Emas tidak ada di sini.". Petunjuk di kotak 2 tertulis: "Emas tidak ada di sini.". Petunjuk di kotak 3 tertulis : "Ada emas di kotak ke-2.".  
 Hanya ada satu petunjuk yang benar, sedangkan dua petunjuk yang lain salah. Tentukan kotak yang berisi emas.  
 Representasikan dalam logika proposisi (maksimal 2 proposisi), dan gunakan tabel kebenaran untuk menjawab pertanyaan tersebut menggunakan *reverse evaluation*.
4. Terdapat premis sebagai berikut.  
 Seseorang yang pergi belajar selalu menyisir rambut.  
 Seseorang yang tidak pergi belajar tidak memiliki kontrol diri.  
 Seseorang tidak tampak menarik jika orang tersebut tidak rapi.  
 Seseorang yang menyisir rambut tampak menarik.

Buktikan bahwa kesimpulan : Jika seseorang memiliki kontrol diri maka orang tersebut rapi, dapat ditarik dari kumpulan premis tersebut, dengan menggunakan:

- a. Kaidah inferensi dan/atau skema aksiom (NILAI 10)
- b. Resolusi (NILAI 10)

Gunakan proposisi  $p, q, r, s$ , dan  $t$  sebagai berikut.

- p : seseorang pergi belajar;  
 q : seseorang menyisir rambut;  
 r : seseorang tampak menarik;  
 s : seseorang rapi;  
 t : seseorang memiliki kontrol diri.

5. Apakah ekspresi logika relasional di bawah ini legal atau tidak? Jika tidak, jelaskan di mana letak kesalahan dan sebabnya.
  - Konstanta : alif, habibie, buku
  - Fungsi : ayah (aritas 1), ibu (aritas 1), cinta (aritas 2)
  - Relasi : susahtidur (aritas 1), sayang (aritas 2), anak (aritas 2)
  - a. susahtidur(alif)
  - b. ayah(habibie)
  - c. sayang(habibie, buku)  $\wedge$  ayah(habibie)  $\Rightarrow$  cinta(habibie, buku)
  - d. sayang(ibu(anak(habibie, alif)), habibie)
  - e. susahtidur(ibu(ayah(ibu(x))))  $\Rightarrow$  sayang(x, anak(x, alif))
6. Ubahlah ekspresi logika relasional di bawah ini menjadi bentuk klausal.  
 $\sim\exists.(g(x) \wedge \forall z.(r(z) \Rightarrow f(x,z)))$
7. “Batman menangkap semua penjahat. Joker melakukan sesuatu aktivitas yang ilegal. Orang yang melakukan aktivitas yang ilegal disebut penjahat. Jika seseorang ditangkap oleh Batman, dan melakukan aktivitas ilegal, maka dia masuk penjara.”

Relasi :

menangkap(x) artinya “Batman menangkap x”  
 melakukan(x, y) artinya “seseorang x melakukan aktivitas y”  
 penjahat(x) artinya “orang x adalah penjahat”  
 ilegal(x) artinya “aktivitas x adalah ilegal”  
 dipenjara(x) artinya “x dipenjara”

- a. Ubahlah fakta-fakta di atas menjadi ekspresi logika relasional hanya dengan menggunakan relasi yang disebutkan di atas saja. (NILAI 20)
- b. Buktikan bahwa “Joker masuk penjara” dengan menggunakan *relational proof* dan hanya menggunakan metode EI, UI, AI, AE, dan MP, tanpa menggunakan standard *axiom schemata*.

--- SELAMAT MENGERJAKAN ---

## UTS SEMESTER I – 2014/2015

Program Studi Teknik Informatika  
Institut Teknologi Bandung

UJIAN TENGAH SEMESTER GANJIL 2014/2015

Matakuliah : IF2121 Logika Informatika

Kelas : K1 dan K2

Waktu : 100 menit

Sifat Ujian : Closed-book.

1. Buktiakan bahwa penalaran logika proposisi di bawah ini valid atau tidak. Gunakan pembuktian dengan kaidah inferensi, *axiom schema*, atau resolusi. Tidak diperkenankan menggunakan *semantic reasoning*. (Masing-masing bernilai 15)
  - a. If time stretches back infinitely, then today wouldn't have been reached.  
If today wouldn't have been reached, then today wouldn't exist.  
Today exists.  
If time doesn't stretch back infinitely, then there was a first moment of time.  
So, There was a first moment of time.
  - b. If the butler shot Jones, then he knew how to use a gun.  
If the butler was a former marine, then he knew how to use a gun.  
The butler was a former marine.  
So, the butler shot Jones.
  - c. If Socrates escapes from jail, then he's willing to obey the state only when it pleases him.  
If he's willing to obey the state only when it pleases him, then he doesn't really believe what he says and he's inconsistent.  
*kesimpulan*: If Socrates really believes what he says, then he won't escape from jail.
2. Translasikan kalimat berikut ini menjadi kalimat *relational logic*. Gunakan relasi:  
 $I(x)$ : bilangan integer;  $E(x)$ : bilangan genap;  $O(x)$ : bilangan ganjil
  - a. Semua bilangan integer adalah bilangan genap. (Nilai 5)
  - b. Beberapa bilangan integer adalah bilangan ganjil. (Nilai 5)
  - c. Bilangan disebut bilangan genap hanya jika bilangan tersebut adalah bilangan integer. (Nilai 5)
  - d. Dua adalah bilangan integer genap. (Nilai 5)
3. Terdapat kumpulan premis sebagai berikut! Cek dikatakan tidak berlaku jika tidak diuangkan selama 30 hari. Anda tidak dapat menguangkan cek yang tidak berlaku. Cek ini tidak diuangkan selama 30 hari. Buktiakan bahwa ada cek yang tidak bisa diuangkan, dengan menggunakan Kaidah Inferensi saja (Rule of Inference). Gunakan relasi:  
 $C(x)$ :  $x$  adalah Cek;  $T(x)$ :  $x$  sudah diuangkan dalam 30 hari;  $V(x)$ :  $x$  tidak berlaku;  
 $S(x)$ :  $x$  dapat diuangkan;  
 dan gunakan konstansa objek cek\_ini untuk translasi ke bentuk *relational logic*. (Nilai 20)
4. Ubahkan kalimat relational logic berikut ini ke dalam representasi klausula. (Nilai 20)

$$\forall x (f(x) \Rightarrow \exists y (m(x,y))) \wedge \forall x \forall z ((a(x,z) \vee b(x,z)) \wedge c(z,3))$$

— SELAMAT MENGERJAKAN —

**UTS SEMESTER I – 2016/2017**  
**Mata Kuliah: IF2121 Logika Informatika**  
**Kelas: K1, K2, dan K3**  
**Waktu: 100 menit**  
**Sifat Ujian: Closed-book**

---

**Bagian Propositional Logic**

1. Translasikan kalimat dalam bahasa alami berikut ini ke dalam representasi logika proposisi. Tentukan dan tuliskan representasi proposisi yang anda gunakan di setiap butir soal. (Nilai 20)
  - a. Jika hujan, salju, dan hujan es tidak menghambat saya untuk meletakkan surat pada kotak pos anda, maka jika saya bukan seorang tukang pos ataupun sosiopat, berarti saya adalah seorang kawan yang baik.
  - b. "A necessary condition of an argument being valid is that it be deductive."
  - c. Ang mengatakan bahwa kita harus memaafkan, tapi Ang tidak memaafkan baik Beng maupun Cing, dan Ang menghukum keturunan mereka hingga hari ini.
  - d. "If Christina is singing soul and Justin is singing pop are necessary and sufficient conditions for Kelly is singing rock, then neither Beyoncé nor Shakira will sing rap."
2. Buktikan bahwa  $\{p \Rightarrow \neg q, \neg q \Rightarrow (r \wedge s)\} \models ((p \Rightarrow \neg(r \wedge s)) \Rightarrow \neg p)$  dengan menggunakan Axiom Schemata dan Modus Ponens saja, dan tidak menerapkan Deduction Theorem. (Nilai 10)
3. Terdapat kumpulan premis sebagai berikut.
  - Jika  $p$  adalah bilangan prima terbesar, maka  $n$  (suatu variabel yang kita tetapkan) adalah suatu nilai hasil perkalian semua bilangan prima lebih kecil dari  $p$  ditambah dengan satu.
  - Jika  $n$  adalah suatu nilai hasil perkalian semua bilangan prima lebih kecil dari  $p$  ditambah dengan satu, maka  $n$  adalah bilangan prima atau  $n$  bilangan yang memiliki faktor prima yang lebih besar dari  $p$ .
  - Jika  $n$  adalah bilangan prima, maka  $p$  bukan bilangan prima terbesar.
  - Jika  $n$  adalah bilangan yang memiliki faktor prima yang lebih besar dari  $p$ , maka  $p$  bukan bilangan prima terbesar.

Gunakan proposisi  $p$  ( $p$  adalah bilangan prima terbesar),  $q$  ( $n$  adalah suatu nilai hasil perkalian semua bilangan prima lebih kecil dari  $p$  ditambah dengan satu),  $r$  ( $n$  adalah bilangan prima), dan  $s$  ( $n$  bilangan yang memiliki faktor prima yang lebih besar dari  $p$ ) untuk membuktikan dengan metode propositional resolution bahwa  $p$  bukan bilangan prima terbesar. (Nilai 20)

**Bagian Relational Logic/Predicate Logic**

4. Jika ada
  - relational constant dengan arity 1:  $orang(x)$ , artinya  $x$  adalah orang
  - relational constant dengan arity 2:  $cinta(x,y)$ , artinya  $x$  mencintai  $y$   
 $kekasih(x,y)$ , artinya  $y$  adalah kekasihnya  $x$

Terjemahkanlah kalimat dalam bahasa alami di bawah ini ke dalam relational logic.

*Ada orang yang tidak cinta pada siapapun kecuali pada kekasihnya saja.* (Nilai 10)

5. Diketahui sebagai berikut

- *variable: x, y, z*
- *object constants: c, sc, sh, ss, hw*
- *function constants: m, s, p*
- *relational constants dengan arity satu: g, b, h, cn, fd*
- *relational constants dengan arity dua: so, l, d, f*

Tentukanlah apakah pernyataan logika relasi di bawah ini benar atau salah menurut kaidah logika relasional. Jika menurut anda salah maka harus dituliskan penyebab kesalahannya. Jika penyebab kesalahan yang anda sebutkan salah, maka nilai menjadi 0. (Nilai total 30)

- a.  $p(s, z) \Rightarrow l(m(sh), sc)$
- b.  $g(x) \wedge f(x) \Rightarrow c(x)$
- c.  $\neg h(z) \wedge so(sh, s(cn(c)))$
- d.  $l(sc, ss) \wedge l(sh, ss) \Rightarrow f(sh)$
- e.  $so(sc, cn(c))$
- f.  $f(c, sc) \wedge \neg so(sh, g(c))$
- g.  $p(m(s(z)), s(s(ss))) \wedge p(sc, sh)$

6. Diketahui ada 4 buah relasi/predikat (*melakukan, korupsi, penjahat, ditangkap, dipenjara*) dan sebuah konstanta *gayus*. Diberikan 4 buah premis di bawah ini

- $\forall x. \forall y. (melakukan(x, y) \wedge korupsi(y) \Rightarrow penjahat(x))$
- $\forall x. (penjahat(x) \Rightarrow ditangkap(x))$
- $\exists x. (melakukan(gayus, x) \wedge korupsi(x))$
- $\forall x. \forall y. (ditangkap(x) \wedge melakukan(x, y) \wedge (korupsi(y) \rightarrow dipenjara(x)))$

Buktikanlah dengan menggunakan relational proof kesimpulan di bawah ini (Nilai 10)

$dipenjara(gayus)$

## UTS SEMESTER I – 2017/2018

Bagian Propositional Logic

Open with ▾

1. Translasikan kalimat dalam bahasa alami berikut ini ke dalam proposisi. (Nilai 20)
- Ketika kamu menendang bola dan dipotong oleh lawan maka temanmu tidak akan mendapatkan bola tersebut. Ternyata kamu menendang bola. Tendanganmu tidak dipotong oleh lawan. Tentu temanmu mendapatkan bola tersebut.
  - Government's function is to protect life, liberty, and the pursuit of happiness. The British colonial government doesn't protect these. The only way to change it is by revolution. If government's function is to protect life, liberty, and the pursuit of happiness and the British colonial government doesn't protect these, then the British colonial government ought to be changed. If the British colonial government ought to be changed and the only way to change it is by revolution, then we ought to have a revolution. We ought to have a revolution.  
[Pergunakan proposisi g, b, o, c, dan r. Ini adalah alasan terjadinya American Declaration of Independence.]
  - Jika pelayan menembak Badu maka dia tahu cara menggunakan senjata. Jika pelayan mantan tentara dia tahu cara menggunakan senjata. Ternyata pelayan tersebut mantan tentara. Kesimpulannya bahwa pelayan menembak Badu.
  - Either Socrates's death will be perpetual sleep, or if the gods are good then his death will be an entry into a better life. If Socrates's death will be perpetual sleep, then he shouldn't fear death. If Socrates's death will be an entry into a better life, then he shouldn't fear death.  
So, Socrates shouldn't fear death. [ Pergunakan proposisi p, g, b, and f ]
2. Buktikan kesimpulan pada soal nomor 1a dengan menggunakan teknik *unsatisfiability checking*. (Nilai 10)
3. Terdapat kumpulan premis sebagai berikut:  
 Kayla tidak suka bermain tapi suka membaca. Kayla suka bermain jika punya banyak teman.  
 Kayla suka olahraga dan bercocok tanam. Jika Kayla tidak punya banyak teman tapi suka olahraga maka Kayla bahagia.  
 Buktikan bahwa **Kayla bahagia atau hidup sehat** dengan menggunakan **kaidah inferensi saja** (tanpa *axiom schema*), dengan menggunakan proposisi:  
 s: Kayla bahagia; t: Kayla hidup sehat; p: Kayla suka bermain; q: Kayla suka olahraga; r: Kayla suka bercocok tanam; m: Kayla punya banyak teman; n: Kayla suka membaca. (Nilai 15)

Bagian Relational Logic/ Predicate Logic/ First Order Logic

4. Jika diketahui bahwa cinta, pria, wanita, menikah adalah relasi, dan ayah, kakek adalah fungsi, operator  $\rightarrow$  adalah implikasi (jika-maka), operator  $\wedge$  adalah conjunction (and), dan  $\leftrightarrow$  adalah jika dan hanya jika (ekuivalen), periksalah apakah ekspresi logika relasi di bawah ini secara *syntax* benar atau salah. Jika salah kemukakanlah alasannya mengapa salah. (Nilai : 15)
- Untuk merepresentasikan fakta : jika x adalah ayahnya si y adalah ayahnya si z maka x itu adalah kakeknya si z. Ekspresi logikanya sebagai berikut :  

$$\text{ayah}(x,y) \wedge \text{ayah}(y,z) \rightarrow \text{kakek}(x,z)$$
  - Untuk merepresentasikan fakta : pria x dan wanita y saling mencintai jika dan hanya jika x menikah dengan y. Ekspresi logikanya sebagai berikut :  

$$\text{cinta}(\text{pria}(x), \text{wanita}(y)) \leftrightarrow \text{menikah}(x,y)$$
5. Jika diketahui terdapat *unary relation constant*: informatikawan, filsuf; *binary relation constant*: suka(x,y) adalah x menyukai y, *object constant*: russel; ubahlah kalimat-kalimat di bawah ini ke dalam bentuk *Relational Logic*. (Nilai 10)
- Setiap informatikawan yang menyukai dirinya sendiri juga menyukai Russel.

- b. Tidak ada filsuf yang menyukai Russel yang juga menyukai dirinya sendiri.  
c. Beberapa filsuf menyukai semua informatikawan.  
d. Setiap informatikawan yang menyukai Russel adalah seorang filsuf.
6. Jika diketahui bahwa semua universitas di singapura lebih bagus daripada semua universitas di Indonesia. Ada universitas di propinsi Jawa Barat yang lebih baik daripada seluruh universitas di Malaysia. Buktikanlah bahwa NTU (Nanyang Technological University di Singapura) lebih baik dari University of Technology Malaysia (UTM) dengan menggunakan relational proof.  
Pergunakanlah hanya 5 buah relasi di bawah ini saja (anda tidak boleh membuat relasi yang lain). Tuliskanlah semua premis baik yang tersurat (tertulis) maupun yang tersirat (tidak tertulis) dari paparan di atas yang diperlukan untuk membuktikan kesimpulan di atas.  
 $\text{singUni}(x)$  : artinya  $x$  adalah universitas di Singapura.  
 $\text{indoUni}(x)$  : artinya  $x$  adalah universitas di Indonesia.  
 $\text{malayUni}(x)$  : artinya  $x$  adalah universitas di Malaysia.  
 $\text{jabarUni}(x)$  : artinya  $x$  adalah universitas di Jawa Barat.  
 $\text{lebihbaik}(x,y)$  : artinya  $x$  adalah universitas yang lebih bagus daripada universitas  $y$ .  
(NILAI : 20)
7. Ubahlah kalimat relational logic berikut ini ke dalam bentuk klausa. (Nilai 10)  
$$\forall x. \exists y. \forall z. ((q(x, y)) \leftarrow (\forall x. (\neg p(x, z) \wedge r(x))))$$

**UTS SEMESTER I – 2019/2020****Mata Kuliah:** IF2121 Logika Komputasional**Kelas:** K1, K2, dan K3**Waktu:** 100 menit**Sifat Ujian:** Closed-book**Bagian Propositional Logic**

1. Translasikan kalimat dalam bahasa alami berikut ini ke dalam representasi logika proposisi. (Nilai 20)
  - a. To get a bachelor's degree, you only need to study at ABC University.
  - b. Tidak benar bahwa Popi bisa berbahasa Jepang atau bahasa Mandarin, tetapi tidak bahasa Inggris.
  - c. The system is in a multiuser state if and only if it is operating normally. If the system is operating normally, the kernel is functioning. Either the kernel is not functioning or the system is in interrupt mode. If the system is not in multiuser state, then it is in interrupt mode. The system is not in interrupt mode.
  - d. Jika pistol ada di gudang, maka kami melihatnya saat kami membersihkan gudang. Si pembunuh melakukan kejahatannya di garasi atau di dalam Rumah. Jika si pembunuh melakukan kejahatannya di garasi, maka pistolnya ada di dalam kardus hijau. Kami tidak melihat pistol saat kami membersihkan gudang. Jika pembunuh melakukan kejahatannya di luar Rumah, maka kami tidak dapat menemukan pistolnya. Jika pembunuh melakukan kejahatannya di dalam Rumah, maka pistol ada di gudang.
2. Pada soal ini, buktikan bahwa kesimpulan dapat diturunkan dari kumpulan premis dengan memanfaatkan kaidah inferensi (*rule of inference*) tanpa *axiom schemata*. (Nilai 10)
  - a. Premis:  $p \wedge (q \vee r); s \wedge p$   
Kesimpulan:  $(q \wedge s) \vee (r \wedge s)$
  - b. Premis: Kesimpulan  $(p \wedge q) \vee (p \wedge r); (p \vee s) \rightarrow t$   
Kesimpulan:  $t$
3. Diketahui premis di bawah ini : (Nilai 20)
 
$$\begin{array}{ll} p \wedge q & s \\ \sim p \wedge \sim q & t \\ r & s \vee t \end{array}$$

Buktikan dengan propositional resolution bahwa :  $((\sim p \vee \sim q) \rightarrow (\sim(p \vee q)) \rightarrow r)$

**Bagian Relational Logic/Predicate Logic/First Order Logic**

4. Diberikan definisi seperti di bawah ini: (Nilai 10)

*variables:*  $x, y, z$ *object constants:* giant, nobita, suneo, pr, donat*function constants:* ibu, anak, tambah*relational constants dengan artitas 1:* kunitlanak, sedih, ramah, senang, sepupu*relational constants dengan artiras 2:* takut, menyukai, mengerjakan, teman

Tentukan secara sintaks kalimat-kalimat dalam logika relasional di bawah ini benar atau tidak. Jika jawabannya tidak, sebutkan alasannya.

- (a) teman(giant, nobita)  $\wedge \sim$ takut(suneo, kunitlanak(giant))
- (b) tambah(ibu(anak(z))), anak(anak(anak(donat))))  $\wedge$  tambah(nobita, suneo)

5. Diketahui terdapat

*unary relation constant:* orang, selebritis, pejabat

*binary relation constant:* kenal(x, y) adalah x mengenal y; berebda(x, y) adalah x berbeda dengan y.

*object constant:* maudy, tulus

Ubahlah kalimat-kalimat di bawah ini ke dalam bentuk *Relational Logic*: (Nilai 10)

- a. Tidak ada selebritis atau orang yang mengenal pejabat manapun.
- b. Setiap orang pasti mengenal tiga selebritis.

Ubahlah *relational logic* berikut menjadi kalimat biasa: (Nilai 10)

- c.  $\exists x (\text{pejabat}(x) \wedge \text{kenal}(x, \text{Maudy}) \wedge \neg \text{kenal}(x, \text{Tulus}))$
- d.  $\exists x \forall y (\text{selebritis}(x) \wedge \text{pejabat}(x) \wedge \text{orang}(y) \wedge \text{kenal}(x, y))$

6. Buktikan bahwa kesimpulan berikut ini dapat diturunkan dari kumpulan premis yang ada dengan memanfaatkan *relational proof*. (Nilai 20)

- a. Premis:  $\forall x. (p(x) \rightarrow q(x)); \exists x. (p(x) \wedge r(x))$

Kesimpulan:  $\exists x. (r(x) \wedge q(x))$

- b. Premis:  $(\forall x. (p(x) \rightarrow q(x))) \wedge (\exists x. (r(x) \wedge s(x))); (\forall x. (p(x) \rightarrow s(x))) \wedge (\forall x. (s(x) \rightarrow q(x)))$

Kesimpulan:  $\exists x. (s(x))$

--- SELAMAT MENIKMATI ---

## UTS Logkom 2020/2021

### 12 Oktober 2020

#### **MEMBENTUK KALIMAT LOGIKA PROPOSISI**

1. Jika  $f$  : get married,  $s$  : poor,  $h$  : work harder,  $r$  : save money. Maka ubahlah kalimat ini ke dalam ekspresi logika proposisi. "You can get married if you are not poor, but otherwise you must work harder and save money".

(Petunjuk : Pilih satu jawaban)

- o  $(\sim s \rightarrow f) \vee (s \rightarrow (h \wedge r))$
- o  $(f \rightarrow \sim s) \vee (s \rightarrow (h \wedge r))$
- o  $\sim(\sim s \rightarrow f) \rightarrow (h \wedge r)$
- o  $(f \rightarrow \sim s) \wedge (s \rightarrow (h \wedge r))$
- o  $(\sim s \rightarrow f) \wedge ((h \wedge r) \rightarrow s)$
- o  $(\sim s \rightarrow f) \wedge (s \rightarrow (h \wedge r))$
- o Jawaban di pilihan kesatu dan kedua benar
- o Jawaban di pilihan ketiga dan keempat benar
- o Jawaban di pilihan kelima dan keenam benar
- o Semua pilihan jawaban salah

2. "When I study at ITB campus, I like to do coding only if I can listen the music as well."  $s = I$ ,  $h = I$  am at ITB campus,  $d = I$  like to do coding,  $c = I$  can listen music. Ubahlah kalimat di atas ke dalam logika proposisi.

(Petunjuk : Pilih satu jawaban)

- o  $(s \wedge h) \rightarrow (c \rightarrow d)$
- o  $(s \wedge h) \rightarrow (c \wedge d)$
- o  $(s \wedge h) \wedge (c \rightarrow d)$
- o  $(s \rightarrow h) \wedge (c \rightarrow d)$
- o  $(s \wedge h) \rightarrow (c \vee d)$
- o  $(s \vee h) \rightarrow (c \vee d)$
- o  $(s \rightarrow h) \wedge (c \rightarrow d)$
- o  $(s \wedge h) \wedge (d \rightarrow c)$
- o  $(s \wedge h) \vee (d \rightarrow c)$
- o  $(s \wedge h) \vee (d \wedge c)$
- o  $(s \wedge h) \wedge (d \wedge c)$
- o  $(s \wedge h) \rightarrow (d \rightarrow c)$

#### **LOGIKA PROPOSISI : SEMANTICS**

3. Ada 2 tipe manusia : Knight dan Knave. Knight selalu berkata benar. Knave selalu berkata bohong. Ada 2 orang inggris dating ke kampus ITB. Kedua orang tersebut bernama James Bond dan Ethan Hunt. Si James Bond berkata "Ethan Hunt is a knight". Si Ethan Hunt berkata " Two of us are opposite types". Kesimpulannya adalah :

(Petunjuk : Pilih satu jawaban)

- o James Bond adalah Knight dan Ethan Hunt adalah Knight
- o James Bond adalah Knave dan Ethan Hunt adalah Knave
- o James Bond adalah Knight dan Ethan Hunt adalah Knave
- o James Bond adalah Knave dan Ethan Hunt adalah Knight

#### **LOGIKA PROPOSISI : PROOF METHOD**

Standard Axiom Schemata

1. Implication Introduction (II) :  $A \rightarrow (B \rightarrow A)$
2. Implication Distribution (ID) :  $A \rightarrow (B \rightarrow C) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$
3. Contradiction Realization (CR) :
  - $(A \rightarrow \neg B) \rightarrow ((A \rightarrow B) \rightarrow \neg A)$
  - $(\neg A \rightarrow B) \rightarrow ((\neg A \rightarrow \neg B) \rightarrow A)$
4. Equivalence (EQ) :
  - $(A \leftrightarrow B) \rightarrow (A \rightarrow B)$
  - $(A \leftrightarrow B) \rightarrow (B \rightarrow A)$
  - $(A \rightarrow B) \rightarrow ((B \rightarrow A) \rightarrow (A \leftrightarrow B))$
5. Other operators :
  - $(A \leftarrow B) \leftrightarrow (B \rightarrow A)$
  - $(A \vee B) \leftrightarrow (\neg A \rightarrow B)$
  - $(A \wedge B) \leftrightarrow \neg(\neg A \vee \neg B)$

4. Terdapat dua buah premis  $(q \rightarrow p) \wedge (r \rightarrow q)$  dan  $\neg((p \wedge q) \vee \neg(q \rightarrow r))$ . Berikut ini adalah penerapan ‘Inference Rules’ atau ‘Axiom Schemata’ yang dimungkinkan pada premis tersebut.

(Petunjuk : Pilih benar/salah untuk tiap pernyataan)

| Pernyataan                                                                                                                                                                | Jawaban     |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|
| Penerapan II pada premis pertama menghasilkan : $(q \rightarrow p) \rightarrow ((r \rightarrow q) \rightarrow (q \rightarrow p))$                                         | Benar/Salah |
| Penerapan Hukum De Morgan pada premis kedua menghasilkan: $\neg(p \wedge q) \vee (q \rightarrow r)$                                                                       | Benar/Salah |
| Penerapan II pada premis pertama menghasilkan:<br>$((q \rightarrow p) \wedge (r \rightarrow q)) \rightarrow (r \rightarrow ((q \rightarrow p) \wedge (r \rightarrow q)))$ | Benar/Salah |
| Penerapan Simplification (And Elimination) pada premis pertama menghasilkan:<br>$(q \rightarrow p)$                                                                       | Benar/Salah |
| Penerapan Hukum De Morgan pada premis kedua menghasilkan:<br>$(\neg p \vee \neg q) \wedge (q \rightarrow r)$                                                              | Benar/Salah |

5. Terdapat dua buah premis  $(q \rightarrow p)$  dan  $(r \rightarrow q)$ . Jika diinginkan untuk mendapatkan ekspresi  $\neg s \vee (\neg r \vee p)$  dalam empat langkah (atau empat baris, penghitungan baris tidak melibatkan baris premis pertama dan premis kedua). Maka urutan penerapan ‘Inference Rules’ atau ‘Axiom Schemata’ adalah sebagai berikut.
- (Petunjuk: Berikan tanda (v) untuk menjawab, tiap kolom langkah hanya boleh ada 1 tanda)

|                        | Langkah ke-1 | Langkah ke-2 | Langkah ke-3 | Langkah ke-4 | Tidak diperlukan |
|------------------------|--------------|--------------|--------------|--------------|------------------|
| Disjunctive Syllogism  |              |              |              |              |                  |
| Hypothetical Syllogism |              |              |              |              |                  |
| Modus Ponens           |              |              |              |              |                  |
| Modus Tolens           |              |              |              |              |                  |
| Material Implication   |              |              |              |              |                  |
| De Morgan              |              |              |              |              |                  |
| CR                     |              |              |              |              |                  |
| II                     |              |              |              |              |                  |

6. Terdapat sebuah ekspresi  $\forall w. \forall z. (p(w,y) \vee q(y,z) \vee \exists x. (s(z,x)))$ . Jika diterapkan EI (Existential Instantiation) pada ekspresi tersebut, maka berikut ini ekspresi yang bisa dihasilkan.

(Petunjuk : Pilih benar/salah untuk tiap pernyataan)

| Pernyataan                                                 | Jawaban     |
|------------------------------------------------------------|-------------|
| $\forall w. \forall z. (p(w,y) \vee q(y,z) \vee (s(z,w)))$ | Benar/Salah |

|                                                               |             |
|---------------------------------------------------------------|-------------|
| $\forall w. \forall z. (p(w,y) \vee q(y,z) \vee (s(z,a)))$    | Benar/Salah |
| $\forall w. \forall z. (p(w,y) \vee q(y,z) \vee (s(z,f(a))))$ | Benar/Salah |
| $\forall w. \forall z. (p(w,y) \vee q(y,z) \vee (s(f(z,a))))$ | Benar/Salah |

7. Terdapat dua buah premis  $\forall x. (p(x) \rightarrow q(x))$  dan  $p(a)$ . Jika diinginkan untuk mendapatkan ekspresi  $\neg \exists x. (\neg p(x) \wedge \neg q(x))$  dalam lima langkah (atau lima baris, penghitungan baris tidak melibatkan baris premis pertama dan premis kedua), maka urutan penerapan ‘Inference Rules’ atau ‘Axiom Schemata’ adalah sebagai berikut.

(Petunjuk: Berikan tanda (v) untuk menjawab, tiap kolom langkah hanya boleh ada 1 tanda)

|              | Langkah ke-1 | Langkah ke-2 | Langkah ke-3 | Langkah ke-4 | Langkah ke-5 | Tidak diperlukan |
|--------------|--------------|--------------|--------------|--------------|--------------|------------------|
| CR           |              |              |              |              |              |                  |
| UI           |              |              |              |              |              |                  |
| UG           |              |              |              |              |              |                  |
| EG           |              |              |              |              |              |                  |
| ED           |              |              |              |              |              |                  |
| Modus Ponens |              |              |              |              |              |                  |
| Modus Tolens |              |              |              |              |              |                  |
| Addition     |              |              |              |              |              |                  |

## LOGIKA PROPOSISSI: RESOLUTION

8. Terdapat sekumpulan premis sebagai berikut. Andi selalu mencuci tangan atau memakai masker. Andi suka berjabat tangan jika memakai masker, namun dia tidak suka berjabat tangan. Jika Andi selalu mencuci tangan maka dia suka berolahraga. Andi suka berolahraga jika dia sehat. Buktiakan Andi sehat, dengan menggunakan prinsip propositional resolution! (Petunjuk: Ini soal esai)
9. Perhatikan penggunaan prinsip resolusi berikut. Apakah penerapan resolusi tsb sudah benar? Jika tidak, di mana letak kesalahannya dan perbaiki langkah pembuktian tersebut. (Petunjuk: Ini soal esai)
1.  $\{p, \neg q, r\}$  premise
  2.  $\{\neg p, r\}$  premise
  3.  $\{q, s, t\}$  premise
  4.  $\{\neg r, \neg s\}$  premise
  5.  $\{\neg t\}$  premise
  6.  $\{p, r, s, t\}$  1, 3
  7.  $\{r, s, t\}$  6, 2
  8.  $\{t\}$  7, 4
  9.  $\{\}$  5, 8

## MEMBENTUK KALIMAT LOGIKA

10. “mahasiswa dan mahasiswi boleh menghadiri syukwis.”

Kalimat logika yang paling tepat untuk kalimat tersebut adalah: (Petunjuk : Pilih satu jawaban)

- o  $\forall x. ((\text{mahasiswa}(x) \vee \text{mahasiswi}(x)) \wedge \text{hadir}(x, \text{syukwis}))$
- o  $\forall x. ((\text{mahasiswa}(x) \wedge \text{mahasiswi}(x)) \rightarrow \text{hadir}(x, \text{syukwis}))$
- o  $\forall x. ((\text{mahasiswa}(x) \wedge \text{mahasiswi}(x)) \vee \text{hadir}(x, \text{syukwis}))$
- o  $\forall x. (\text{mahasiswa}(x) \rightarrow \text{hadir}(x, \text{syukwis})) \wedge \forall y. (\text{mahasiswi}(y) \rightarrow \text{hadir}(y, \text{syukwis}))$

11. “Manusia yang disiplin, setiap waktu selalu ingat tugasnya.”

Kalimat logika yang paling tepat untuk kalimat alami tersebut: (Petunjuk : Pilih satu jawaban)

- $\forall y. \exists x. ((waktu(x) \wedge disiplin(y)) \rightarrow ingat(y, tugas, x))$
- $\forall y. \exists x. (waktu(x) \rightarrow (disiplin(y) \wedge ingat(y, tugas, x)))$
- $\forall x. \exists y. ((waktu(x) \wedge disiplin(y)) \wedge ingat(y, tugas, x))$
- $\forall x. \exists y. (waktu(x) \rightarrow (disiplin(y) \rightarrow ingat(y, tugas, x)))$

12. "Sekurang-kurang ada dua orang yang menghadiri pesta dengan anaknya"

Kalimat logika yang paling tepat untuk kalimat tersebut adalah: (Petunjuk : Pilih satu jawaban)

- $\exists x. \exists y. ((hadir(x, pesta) \wedge hadir(y, pesta)) \wedge \neg(sama(x, y)))$
- $\exists x. \exists y. ((hadir(x, pesta) \vee hadir(y, pesta)) \wedge (sama(x, y)))$
- $\exists x. \exists y. ((hadir(x, pesta) \wedge hadir(y, pesta)) \rightarrow \neg(sama(x, y)))$
- $\exists x. \exists y. ((hadir(x, pesta) \vee hadir(y, pesta)) \rightarrow \neg(sama(x, y)))$

13. Belajar rajin belum tentu mendapat nilai A, tetapi ada yang mendapat nilai A tanpa belajar rajin.

Kalimat logika yang paling tepat untuk kalimat tersebut adalah: (Petunjuk : Pilih satu jawaban)

(misalkan p: belajar; q: mendapat)

- $\forall x. (p(x, \text{rajin}) \wedge (q(x, a) \vee \neg q(x, a)) \rightarrow \exists x. (\neg p(x, \text{rajin}) \wedge q(x, a)))$
- $\forall x. (p(x, \text{rajin}) \rightarrow (q(x, a) \vee \neg q(x, a)) \vee \exists x. (\neg p(x, \text{rajin}) \wedge q(x, a)))$
- $\forall x. (p(x, \text{rajin}) \rightarrow (q(x, a) \vee \neg q(x, a)) \wedge \exists x. (\neg p(x, \text{rajin}) \wedge q(x, a)))$
- $\forall x. (p(x, \text{rajin}) \wedge (q(x, a) \vee \neg q(x, a)) \vee \exists x. (\neg p(x, \text{rajin}) \wedge q(x, a)))$

## RELATIONAL: SEMANTICS

14. Misalkan diberikan bahasa dengan objek konstan a dan b, dan relasi unary p, dan binary q.

Asumsikan  $p(b) = \text{true}$ ;  $q(a, b) = \text{true}$ ; dan  $q(b, a) = \text{true}$ , selain interpretasi tersebut dianggap false.

Tentukan nilai kebenaran dari kalimat berikut: (Petunjuk : Pilih benar/salah untuk tiap pernyataan)

| Pernyataan                                                                     | Jawaban     |
|--------------------------------------------------------------------------------|-------------|
| $\forall x. \exists y. (p(x) \rightarrow q(x, y))$                             | Benar/Salah |
| $\forall x. \exists y. (p(x) \wedge q(x, y)) \rightarrow \exists y. p(y)$      | Benar/Salah |
| $(\exists y. \forall x. q(x, y)) \vee (\forall x. \exists y. q(x, y))$         | Benar/Salah |
| $\forall x. \exists y. (p(x) \rightarrow q(x, y)) \rightarrow \forall y. p(y)$ | Benar/Salah |

15. Konsistensi

Suatu kalimat  $\phi$  dikatakan konsisten terhadap kalimat-kalimat dalam himpunan  $\Delta$  jika dan hanya jika ada interpretasi yang memenuhi semua kalimat dalam  $\Delta \cup \{\phi\}$ .

Jika diketahui

- $\text{on}(b, a);$
- $\text{on}(a, c);$
- $\forall x. \forall z. (\text{above}(x, z) \leftrightarrow \text{on}(x, z) \vee \exists y. (\text{above}(x, y) \wedge \text{above}(y, z))).$

Tentukan apakah kalimat berikut konsisten terhadap kalimat tentang on dan above:

(Petunjuk : Pilih benar/salah untuk tiap pernyataan)

| Pernyataan           | Jawaban     |
|----------------------|-------------|
| $\text{above}(a, a)$ | Benar/Salah |
| $\text{above}(a, b)$ | Benar/Salah |
| $\text{above}(a, c)$ | Benar/Salah |
| $\text{above}(b, c)$ | Benar/Salah |

**UTS IF2121 Logika Komputasional 2021/2022**

Relational Logic : Translation, Syntax, Semantic

1. Di bawah ini manakah translasi yang benar dari kaliman “everyone is loved by someone” menjadi kalimat logika relasional? Misalkan ada relasi love(x,y) yang artinya x loves y

- 1)  $\forall x. \exists y. \text{love}(x,y)$
- 2)  $\exists y. \forall x. \text{love}(x,y)$
- 3)  $\exists y. \forall x. \text{love}(y,x)$
- 4)  $\forall x. \exists y. \text{love}(y,x)$
- Jawaban 1 dan 2 benar
- Jawaban 3 dan 4 benar
- Jawaban 1 dan 3 benar
- Jawaban 1 dan 4 benar
- Jawaban 2 dan 3 benar
- Jawaban 2 dan 4 benar
- Jawaban 1,2, dan 3 benar
- Jawaban 2, 3, dan 4 benar
- Semua jawaban benar
- Semua jawaban salah

2. Dalam logika relasional, manakah di bawah ini yang bukan merupakan terms ?

- variable
- object constant
- function
- relation
- variable dan object constant
- function dan relation
- variable dan function
- object constant dan function
- variable dan relation
- object constant dan relation
- variable, object constant, function
- variable, object constant, function, dan relation
- semua jawaban salah
- semua jawaban benar

3. Jika diketahui bahwa manusia adalah sebuah relasi, Dalam logika relasional, manakah di bawah ini yang merupakan ground sentence :

- manusia(y)
- manusia(b)
- $\forall x. \text{mahasiswa}(x)$
- semua jawaban salah
- semua jawaban benar

4. kalimat logika relasional  $\exists x. \forall y. p(x,y)$  dan  $\forall y. \exists x. p(x,y)$  secara semantik adalah :

- sama secara semantik
- beda secara semantik
- Tidak bisa ditentukan
- Semua jawaban salah

## Propostional Proof : Axiom Schemata, Rule of Inference, Clausal Form, Resolution Principle

1. Jika terdapat kumpulan premis sebagai berikut: (i)  $(p \vee q) \rightarrow r$ , (ii)  $r \rightarrow s$ , dan (iii)  $\sim s$ ; tentukan apakah kalimat berikut salah atau benar.

|                                                                                                                                                                 | Benar                 | Salah                 |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|-----------------------|
| a. Penerapan Implication Introduction pada premis (ii) menghasilkan $(r \rightarrow (t \rightarrow s))$ .                                                       | <input type="radio"/> | <input type="radio"/> |
| b. Kaidah Modus Tolens dapat diterapkan pada premis (ii) dan premis (iii) dan menghasilkan $r$ .                                                                | <input type="radio"/> | <input type="radio"/> |
| c. Dengan menerapkan Axiom Schemata saja tanpa menggunakan Kaidah/ Aturan Inferensi apapun, kesimpulan $\sim q$ dapat dihasilkan dari kumpulan premis tersebut. | <input type="radio"/> | <input type="radio"/> |
| d. Dengan menerapkan Kaidah/ Aturan Inferensi saja tanpa menggunakan Axiom Schemata apapun, kesimpulan $\sim q$ dapat dihasilkan dari kumpulan premis tersebut  | <input type="radio"/> | <input type="radio"/> |
| e. Penerapan Hypothetical Syllogism pada premis (i) dan (ii) dapat menghasilkan $(p \vee q)$ .                                                                  | <input type="radio"/> | <input type="radio"/> |
| f. Contradiction Realization pada premis (i) menghasilkan $((p \vee q) \rightarrow r) \rightarrow ((\sim(p \vee q) \rightarrow \sim r) \rightarrow r)$ .        | <input type="radio"/> | <input type="radio"/> |

2. Ingin dibuktikan bahwa  $\{p \rightarrow \neg q, \neg q \rightarrow (r \wedge s)\} \models ((p \rightarrow \neg(r \wedge s)) \rightarrow \neg p)$ .

Tentukan kalimat yang mungkin dihasilkan dari penerapan Axiom

Schemata satu langkah pada premis.

- a.  $(p \rightarrow \neg q) \rightarrow (\neg q \rightarrow (p \rightarrow \neg q))$
- b.  $(\neg q \rightarrow (r \wedge s)) \rightarrow ((\neg q \rightarrow r) \wedge (\neg q \rightarrow s))$
- c.  $(s \rightarrow (p \rightarrow \neg q))$
- d.  $((p \rightarrow q) \rightarrow \neg p)$
- e.  $(p \rightarrow \neg q) \rightarrow ((r \wedge s) \rightarrow (p \rightarrow q))$
- f.  $(p \rightarrow (\neg q \rightarrow (r \wedge s)))$
- g. Salah semua
- h. Benar semua

3. Ingin dibuktikan bahwa  $\{p \rightarrow q, q \rightarrow (r \vee s), r \rightarrow \neg p, s \rightarrow \neg p\} \models (\neg p)$ .

Tentukan kalimat yang mungkin dihasilkan dari penerapan Kaidah Inferensi  
satu langkah pada kumpulan premis.

- a.  $\neg(r \vee s) \rightarrow q$
- b.  $\neg p \vee r$
- c.  $(p \rightarrow s) \wedge (p \rightarrow r)$
- d.  $p \rightarrow (r \vee s)$
- e.  $\neg q \vee (\neg r \wedge \neg s)$
- f.  $(r \vee s)$
- g. Salah semua
- h. Benar semua

## 4. Tentukan pernyataan berikut benar atau salah.

|                                                                                                                                                                                                                                                     | Benar                 | Salah                 |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|-----------------------|
| a. Operator logika disjungsi, konjungsi, dan biimplikasi bersifat komutatif dalam Kaidah Inferensi.                                                                                                                                                 | <input type="radio"/> | <input type="radio"/> |
| b. Operator logika disjungsi, konjungsi, dan implikasi bersifat asosiatif dalam Kaidah Inferensi.                                                                                                                                                   | <input type="radio"/> | <input type="radio"/> |
| c. Kaidah De Morgan menyatakan: $\sim\sim p \leftrightarrow p$ (catatan: $\leftrightarrow$ adalah biimplikasi).                                                                                                                                     | <input type="radio"/> | <input type="radio"/> |
| d. Jika terdapat kumpulan premis $\{p, q \rightarrow r\}$ dapat dihasilkan $p \vee q$ dengan menerapkan satu langkah aturan pada kaidah Inferensi.                                                                                                  | <input type="radio"/> | <input type="radio"/> |
| e. Untuk mendapatkan kalimat $p \rightarrow (\sim q \rightarrow (r \wedge s))$ dari kumpulan premis $\{p \rightarrow \sim q, \sim q \rightarrow (r \wedge s)\}$ minimal diperlukan dua langkah penerapan axiom schemata dan/ atau kaidah inferensi. | <input type="radio"/> | <input type="radio"/> |
| f. Klausa dalam 'proposal resolution' selalu berupa disjungsi dari literal.                                                                                                                                                                         | <input type="radio"/> | <input type="radio"/> |
| g. Pembuktian dengan 'proposal resolution' mirip dengan cara pembuktian semantic reasoning dengan unsatisfiability checking dalam aspek bagaimana premis dan kesimpulan digabungkan.                                                                | <input type="radio"/> | <input type="radio"/> |

h. Pembuktian dengan 'proposal resolution' harus mengubah premis dan kesimpulan dari representasi logika proposisi ke bentuk klausa.



5. Jika terdapat kalimat premis  $q \rightarrow \sim(\sim r \leftrightarrow p)$  dan akan diubah ke bentuk klausa, maka hasil dari tahap I (tahap pertama) adalah:

- a.  $\sim q \vee (\sim((\sim r \wedge \sim p) \vee (p \wedge r)))$
- b.  $\sim q \vee (\sim((r \vee p) \wedge (p \vee \sim r)))$
- c.  $\sim q \vee ((\sim r \wedge \sim p) \vee (p \wedge r))$
- d.  $\sim q \vee (\sim((\sim r \vee p) \wedge (\sim p \vee \sim r)))$
- e.  $\sim q \vee (\sim((\sim \sim r \vee p) \wedge (\sim p \vee \sim r)))$
- f. Salah semua

6. Jika ingin dibuktikan bahwa  $\{p \rightarrow q, p \vee q, q \rightarrow \sim(\sim r \leftrightarrow p), r\} \models (p)$  dengan 'proposal resolution', maka klausa yang mungkin dihasilkan sebelum dilakukan prinsip resolusi adalah:

|  | Benar | Salah |
|--|-------|-------|
|--|-------|-------|

- |                            |                       |                       |
|----------------------------|-----------------------|-----------------------|
| a. $\{p\}$                 | <input type="radio"/> | <input type="radio"/> |
| b. $\{r\}$                 | <input type="radio"/> | <input type="radio"/> |
| c. $\{\sim q, r, p\}$      | <input type="radio"/> | <input type="radio"/> |
| d. $\{\sim q, \sim r, r\}$ | <input type="radio"/> | <input type="radio"/> |
| e. $\{\sim q, \sim p, r\}$ | <input type="radio"/> | <input type="radio"/> |

7. Tentukan apakah pernyataan berikut ini salah atau benar.

Benar

Salah

a. Sebuah kalimat logika

proposisi  $(p \rightarrow (q \rightarrow r)) \rightarrow ((p \wedge q) \rightarrow r)$  dapat dinyatakan valid (tautology) dengan memanfaatkan 'propositional resolution'; dengan mengubah kalimat tersebut ke bentuk klausa dan menerapkan prinsip resolusi pada klausa-klausa tersebut untuk mendapatkan klausa kosong.

b. Kumpulan klausa  $\{p, \neg q\}$ ,  $\{q, r\}$ ,  $\{\neg p, q, r\}$ ,  $\{\neg r\}$  tidak dapat menghasilkan klausa kosong.

c. Penerapan prinsip resolusi terhadap klausa  $\{\neg q, r, p\}$  dan  $\{\neg q, \neg r, r\}$  adalah  $\{\neg q, p\}$ .

d. Klausa  $\{r, s\}$  dapat dihasilkan dengan penerapan prinsip resolusi minimal 2 langkah pada kumpulan klausa  $\{\neg p, q\}$ ,  $\{\neg s, \neg p\}$ ,  $\{\neg q, r, s\}$ ,  $\{p\}$ ,  $\{\neg r, \neg p\}$ .

e. Klausa yang mungkin

dihadirkan dengan menerapkan satu langkah prinsip resolusi pada kumpulan klausa  $\{p, \neg q, r\}$ ,  $\{\neg p, q, \neg r\}$  adalah:  $\{\neg q, \neg r, q, r\}$  atau  $\{\neg p, \neg r, p, r\}$  atau  $\{p, \neg p, q, \neg q\}$ .

## Relational Proof

1.

Terdapat sebuah ekspresi  $\forall x. \forall z. (\exists w. (s(z,w)) \vee p(x,y) \vee q(y,z))$ . Jika diterapkan Existential Instantion pada ekspresi tersebut, maka berikut ini ekspresi yang bisa dihasilkan. \*

|                                                             | Benar                 | Salah                 |
|-------------------------------------------------------------|-----------------------|-----------------------|
| $\forall x. \forall z. (s(z,w) \vee p(x,y) \vee q(y,z))$    | <input type="radio"/> | <input type="radio"/> |
| $\forall x. \forall z. (s(z,a) \vee p(x,y) \vee q(y,z))$    | <input type="radio"/> | <input type="radio"/> |
| $\forall x. \forall z. (s(z,f(a)) \vee p(x,y) \vee q(y,z))$ | <input type="radio"/> | <input type="radio"/> |
| $\forall x. \forall z. (s(f(z,a)) \vee p(x,y) \vee q(y,z))$ | <input type="radio"/> | <input type="radio"/> |
| $\forall z. (s(z,w) \vee p(a,y) \vee q(y,z))$               | <input type="radio"/> | <input type="radio"/> |
| $\forall z. (s(z,w) \vee p(x,y) \vee q(y,z))$               | <input type="radio"/> | <input type="radio"/> |

2.

Terdapat 3 buah premis seperti tertulis di bawah ini. Jika diinginkan untuk mendapatkan ekspresi  $p(b,c) \wedge q(c) \Rightarrow r(b)$  dalam lima langkah (atau lima baris, penghitungan baris tidak melibatkan baris premis pertama s.d. premis ketiga), maka urutan penerapan 'Inference Rules' atau 'Axiom Schemata' adalah sebagai berikut. \*

3. Jika terdapat premis  $\exists x. (p(x) \wedge q(a,x))$  dan  $\forall y. (\exists x. p(x) \wedge q(y,x)) \Rightarrow r(y)$ , maka: \*

|                                                                                                                                                                                                                                                                   | Benar                 | Salah                 |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|-----------------------|
| Penerapan kaidah "Existensial Instantiation" pada premis yang pertama menghasilkan kalimat $p(x) \wedge q(a,x)$ .                                                                                                                                                 | <input type="radio"/> | <input type="radio"/> |
| Penerapan kaidah "Universal Instantiation" pada premis yang kedua menghasilkan kalimat $(p(x) \wedge q(y,x)) \Rightarrow r(y)$                                                                                                                                    | <input type="radio"/> | <input type="radio"/> |
| Setelah diterapkan kaidah "Universal Instantiation" pada premis yang kedua, dan berikutnya diterapkan kaidah "Existensial Instantiation" maka kalimat yang dihasilkan adalah $p(b) \wedge q(a,b)) \Rightarrow r(a)$                                               | <input type="radio"/> | <input type="radio"/> |
| Penerapan kaidah "Existensial Instantiation" pada premis yang pertama, kemudian kaidah "And Elimination" menghasilkan kalimat $q(a,b)$ .                                                                                                                          | <input type="radio"/> | <input type="radio"/> |
| Jika pada premis 1 diterapkan kaidah "Existensial Instantiation", dan pada premis yang kedua diterapkan kaidah "Universal Instantiation" kemudian kaidah "Existensial Instantiation"; maka hasil untuk kedua premis tadi dapat diterapkan kaidah "Modus Tollens". | <input type="radio"/> | <input type="radio"/> |

4.

Diketahui dua fakta sebagai berikut: [1]  $\forall x. (\text{pred1}(x) \Rightarrow \text{pred2}(x)) \Rightarrow \exists x. (\text{pred3}(x) \wedge \text{pred4}(x)$  dan [2]  $\forall x. (\text{pred1}(x) \Rightarrow \text{pred4}(x)) \wedge \forall x. (\text{pred4}(x) \Rightarrow \text{pred2}(x))$ . Buktikan bahwa kesimpulan:  $\exists x. \text{pred4}(x)$  dapat ditarik dari kumpulan fakta tersebut dengan memanfaatkan kaidah inferensi dan/atau axiom schemata. Tuliskan aturan inferensi dan/atau axiom schemata yang digunakan dan nomor baris penerapannya. \*

- [3] Universal Instantiation 2
- [4] And Elimination 3
- [5] And Elimination 3
- [6] Hypothetical Syllogism 4,5
- [7] Universal Instantiation 1
- [8] Modus Ponens 7,6
- [9] Existential Instantiation 8
- [10] And Elimination 9
- [11] Existential Generalization

## UAS LOGIKA KOMPUTASIONAL

Program Studi Teknik Informatika  
 Sekolah Teknik Elektro dan Informatika  
 Institut Teknologi Bandung

IF2252 Logika Informatika  
 Kelas K2, Waktu: 120 menit

### UAS SEMESTER II

#### Soal 1 (30):

Diberikan pernyataan berikut ini :

Semua mahasiswa itu pintar. Fulan adalah mahasiswa. Semua orang yang pintar memiliki kelemahan. Jadi, Fulan memiliki kelemahan.

- a. Representasikanlah pernyataan tersebut ke dalam logika predikat. Gunakanlah relasi:
  - mahasiswa(x) = x adalah mahasiswa
  - pintar(x) = x adalah orang yang pintar
  - lemah(x) = x memiliki kelemahan
- b. Buktikanlah dengan kaidah inferensi.
- c. Buktikan dengan standard axiom schemata dan modus ponen.

#### Soal 2 (30):

Diberikan kalimat dalam representasi logika relasional.

$$\exists x. (\text{makanan}(x) \wedge \forall y. (\text{mahasiswa}(y) \rightarrow \text{suka}(y,x)))$$

$$\text{Jadi } \forall y. (\text{mahasiswa}(y) \rightarrow \exists x. (\text{makanan}(x) \wedge \text{suka}(y,x)))$$

$\text{makanan}(x) = x$  adalah makanan

$\text{mahasiswa}(x) = x$  adalah mahasiswa

$\text{suka}(x,y) = x$  suka  $y$

- a. Terjemahkanlah kalimat tersebut ke dalam bahasa alami.
- b. Ubahlah kalimat tersebut ke bentuk clausal.
- c. Buktikan pernyataan tersebut dengan resolusi relasional. Lengkapilah mgu pada setiap inferensi yang digunakan.

#### Soal 3 (20):

Jelaskan perbedaan dari ketiga program untuk menentukan nilai maksimum berikut ini:

`max(X,Y,X) :- X >= Y.`

`max(X,Y,Y) :- X < Y.`

`max(X,Y,X) :- X >= Y, !.`

`max(X,Y,Y) :- X < Y.`

`max(X,Y,X) :- X >= Y, !.`

`max(X,Y,Y) .`

#### Soal 4 (20):

Persoalan water-jug problem:

Diberikan 2 water-jug, Jug-A (kapasitas 4 liter) dan Jug-B (kapasitas 3 liter). Tidak ada penanda liter pada kedua jug. Bagaimana cara mengisi air sebanyak 2 liter di Jug-A?

Predikat move dalam bahasa prolog di bawah ini yang menyatakan perpindahan state yang terjadi. Tipe komposisi: jugs(jugA,jugB) yang menyatakan kapasitas jugA dan jugB.

```
move(jugs(J1,J2), jugs(4,J2)) :- J1 < 4.
move(jugs(J1,J2), jugs(J1,3)) :- J2 < 3.
move(jugs(J1,J2), jugs(0,J2)) :- J1 > 0.
move(jugs(J1,J2), jugs(J1,0)) :- J2 > 0.
move(jugs(J1,J2), jugs(4,N)) :- J2 > 0, J1+J2 >= 4, N is J2 - (4 - J1).
move(jugs(J1,J2), jugs(N,3)) :- J1 > 0, J1+J2 >= 3, N is J1 - (3 - J2).
move(jugs(J1,J2), jugs(N,0)) :- J2 > 0, J1+J2 < 4, N is J1+J2.
move(jugs(J1,J2), jugs(0,N)) :- J1 > 0, J1+J2 < 3, N is J1+J2.
```

Periksalah apakah program prolog ini bisa menyelesaikan persoalan water-jug tersebut (state awal : jugs(0,0); state akhir : jugs(2,0)). Jika ya, jelaskan solusinya.

## UAS SEMESTER II – 2005/2006

IF2252 – Logika Informatika

90 menit

Sifat : Buku Tertutup

1. Jika diberikan premis:

1.  $\forall x. (\text{rajin}(x) \rightarrow \text{lulus}(x))$
  2.  $\forall x. (\text{mahasiswa}(x) \wedge \text{rajin}(x))$
- Buktikan bahwa mahasiswa(Ina) → lulus(Ina). (Nilai : 25)

2. Buatlah sebuah rule getFamily yang menerima masukan list nama-nama orang, kemudian menghasilkan list yang berisi nama-nama orang yang merupakan *cousin* atau *sibling* dari seseorang. Jika diperlukan rule antara, tuliskan rule tersebut. (NILAI : 25)

%% Diberikan fakta sebagai berikut :

%% Fakta

|                 |                     |
|-----------------|---------------------|
| %% sibling(X,Y) | X saudara kandung Y |
| %% cousin(X,Y)  | X sepupu Y          |

cousin(a,b).  
 cousin(a,c).  
 cousin(a,d).  
 sibling(a,e).  
 sibling(a,f).  
 sibling(a,g).

%% Buatlah rule berikut :

%% Rule  
 %% getFamily(L1,X,L2)  
 %% L2 adalah list nama-nama orang. L1 adalah nama orang  
 %% yang terdapat pada L2 yang merupakan saudara dari X

Contoh eksekusi :

```

| ?- getFamily(L,a, [d,z]).

L = [d] ?

yes

| ?- getFamily(L,a, [d,z,b,k,l,m,e])

L = [d,b,e] ?

yes

```

3. Buatlah sebuah rule gabunglist yang akan menggabungkan list L1 dengan list L2 menjadi list L3. Jika diperlukan rule antara, tuliskan rule tersebut. (NILAI : 25)

Contoh eksekusi:

```

| ?- gabunglist(L, [a,b,c], [d,e])

L = [a,b,c,d,e]

```

yes

```
| ?- gabunglist(L, [a,k,m], [d,z,b,k,l,m,e])
L = [a,k,m,d,z,b,k,l,m,e]
yes
```

4. Buatlah sebuah rule saudarakandung yang akan memberikan daftar orang-orang yang bersaudara kandung seibu sebapak (NILAI : 25)

%% Diberikan fakta sebagai berikut :

%% Fakta

|             |                              |
|-------------|------------------------------|
| %%ayah(X,Y) | X adalah ayah kandung dari Y |
| %%ibu(X,Y)  | X adalah ibu kandung dari Y  |

```
ayah(phiips,charles).
ayah(phiips,andy).
ayah(phiips,anne).
ibu(elizabeth,charles).
ibu(elizabeth,andy).
ibu(elizabeth,anne).
ayah(charles,william).
ibu(diana,william).
```

%% Buatlah rule berikut :

%% Rule  
 %% saudarakandung(X,Y)  
 %% yang artinya X adalah saudara kandung dari Y

Contoh eksekusi :

```
| ?- saudarakandung(X,Y).
X = charles, Y = andrew;
X = charles, Y = anne;
X = andrew, Y = charles;
X = andrew, Y = anne;
X = anne, Y = charles;
X = anne, Y = andrew;
yes
```

Perhatian!!! Jika jawaban dari program anda adalah sebagai berikut :

```
| ?- saudarakandung(X,Y).
X = charles, Y = charles;
X = charles, Y = andrew;
X = charles, Y = anne;
X = andrew, Y = charles;
X = andrew, Y = andrew;
X = andrew, Y = anne;
```

X = anne, Y = charles;

X = anne, Y = andrew;

X = anne, Y = anne;

yes

maka anda akan mendapat nilai 0(nol) untuk nomor ini.

**UAS SEMESTER II**

Mata Kuliah : IF2050 Logika Informatika

Kelas : K1

Dosen : Dr. Ir. Rila Mandala, M.Eng

Waktu : 90 menit

Sifat Ujian : Closed-Book

1.  $\text{Ax . Ay . (h(x) \& d(y) \Rightarrow f(x,y))}$   
 $\text{Ey . (g(y) \& Az . (r(z) \Rightarrow f(y,z)))}$   
 $\text{Ay . (g(y) \Rightarrow d(y))}$   
 $\text{Ax . Ay . Az . (f(x,y) \& f(y,z) \Rightarrow f(x,z))}$

*h(harry)*

*r(ralph)*

- a. Ubah ke dalam clausal form (NILAI : 10)
- b. Apakah dari premis-premis di atas kita bisa menyimpulkan  $f(harry, ralph)$ ? Buktikan dengan prinsip resolution (NILAI : 10)

2. Tuliskan apa output program di bawah ini

(NILAI : 15) DOMAINS

```

list = integer*
PREDICATES
 itb(list,list)
CLAUSES
 itb([],[]).
 itb([Head | [Head2 | Tail]], [Head2 | Tail1]):-
 itb(Tail,Tail)
GOAL 1).

itb([1,2,3,4,
5,6,7,8],Inf
 ormatika).
```

3. Tuliskan apa output program di bawah ini (NILAI : 15) :

```

DOMAINS
 integerlist = integer*
PREDICATES
 lab_irk(integerlist,integerlist,integerlist)
CLAUSES
 lab_irk([],List,List).
 lab_irk([H | L1] , List2 , [H | L3]):-
 1
 GOAL a
 b
```

|   |   |
|---|---|
| - | - |
| i | i |
| r | r |
| k | k |
| ( | ( |
| L | L |
| 1 | 1 |
| , | , |
| L | L |
| i | 2 |
| s | , |
| t | [ |
| 2 | 1 |
| , | , |
| L | 2 |
| 3 | , |
| ) | 4 |
| . | ] |
| 1 | ) |
| a | . |
| b |   |

4. Tuliskan apa output program di bawah ini (NILAI : 15) ? Analisislah apakah mengerjakan apa program tersebut di bawah ini (NILAI : 10)

DOMAINS  
list = integer\*

## PREDICATES

secrets(list,list)

## CLAUSES

```
secrets([],[]).
secrets([H | T], Lab_IRK):-
 H < 0,
 !,
 secrets(T, Lab_IRK).
```

GOAL

```
s
e
c
r
e
t
s
(
[
H
|
T
]
,
[
H
|
L
a
b

-
I
R
K
]
)
:
-
s
e
c
r
e
t
s
(
T
,
L
a
b

-
I
R
K
)
.
```

secrets([2,-45,3,-46], X).

5. Tuliskanlah apa output program di bawah ini (NILAI : 15) ? Analisilah mengerjakan apa program tersebut di bawah ini (NILAI : 10).

```
DOMAINS
 list = integer*
PREDICATES
 lab(list,list)
CLAUSES
 lab([],[]).
 lab([H | T], [H, H | IRK]):-
 lab(T, IRK).
GOAL
 lab(1,2,3,4], X).
```

## UAS SEMESTER II – 2010/2011

Waktu: 120 menit

1. Tentukan apakah masing-masing pasangan pedikat di bawah ini unifiable atau tidak. Jika ya, temukanlah Most general Unifier (MGU)-nya. Keterangan: v,w,x,y,z adalah variabel, sedangkan a dan b adalah konstanta.

- $p(x, b, f(y))$  dan  $p(a, y, f(z))$
- $p(f(g(a)), x, f(h(z, z)), h(y, g(w)))$  dan  $p(y, g(z), f(v), h(f(w), x))$
- $p(t(x, y), r(z, z))$  dan  $p(t(t(w, z), v), w)$

2. Jika diketahui premis di bawah ini:

$$\exists x. \forall y. (p(x, y) \leftrightarrow q(x, y))$$

$$\forall x. (\exists y. p(x, y) \vee \exists z. q(x, z))$$

Buktikan dengan menggunakan teori resolusi kesimpulan di bawah ini:

$$\exists x. \exists y. (p(x, y) \wedge q(x, y))$$

3. Terdapat program Prolog sebagai berikut

```
append([], X, X) :- !.
append([A|B], C, [A|D]) :- append(B, C, D).
```

Tentukan hasil dari query berikut, yang diterapkan pada program di atas.

- $\text{append}([a, b, c], [d, e], X)$ .
- $\text{append}(X, Y, [a, b, c])$ .
- $\text{append}([a, b, c], X, Y)$ .
- $\text{append}(X, [], Y)$ .

4. Terdapat dua program prolog sebagai berikut:

```
i. number_of_parents(adam, 0) :- !.
 number_of_parents(hawa, 0) :- !.
 number_of_parents(X, 2).
ii. number_of_parents(adam, N) :- !, N=0.
 number_of_parents(hawa, N) :- !, N=0.
 number_of_parents(X, 2).
```

- Jelaskan dengan singkat perbedaan dari kedua program tersebut.
- Untuk setiap program i dan ii, berikan hasil penerapan query:  $\text{number\_of\_parents(john,X)}$ .
- Untuk setiap program i dan ii, berikan hasil penerapan query:  $\text{number\_of\_parents(hawa,2)}$ .

**UAS SEMESTER II – 2011/2012**

Mata Kuliah : IF2050 Logika Informatika

Kelas : K1 dan K2

Waktu : 120 menit

Sifat Ujian : Closed-book

1. Dari pasangan ekspresi di bawah ini, tentukan MGU (Most General Unifier) dari tiap pasangan tersebut jika ada. (NILAI: 15)
  - a.  $p(f(g(a)), x, f(h(z, z)), h(y, g(w)))$  dan  $p(y, g(z), f(y), h(f(w), x))$
  - b.  $p(f(x), r(x), c)$  dan  $q(y, r(z), z)$
  - c.  $p(x, a, b, x)$  dan  $p(c, y, z, y)$
  - d.  $p(t(x, y), r(z, z))$  dan  $p(t(t(w, z), v), w)$
  - e.  $p(x, b, f(y))$  dan  $p(a, y, f(z))$

2. Terdapat basis pengetahuan sebagai berikut:

Tony, Mike, dan John adalah anggota Alpine Club. Setiap anggota klub yang bukan pemain ski adalah pemanjat gunung. Pemanjat gunung tidak suka hujan. Setiap orang yang tidak suka salju bukanlah pemain ski. Mike menyukai apa yang tidak disukai Tony. Tony menyukai hujan dan salju.

Buktikan apakah kesimpulan “ada anggota klub yang seseorang pemanjat gunung, namun bukan pemain ski” dapat diturunkan dari basis pengetahuan di atas, dengan memanfaatkan *relational resolution*. Ubahlah basis pengetahuan di atas ke dalam bentuk FOL terlebih dulu.

Gunakan relasi: member(x); skier(x); mountain(x); dan likes(x,y) (NILAI 15)

3. Terdapat program yang dituliskan dalam Prolog sebagai berikut, untuk mengklasifikasikan bilangan bulat ke dalam kelas zero, negative, atau positive. (NILAI 20)

```
classify(0, zero) :- !.
classify(N, negative) :- !, N<0.
classify(N, positive).
```

- a. Tentukan jawaban beberapa query berikut, jika diterapkan pada program di atas.
  - i.  $?- classify(0, N).$
  - ii.  $?- classify(-7, N).$
  - iii.  $?- classify(8, N).$
  - iv.  $?- classify(-5, positive).$
- b. Berdasarkan jawaban dari query pada butir (a), tentukan apakah program Prolog di atas sudah tepat. Jika belum tepat, berikan alasan dan tuliskan program yang lebih tepat.

4. Diketahui premis di bawah ini

1.  $\forall x. (orang(x) \Rightarrow berakal(x))$
2.  $\forall x. (\neg manusia(x) \Rightarrow \neg berakal(x))$
3.  $\exists x. (orang(teman(x)))$

Buktikan dengan menggunakan *relational-proof* bahwa kesimpulan ini tercapai

$\exists x. (orang(x) \wedge berakal(x) \wedge manusia(x))$  (NILAI 20)

5. Buatlah sebuah program prolog untuk mendefinisikan predikat yang akan menghilangkan semua bilangan negative dari sebuah list. Sebagai contoh buangnegatif([1,2,-3,4],L), maka hasilnya L akan berisi nilai [1,2,4]. (NILAI 10)
6. Diketahui program prolog di bawah ini:

```
domains
list = integer*

predicates
misterius(list,integer,integer)

clauses
misterius([], Horor, Horor).
misterius([_|T], Horor, Rahasia).
 misteri = Rahasia + 1,
 misterius(T, Horor, Misteri).
```

- a. Jika goal-nya adalah misterius([1,2,3,4,5],L,5), maka apa output dari prolog? (NILAI 10)
- b. Untuk apakah kegunaan dari predikat misterius tersebut (NILAI 10)

Program Studi Teknik Informatika  
Institut Teknologi Bandung

UAS SEMESTER II – 2012/2013

Mata kuliah : IF2050 Logika Informatika  
Kelas : K1 dan K2  
Waktu : 150 menit  
Ujian : Closed book

1. Buatlah program prolog sederhana untuk operasi list sederhana bilangan integer (tidak mengandung list di dalam list), yang terdiri atas: (NILAI 15)
  - a. Predikat `size(list, integer)` untuk menhitung banyaknya elemen list; contoh query : `size([1,2,3,4], N)` akan menghasilkan  $N = 4$ .
  - b. Predikat `sumlist(list, integer)` untuk menghasilkan elemen-elemen dalam list; contoh query : `sumlist([1,2,3,4], N)` menghasilkan  $N = 10$ .
  - c. Predikat `ismember(integer, list)` untuk memeriksa apakah elemen integer ada di dalam list; contoh query : `ismember(0, [1,2,3,4])` menghasilkan no.

2. Terdapat program prolog sederhana berikut ini. (NILAI 15)

```
parent(charles, william).
parent(charles, harry).
parent(phillips, charles).
parent(elizabeth, charles).
parent(victoria, elizabeth).
ancestor(X, Y) :- parent(X, Y).
ancestor(X, Y) :- ancestor(X, Z), parent(Z, Y).
```

- a. Tentukan hasil dari query `ancestor(X, harry)`.
  - b. Tentukan hasil dari query `ancestor(victoria, Y)`.
  - c. Tentukan hasil dari query `ancestor(victoria, william)`.
3. Tentukan apakah pernyataan berikut ini benar atau salah, dan jika salah tuliskan pernyataan yang benar. (NILAI 10)
    - a. Prolog adalah salah satu contoh bahasa pemrograman dengan paradigma fungsional.
    - b. Red cut pada prolog berfungsi untuk meningkatkan efisiensi eksekusi program, sedangkan penambahan green cut dapat mengubah arti program.
    - c. Analisa kasus untuk merealisasikan suatu aturan (*rule*) yang bersifat ‘OR’ pada prolog, dituliskan dengan membuat banyak kondisi pada satu aturan (*rule*) tersebut.
    - d. Aspek rekurens pada prolog mungkin diterapkan pada fakta maupun aturan.

4. Terdapat program prolog sederhana berikut ini. (NILAI 15)

```
male(arief).
male(pandu).
female(dyah).
female(arini).
parent(pandu, arief, dyah).
parent(arini, arief, dyah).
sister_of(X, Y) :- female(X), parents(X, M, E),
parents(Y, M, E).
```

- a. Tentukan hasil dari query `sister_of(arini, Y)` yang diterapkan pada program tersebut.
- b. Berdasarkan hasil query program tersebut, jelaskan dengan singkat apakah program tersebut seharusnya sudah benar atau belum. Jika belum benar, tuliskan bagaimana program yang seharusnya.

5. Ubahlah ke dalam bentuk klausal (menggunakan prinsip INSEADO). (NILAI 10)
  - a.  $\exists x. \text{kuda}(x) \Rightarrow \neg \forall y. (\text{binatang}(y) \Rightarrow \neg \text{eats}(y, \text{rumput}))$
  - b.  $\neg \exists x. (g(x) \wedge \forall z. (r(z) \Rightarrow f(x, z)))$
6. Carilah *most-general unifier* dari kalimat-kalimat berikut (jika tidak ada, maka tuliskan tidak ada). (NILAI 20)
  - a.  $s(t(x), u(x), z)$  dan  $s(y, z, u(x))$
  - b.  $q(f(y), y)$  dan  $q(x, f(x))$
  - c.  $p(t(x,y), r(z,z))$  dan  $p(t(t(w, z), v), w)$
  - d.  $f(h(r, s, t), h(w, x, y), t, w)$  dan  $f(h(g(u, v), s, w), h(x, y, w), t, t)$
7. Jika diketahui premis :  
 $\exists x. \exists y. (p(x, y) \Leftrightarrow q(x, y))$   
 $\forall x. \exists y. (p(x, y) \vee q(x, y))$   
Buktikan dengan *relational relation* kesimpulan berikut ini.  
 $\exists x. \exists y. (p(x, y) \wedge q(x, y))$   
(NILAI 20) (mengubah ke *clausal form* nilai 10, pembuktian dengan prinsip resolusi nilai 10)

--- SELAMAT MENGERJAKAN ---

**UAS SEMESTER I – 2014/2015**

Program Studi Teknik Informatika  
Institut Teknologi Bandung

**UJIAN AKHIR SEMESTER GANJIL 2014/2015**

Matakuliah : IF2121 Logika Informatika  
Kelas : K1 dan K2  
Waktu : 120 menit  
Sifat Ujian : Closed-book.

**Bagian I. Relational Logic**

1. Terdapat suatu ‘cerita’ sebagai berikut.

“Anyone passing his or her logic exam and winning the programming competition is happy. But anyone who studies or is lucky can pass all his/ her exams. Pete did not study but is lucky. Anyone who is lucky wins the programming competition. Is Pete happy?”

Buktikan apakah “Pete is happy” dapat diturunkan dari kalimat premis sebelumnya, menggunakan *relational resolution*.

Gunakan relasi: pass(x,y), win(x,y), happy(x), study(x), lucky(x);  
dan gunakan object constants berikut: logic\_exam, prog\_comp, Pete. (**NILAI : 20**)

happy (per)  
pass (是)

2. Dengan menggunakan *relational resolution*, buktikan bahwa kalimat berikut ini valid.  
**(NILAI : 10)**

$$\forall x ((\underline{(p(x) \Rightarrow q(x))} \Rightarrow p(x)) \Rightarrow p(x))$$

3. Untuk soal di bawah ini, carilah *most-general-unifier* (MGU) nya apabila ada. Jika tidak ada MGU-nya maka cukup ditulis : Tidak ada MGU-nya.

- a.  $p(x,x)$  dan  $p(a,y)$  (NILAI : 5)
- b.  $p(x,x)$  dan  $p(f(y),z)$  (NILAI : 5)
- c.  $p(x,x)$  dan  $p(f(y),y)$  (NILAI : 5)
- d.  $p(f(x,y),g(z,z))$  dan  $p(f(f(w,z),v),w)$  (NILAI : 5)

~~5,6~~  
5,6 win  
3,5 pass

**Bagian II. Prolog**

4. Terdapat program prolog sederhana sebagai berikut. Program dibuat untuk menyatakan bahwa Pete menyukai semua jenis burger kecuali Big Kahuna Burger.

```

enjoys(pete,X) :- big_kahuna_burger(X), fail.
enjoys(pete,X) :- burger(X).

burger(X) :- big_mac(X).
burger(X) :- big_kahuna_burger(X).
burger(X) :- whopper(X).

big_mac(a).
big_kahuna_burger(b).
big_mac(c).
whopper(d).
```

Berdasarkan program tersebut, jawablah pertanyaan berikut ini.

a. Jawablah *query* di bawah ini jika diterapkan pada program tersebut. (NILAI 5)

- (i) enjoys(pete, a).
- (ii) enjoys(pete,b).
- (iii) enjoys(pete,c).
- (iv) enjoys(pete,d).

b. Tentukan apakah program tersebut sudah benar atau belum. Jika sudah benar, cukup tuliskan sudah benar. Jika belum benar, tuliskan bagaimana seharusnya (NILAI 5)

5. Dari pernyataan di bawah ini, tentukan apakah pernyataan tersebut benar atau salah. Jika salah, tuliskan bagaimana seharusnya pernyataan yang benar. (NILAI 20)

- a. Explanation Subsystem adalah komponen dasar dari suatu Expert System yang wajib ada.
- b. Komputasi program pada pemrograman deklaratif murni adalah pencocokan dan kalkulasi, serta adanya parameter relasi yang diinterpretasikan sebagai parameter output.
- c. Struktur data prolog terdiri atas struktur data primitif berupa list dan struktur data kompleks berupa struktur data simbolik.
- d. List [a,b,c,d] dengan list [H|T], unifikasi dapat dilakukan dengan substitusi H dengan [a], dan T dengan [b,c,d].
- e. Kombinasi predikat Cut dan Fail pada Prolog, digunakan untuk merealisasikan IF Statement, artinya untuk memaksa program Prolog menjalankan kemungkinan lain dari predikat yang menerapkan Cut dan Fail tersebut.

6. Buatlah dua program sederhana yang dapat dijalankan dalam GNU Prolog, untuk kedua persoalan sebagai berikut,

a. Program untuk membalik urutan suatu *list of symbol*. Predikat diberi nama myreversedlist, dengan dua parameter yaitu list yang akan dibalik dan list hasil membalik list yang pertama. Contoh query myreversedlist ([a, b, c, d], X) akan menghasilkan X=[d,c,b,a]. Petunjuk: buatlah predikat antara yang lain untuk melakukan proses rekursif pada list, asumsi tidak ada predikat GNU Prolog yang bisa langsung digunakan. (NILAI 10)

b. Program untuk menghapus elemen bernilai positif ( $> 0$ ) pada suatu *list of integer* sederhana. Predikat diberi nama buangpositif, dan memiliki dua parameter yaitu list awal dan list hasil penghapusan elemen bernilai positif. Contoh query buangpositif ([0, 1, 2, -3, 2, -4], X) akan menghasilkan X=[0,-3,-4]. (NILAI 10)

**UAS SEMESTER I – 2015/2016**  
**Mata kuliah:** IF2121 Logika Informatika  
**Kelas:** K1 dan K2  
**Waktu:** 120 menit  
**Sifat Ujian:** Closed-book

**Bagian I. Relational Logic**

1. Jika suatu mata kuliah mudah maka beberapa mahasiswa senang. Jika suatu mata kuliah mempunyai ujian akhir maka tidak ada mahasiswa yang senang. Pergunakan resolusi untuk menunjukkan bahwa jika suatu mata kuliah mempunyai ujian akhir maka mata kuliah tersebut tidak mudah. **(NILAI 15)**  
 Gunakan relasi: mudah( $xx$ ); senang( $xx$ ); ujian( $xx$ ).
2. Carilah *most-general unifier* dari soal di bawah ini (jika tidak ada maka tuliskan tidak ada dan beri penjelasan singkat). **(NILAI 20)**
  - a. color(tweety, yellow) dan color( $xx$ ,  $yy$ )
  - b. color(tweety, yellow) dan color( $xx$ ,  $xx$ )
  - c. color(hat(postman), blue) dan color(hat( $yy$ ),  $xx$ )
  - d. r(f( $xx$ ), b) dan r( $yy$ , zz)
  - e. r(f( $yy$ ),  $yy$ ) dan r( $xx$ , f(b))
  - f. r(f( $yy$ ),  $yy$ ,  $xx$ ) dan r( $xx$ , f(a), f(vv))
  - g. q(g(ww), h♦ww, f(xx, mm)♦) dan q(g(vv), h(mm, vv))
3. Jika diketahui premis di bawah ini  
 $\exists xx. \forall yy. \Diamond pp(xx, yy) \Leftrightarrow qq(xx, yy) \Diamond$   
 $\forall xx. (\exists yy. pp(xx, yy) \vee \exists zz. qq(xx, zz))$   
 Bukti dengan menggunakan *relational resolution* kesimpulan di bawah ini:  
 $\exists xx. \exists yy. (pp(xx, yy) \wedge qq(xx, yy))$   
**(NILAI 15)**

**Bagian II. Prolog**

4. Buatlah program Prolog sederhana (tanpa menggunakan predikat yang sudah ada pada prolog), untuk operasi *list* sederhana bilangan integer (tidak mengandung *list* di dalam *list*), sebagai berikut. **(NILAI 20)**
  - a. Predikat ukuranlist(list, integer) untuk menghitung banyaknya elemen *list*; contoh *query* ukuranlist([1,2,3,4], N) menghasilkan N=4.
  - b. Predikat jumlahlist(list, integer) untuk menjumlahkan elemen-elemen dalam *list*; contoh *query* jumlahlist([1,2,3,4], N) menghasilkan N=10.
  - c. Predikat buangganjil(list1, list2) untuk menghasilkan *list* baru yang tidak mengandung bilangan ganjil di *list1*, *list1* mungkin kosong; contoh *query* buangganjil([-1,0,1,2,3,4], X) menghasilkan X=[0,2,4].
  - d. Predikat buangnegatif(list1, list2) untuk menghasilkan *list* baru yang tidak mengandung bilangan negatif di *list1*, *list1* mungkin kosong; contoh *query* buangnegatif([-1,0,2,-4], X) menghasilkan X=[0,2].

5. Terdapat sebuah program prolog sederhana sebagai berikut. (NILAI 20)

```
predikat1(L) :- predikat2(L,L).
predikat2([],[]).
predikat2([H|T],R) :- predikat2(T,T1), append(T1,[H],R).
```

a. Apakah hasil query berikut ini, yang diterapkan pada program di atas.

- i. predikat1([]).
- ii. predikat1([a]).
- iii. predikat1([a,b]).
- iv. predikat1([a,b,c]).
- v. predikat1([a,b,a]).

b. Jelaskan dengan singkat, apakah yang dilakukan oleh program ini.

6. Dari pernyataan di bawah ini, tentukan apakah pernyataan tersebut benar atau salah. Jika salah, tuliskan bagaimana seharusnya pernyataan yang benar. (NILAI 10)

- a. Prolog adalah salah satu contoh bahasa pemrograman dalam paradigma fungsional.
- b. Analisis kasus untuk merealisasikan suatu aturan (*rule*) yang bersifat ‘AND’ pada Prolog, dituliskan dengan membuat banyak aturan (*rule*) dengan nama aturan yang sama.
- c. *Red cut* pada Prolog berfungsi untuk meningkatkan efisiensi program, sedangkan penambahan *green cut* dapat mengubah arti program.
- d. Komponen dasar pada Expert System adalah Inference Engine saja.
- e. Kombinasi predikat Cut dan Fail pada Prolog, digunakan untuk merealisasikan *IF Statement*, artinya untuk memaksa program Prolog menjalankan kemungkinan lain dari predikat yang menerapkan Cut dan Fail tersebut.

**UAS SEMESTER I – 2016/2017**

Program Studi Teknik Informatika  
Institut Teknologi Bandung

**UJIAN AKHIR SEMESTER GANJIL 2016/2017**

**Matakuliah :** IF2121 Logika Informatika  
**Kelas :** K1, K2, dan K3  
**Waktu :** 120 menit  
**Sifat Ujian :** Closed-book.

**Bagian Relational Logic/ Predicate Logic**

1. Ubahlah ekspresi logika relasi di bawah ini menjadi bentuk klausul. (**NILAI : 10**)  
 $\exists w. \forall x. (\exists y. (p(x, y) \wedge r(y)) \leftarrow (\exists z. (q(w, z))))$
2. Jika huruf-huruf akhir alphabet (x,t,u,v,w,x,y,z) melambangkan variable, dan huruf-huruf awal alphabet (a,b,c) melambangkan konstanta. Tentukanlah apakah setiap pasangan ekspresi logika di bawah ini terunifikasi atau tidak. Jika terunifikasi carilah most-general-unifier nya. (**NILAI : 15**)
  - a)  $p(t(x, y), r(z, z))$  dan  $p(t(t(w, z), v), w)$
  - b)  $f(h(r, s, t), h(w, x, y), t, w)$  dan  $f(h(g(u, v), s, w), h(x, y, w), t, t)$
3. Jika  $\alpha$  dan  $\beta$  adalah unifier yang *pure* dari 2 buah ekspresi logika relasi, tentukanlah apakah pernyataan di bawah ini benar atau salah, serta kemukakan alasannya. (**NILAI : 15**)
  - a. Jika  $\beta$  adalah *unifier* tapi bukan *most general unifier* maka ada substitusi  $\mu$  sehingga  $\beta\mu$  adalah *most general unifier*.
  - b. Jika  $\alpha$  adalah *most general unifier* dan  $\beta$  adalah *most general unifier* maka  $\alpha\beta$  adalah *most general unifier* juga.
4. Jika diketahui *clausal-form* di bawah ini, gunakanlah prinsip resolusi untuk menghasilkan *empty-clause*. Jika tidak bisa dihasilkan *empty-clause* maka cukup ditulis : Tidak menghasilkan *empty-clause*. (**NILAI : 10**).
 

$\{p(x), q(x), r(x)\}$   
 $\{p(x), q(x), \neg r(x)\}$   
 $\{p(x), \neg q(x), r(x)\}$   
 $\{p(x), \neg q(x), \neg r(x)\}$   
 $\{\neg p(x), q(x), r(x)\}$   
 $\{\neg p(x), q(x), \neg r(x)\}$   
 $\{\neg p(x), \neg q(x), r(x)\}$   
 $\{\neg p(x), \neg q(x), \neg r(x)\}$

**Bagian Prolog**

5. Terdapat sebuah berkas ‘mahasiswa.txt’ berisi data nama mahasiswa, di mana setiap nama mahasiswa terdiri atas satu kata yang diawali huruf kecil, dan diakhiri dengan tanda titik. Banyaknya nama mahasiswa pada berkas tersebut tidak diketahui. Buatlah sebuah program sederhana dalam prolog untuk membaca berkas tersebut dan menyimpan hasil pembacaan ke sebuah list, kemudian menampilkan list tersebut ke layar. (**NILAI 15**)

Untuk pemeriksaan bahwa program sudah membaca sampai pada akhir berkas gunakan:

**at\_end\_of\_stream([nama\_stream])**

Contoh isi berkas ‘mahasiswa.txt’ adalah sebagai berikut:

gram Studi Teknik Informatika  
Institut Teknologi Bandung

|           |
|-----------|
| jonathan. |
| jordhy.   |
| dzar.     |
| faiz.     |
| roddy.    |
| kevin.    |

6. Terdapat sebuah program prolog sebagai berikut. (NILAI 9)

```
parent(charles, william).
parent(charles, harry).
parent(phillips, charles).
parent(elizabeth, charles).
parent(victoria, elizabeth).
ancestor(X, Y) :- parent(X, Y).
ancestor(X, Y) :- ancestor(X, Z), !, parent(Z, Y).
```

- Tentukan hasil dari query `ancestor(X, harry)`.
- Tentukan hasil dari query `ancestor(victoria, Y)`.
- Tentukan hasil dari query `ancestor(victoria, william)`.

7. Terdapat dua program Prolog sebagai berikut. (NILAI 11)

|                                                   |                                                         |
|---------------------------------------------------|---------------------------------------------------------|
| (i) <code>number_of_parents(adam, 0) :- !.</code> | (ii) <code>number_of_parents(adam, N) :- !, N=0.</code> |
| <code>number_of_parents(hawa, 0) :- !.</code>     | <code>number_of_parents(hawa, N) :- !, N=0.</code>      |
| <code>number_of_parents(X, 2).</code>             | <code>number_of_parents(X, 2).</code>                   |

- Jelaskan dengan singkat perbedaan cara kerja dari kedua program tersebut.
- Untuk masing-masing program 4(i) dan 4(ii), berikan hasil penerapan query `number_of_parents(john, X)`.
- Untuk masing-masing program 4(i) dan 4(ii), berikan hasil penerapan query `number_of_parents(hawa, 2)`.

8. Terdapat sebuah program prolog sebagai berikut. (NILAI 15)

```
male(arif).
male(pandu).
female(dyah).
female(arini).
parents(pandu, arif, dyah).
parents(arini, arif, dyah).
sister_of(X, Y) :- female(X), parents(X, M, F), parents(Y, M, F).
```

- Tentukan hasil dari query `sister_of(arini, Y)` yang diterapkan pada program tersebut.
- Berdasarkan hasil query program tersebut, tentukan apakah program tersebut sudah benar atau belum. Jika belum benar, tuliskan alasannya dan bagaimana program yang seharusnya.

--- SELAMAT MENGERJAKAN ---

**UAS SEMESTER I – 2017/2018****Bagian Relational Logic/ Predicate Logic**

1. Tentukan MGU dari pasangan predikat berikut ini. Jika tidak ada MGU, tuliskan tidak ada MGU. (Nilai 15)
  - a.  $r(x, g(x))$  dan  $r(g(x), y)$
  - b.  $p(a, x, f(g(y)))$  dan  $p(z, f(z), f(w))$
  - c.  $q(f(a), g(x))$  dan  $q(y, y)$
2. Untuk masing-masing nomor a) sampai dengan e) diketahui 2 buah *clausal form*. Lakukanlah prinsip resolusi dan tuliskan hasil penerapan resolusi tersebut beserta MGU nya. (Nilai 20)
  - a.  $\{p(x)\}$  dengan  $\{\neg p(f(x))\}$
  - b.  $\{p(x), p(y)\}$  dengan  $\{\neg p(x), \neg p(y)\}$
  - c.  $\{p(x,y), p(y,z)\}$  dengan  $\{\neg p(u, f(u))\}$
  - d.  $\{p(x,x), \neg r(x, f(x))\}$  dengan  $\{r(x,y), q(y,z)\}$
  - e.  $\{p(x,y), \neg p(x,x), q(x, f(x), z)\}$  dengan  $\{\neg q(f(x), x, z), p(x, z)\}$
3. Diberikan premis sebagai berikut:  $\forall x \forall y. (p(f(x), h(y)) \vee q(y))$   
 $\forall x. (\neg q(g(a)))$

Buktikan bahwa  $\exists y \forall x. (p(f(x), h(g(y))))$  dapat diturunkan dari premis dengan memanfaatkan *relational resolution* dengan langkah yang lengkap. (NILAI 20)

**Bagian Prolog**

4. Dalam domain peternakan kuda, diketahui bahwa kuda yang berharga adalah seekor induk kuda yang memiliki anak yang dapat berlari cepat. Dalam suatu peternakan terdapat empat ekor kuda bernama Comet, Prancer, Dasher, dan Thunder. Comet adalah induk dari Dasher dan Prancer. Dasher adalah induk dari Thunder. Dari pengalaman balap kuda, terlihat bahwa Prancer dan Thunder adalah kuda yang cepat. (Nilai 20)
  - a. Buatlah program sederhana dalam gprolog berdasarkan informasi di atas, menggunakan predikat: `is_a_horse/1`, `is_fast/1`, `is_parent_of/2`, dan `valuable/1`.
  - b. Tuliskan query untuk mencari kuda yang berharga.
  - c. Tuliskan jawaban dari program anda ketika mendapatkan query seperti pada butir (b).
5. Terdapat program prolog sederhana sebagai berikut. Program dibuat untuk menyatakan bahwa Pete menyukai semua jenis burger kecuali Big Kahuna Burger.

```

enjoys(pete,X) :- big_kahuna_burger(X), !.
enjoys(pete,X) :- burger(X).

burger(X) :- big_mac(X).
burger(X) :- big_kahuna_burger(X).
burger(X) :- whopper(X).

big_mac(a).
big_mac(c).
big_kahuna_burger(b).
whopper(d).

```

Berdasarkan program tersebut, jawablah pertanyaan berikut ini. **(Nilai 15)**

- a. Jawablah query di bawah ini jika diterapkan pada program tersebut.

|                        |                       |
|------------------------|-----------------------|
| (i) enjoys(pete,a) .   | (ii) enjoys(pete,b) . |
| (iii) enjoys(pete,c) . | (iv) enjoys(pete,d) . |
| (v) enjoys(pete,X) .   |                       |
  - b. Tentukan apakah program tersebut sudah benar atau belum berdasarkan jawaban butir (a). Jika sudah benar, cukup tuliskan sudah benar. Jika belum benar, tuliskan bagaimana seharusnya.
6. Tentukan apakah pernyataan di bawah ini BENAR atau SALAH. Jika SALAH, tuliskan pernyataan yang seharusnya. **(NILAI 10)**
- a. Prolog adalah salah satu contoh bahasa pemrograman dalam paradigma fungsional.
  - b. Analisa kasus untuk merealisasikan suatu aturan (*rule*) yang bersifat 'AND' pada Prolog, dituliskan dengan membuat kondisi berbeda pada banyak aturan (*rule*) dengan nama aturan yang sama.
  - c. *Red cut* pada prolog berfungsi untuk meningkatkan efisiensi eksekusi program, sedangkan penambahan *green cut* dapat mengubah arti program.
  - d. Fakta dalam Prolog dapat bersifat rekursif.
  - e. Kombinasi predikat Cut dan Fail pada Prolog, digunakan untuk merealisasikan IF Statement, artinya untuk memaksa program Prolog menjalankan kemungkinan lain dari predikat yang menerapkan Cut dan Fail tersebut.

**UAS SEMESTER I – 2019/2020**

**Waktu:** 150 menit  
**Sifat Ujian:** Closed-book

**Bagian Relational Logic/ Predicate Logic**

1. Tentukan MGU dari pasangan predikat berikut ini. Jika tidak ada MGU, tuliskan tidak ada MG. (Nilai 15)
  - a.  $p(f(y),w,g(z,y)) = p(x,x,g(z,a))$
  - b.  $q(a,f(a),f(a)) = q(z,g(u),g(u))$
  - c.  $r(x,g(f(a),u)) = r(g(u,v),x)$
2. Ubahlah pernyataan logika relasi sebagai berikut ke dalam bentuk klausal (Nilai 15)  
 $\sim(\sim\exists y\forall z(p(z,y)\leftrightarrow\sim\exists x(p(z,x)\wedge p(x,z))))$
3. Soal Prinsip Resolusi.
  - a. Diketahui premis seperti ini :
  $\forall x \forall y \forall w \forall z (penasaran(x, pesandihapus(w)) \vee penasaran(y, z))$   
 apakah bisa diambil kesimpulan bahwa :  
 $\exists u(penasaran(u, pesandihapus(naufal))) \wedge \sim \forall v(\sim penasaran(v, pesandihapus(naufal))).$   
 Jelaskan langkah demi langkah yang anda lakukan apabila ya atau tidak dengan menggunakan prinsip resolusi. Catatan :  $x, y, u, v, w, z$  adalah variable,  $naufal$  adalah konstanta,  $pesandihapus$  adalah fungsi dengan aritas 1, dan  $penasaran$  adalah predikat/relasi dengan aritas 2. (NILAI : 10)
  - b. Dari bentuk klausal di bawah ini ( $z1, z2, x$  adalah variable ;  $a$  adalah konstanta ;  $f$  adalah fungsi ; dan  $p$  adalah predikat) :
  $\{\sim p(z1, a), \sim p(z1, x), \sim p(x, z1)\}$   
 $\{p(z2, f(z2)), p(z2, a)\}$ 
    - i. Dengan menggunakan sekali resolusi apakah bisa dihasilkan  $p(a, a)$  ? Buktikan. (NILAI : 5)
    - ii. Dengan menggunakan sekali resolusi apakah bisa dihasilkan  $p(a, f(a))$  ? buktikan. (NILAI : 5)

**Bagian Prolog**

4. Buatlah sebuah program prolog sederhana sebagai berikut. (Nilai 20)
  - a. Buatlah sebuah predikat **isMember** dengan dua buah term yaitu sebuah variabel dan sebuah list. Predikat **isMember** akan mengembalikan no (False) jika variabel tidak terdapat dalam list, dan menghasilkan yes (True) jika variabel berada dalam list.
  - b. Buatlah sebuah predikat **intersectList** yang menerima dua buah list (L1 dan L2) dan menghasilkan satu buah list (L3) yang berisi nilai yang beririsan antara L1 dan L2. Jadi predikat **intersectList** memiliki 3 term yaitu dua term berupa dua buah list masukan dan sebuah term sebagai list hasil. Predikat **intersectList** harus memanfaatkan predikat **isMember**.

5. Terdapat program prolog sederhana sebagai berikut. Program bertujuan untuk meminta pengguna memasukkan nilai positif. (Nilai 15)

```
classify(0, zero).
classify(N, negative):-N<0.
classify(N, positive).
go:- write(start), nl,
 repeat, write('enter a positive value'), read(N),
 classify(N, Type), Type=positive,
 write('positive value is '), write(N), nl.
```

- a. Tentukan apa yang dihasilkan program jika diterapkan query go dan pengguna memasukkan nilai 0.
  - b. Tentukan apa yang dihasilkan program jika diterapkan query go dan pengguna memasukkan nilai -17.
  - c. Tentukan apakah program tersebut sudah benar atau belum berdasarkan jawaban butir (a) dan (b). Jika sudah benar, cukup tuliskan sudah benar. Jika belum benar, tuliskan bagaimana seharusnya.
6. Tentukan apakah pernyataan di bawah ini BENAR atau SALAH. Jika SALAH, tuliskan pernyataan yang seharusnya. (NILAI 15)
- a. Fakta dan aturan pada program Prolog dapat dikelompokkan secara terpisah atau tidak terpisah.
  - b.  $X \neq Y$  menyatakan nilai X dan Y adalah tidak sama (secara aritmetik).
  - c. Query dalam Prolog dapat bersifat rekursif.
  - d. Red cut digunakan untuk memanipulasi program sehingga suatu proses tertentu harus dilakukan.
  - e. Basis rekursif pada list (umumnya) adalah list dengan satu elemen. Elemen list adalah head dan tail. Head adalah list dengan satu elemen; tail adalah list sisanya.

--- SELAMAT MENGERJAKAN ---

IF2123

Aljabar Geometri

**UTS ALJABAR LINEAR DAN GEOMETRI**

UTS SEMESTER I – 2007/2008

20 Juli 2007

100 menit

$$v_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \quad v_2 = \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix}$$

1. Diketahui  $X = \{v_1, v_2\}$  dengan  $v_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$  dan  $v_2 = \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix}$ 
  - a. Tunjukkan bahwa  $X$  himpunan orthogonal
  - b. Tentukan vektor tak nol  $v_3 \in R^3$  sehingga  $\{v_1, v_2, v_3\}$  himpunan ortogonal
2. Dari himpunan bebas linier  $X = \{u, v, w\}$ , Ani dan Budi membentuk himpunan baru  $Y = \{u, u + v, u + v + w\}$ . Ani berpendapat bahwa himpunan  $Y$  juga bebas linier, sedangkan Budi berpendapat himpunan  $Y$  bergantung linier. Pendapat siapakah yang benar? Jelaskan jawaban Anda
3. Diketahui  $M_{2x2}$  himpunan semua matriks  $2 \times 2$  dengan komponen real adalah ruang vektor real dan  $K = \{A \in M_{2x2} \mid A = A^t\}$  dengan  $A^t$  menyatakan matriks transpos dari  $A$ 
  - a. Tunjukkan bahwa  $K$  membentuk subruang dari  $M_{2x2}$
  - b. Tentukan basis bagi  $K$
4. Misalkan  $x = \left( \frac{1}{\sqrt{5}}, -\frac{1}{\sqrt{5}} \right)$  dan  $y = \left( \frac{2}{\sqrt{30}}, \frac{3}{\sqrt{30}} \right)$ . Tunjukkan bahwa  $\{x, y\}$  adalah himpunan ortogonal di  $R^2$  dengan hasil kali dalam  $\langle u, v \rangle = 3u_1v_1 + 2u_2v_2$ , tetapi bukan himpunan ortogonal di  $R^2$  dengan hasil kali titik.

UTS SEMESTER I – 2008/2009  
 24 Oktober 2008  
 Waktu 90 menit

- $+ a_1x + a_2$        $x^2$  yang memenuhi
1. Tentukan koefisien  $a_0, a_1, a_2$  dari polinom berbentuk  
 $P(x) = a_0$   
 $P(-1) = 1, P(1) = 1$ , dan  $P(3) = -15$ .
  2. Diberikan dua vektor  $\bar{u} = (0,2,1)$  dan  $\bar{v} = (1,1,-2)$ 
    - a. Tentukan semua vektor  $\bar{w}$  yang tegak lurus terhadap  $\bar{u}$  dan  $\bar{v}$
    - b. Tentukan persamaan bidang yang dibangun oleh  $\bar{u}$  dan  $\bar{v}$  di ruang  $x$ - $y$ - $z$
  3.  $T: R^3 \rightarrow R^2$  pemetaan linear.  $S = \{(1,1,0), (01,1)\}$  basis dari  $V$  ruang vektor bagian  $R^3$  dan berlaku  $T(1,1,0) = (1,1)$  dan  $T(0,1,1) = (1,2)$ .
    - a. Tentukan rumus pemetaan  $T(x, y, z)$
    - b. Tentukan semua  $\bar{x} = (x, y, z)$  yang memenuhi  $T(\bar{x}) = 0$
  4. Tentukan nilai eigen  $\lambda$  dan vektor eigen yang berpadanan dari  

$$A = \begin{vmatrix} 1 & 0 \\ 0 & 1 \\ \hline 1 & -1 \\ \hline 1 \end{vmatrix}$$

## UTS SEMESTER I – 2009/2010

17 Oktober 2009

Waktu 100 menit

1. Tentukan semua Vektor yang tegak lurus terhadap vektor  $a$  dan  $b$  berikut

$$a = \begin{pmatrix} 4 \\ -2 \\ 4 \\ -2 \end{pmatrix} \quad b = \begin{pmatrix} 2 \\ 3 \\ 2 \\ -1 \end{pmatrix}$$

2. Diberikan matriks

$$A = \begin{pmatrix} 3 & 12 & 15 & 21 \\ 0 & 7 & 8 & 1 \\ 4 & 16 & 18 & 31 \\ 1 & 4 & 5 & 12 \end{pmatrix}$$

Gunakan Operasi Baris Elementer untuk membentuk matriks segitiga atas, kemudian hitung determinan matriks  $A$

3. Periksa apakah himpunan  $H$  berikut merupakan basis untuk  $\mathbb{R}^3$

$$H = \left\{ \begin{pmatrix} 4 \\ 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 8 \\ 3 \\ 2 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \right\}$$

4. Jika  $T : \mathbb{R}^3 \rightarrow \mathbb{R}^3$  suatu transformasi linear dengan sifat  $T \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 4 \\ 1 \\ 2 \end{pmatrix}$ ,  $T \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} -5 \\ -2 \\ 0 \end{pmatrix}$ ,  $T \begin{pmatrix} 1 \\ -1 \\ -1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ -6 \end{pmatrix}$ .

Tentukan  $T \begin{pmatrix} -2 \\ 1 \\ -1 \end{pmatrix}$

UTS SEMESTER I – 2010/2011

9 Oktober 2010

Waktu 100 Menit

1. Tentukan perpotongan empat bidang di ruang berikut:

$$x - y = 0$$

$$y + z = 4$$

$$-2x + 2y + z = 3$$

$$x + z = 4$$

2. Hitung determinan dari matriks:

$$A = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 4 & 3 & 2 & 4 \\ 3 & 0 & 0 & 5 \\ 0 & 5 & 0 & 0 \end{bmatrix}$$

3. Tentukan semua vektor yang tegak lurus terhadap  $(1, -1, 3)$  dan  $(0, 2, -1)$
4. Diberikan  $S = \{(1,1,-3), (2,1,1), (0,-1,7)\}$  himpunan vektor yang membangun ruang vektor  $V$ 
  - a. Tunjukkan  $(5,3,-1)$  anggota  $V$
  - b. Tentukan dimensi dari  $V$

5.  $T$  pemetaan linear dengan  $T\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}$ ,  $T\begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} -1 \\ 1 \\ 2 \end{pmatrix}$ ,  $T\begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 3 \end{pmatrix}$ . Tentukan  $\bar{x} = \begin{pmatrix} a \\ b \\ c \end{pmatrix}$  yang memenuhi

$$T(\bar{x}) = 2\bar{x}$$



## UTS SEMESTER I – 2011/2012

8 Oktober 2011

Waktu 100 menit

## Kelompok A

1. Diketahui matriks perluasan

$$\left| \begin{array}{cc|c} 1 & 3 & 2 \\ 1 & -2 & 1 \\ \hline & & 1 \\ & & 4 \end{array} \right| \quad \left| \begin{array}{cc} 0 & 5 \end{array} \right|$$

Terapkan Operasi Baris Elementer pada matriks tersebut dan tentukan nilai  $a$  sehingga SPL mempunyai banyak jawab.

2. Misalkan  $v = (v_1, v_2, v_3)$  vektor dengan panjang  $\sqrt{3}$  dan membentuk sudut  $\frac{\pi}{6}$  terhadap  $u = (1,4,2)$ .

Tuliskan sistem persamaan dalam  $v_1, v_2, v_3$  yang memenuhi kondisi tersebut

3. Tentukan vektor proyeksi dari  $a = (1,1,-1)$  pada  $b = (3,0,7)$
4. Hasil kali dalam (HKD) dari dua vektor di  $R^2$  didefinisikan  $\langle u, v \rangle = 5u_1v_1 + 3u_2v_2$  untuk  $u = (u_1, u_2)$ ,  $v = (v_1, v_2)$ . Periksa apakah  $u = (2, -5)$  dan  $v = (4, 7)$  tegak lurus terhadap HKD diatas?
5.  $T : R^3 \rightarrow R^2$  pementaan linear yang memenuhi

$$T(1,0,0) = (2,4), T(0,1,0) = (1,1), T(0,0,1) = (-1,0).$$

Hitung  $T(2, -4, 6)$ .

## Kelompok B

1. Diketahui persamaan bidang

$$\begin{aligned} 2x - y + z &= 1 \\ x - 2y + z &= 4 \\ 5x - 7y + 4z &= 13 \end{aligned}$$

- a. Tentukan semua titik yang merupakan perpotongan bidang tersebut
- b. Berikan arti geometri dari hasil perpotongan ketiga bidang
2.  $S$  ruang bagian dari  $R^3$  yang dibangun oleh  $X = \{(2,3,1), (1,0,2), (-1,-3,1)\}$  Tentukan himpunan bagian dari  $X$  yang merupakan basis

untuk  $S$ Periksa apakah vector  $w = (4, 7, 5)$  anggota dari  $S$ ?

UTS SEMESTER II – 2007/2008

26 Oktober 2007

100 menit

1. Diberikan Sistem Persamaan Linier (SPL):

$$\begin{aligned}x + 2y - 3z &= 4 \\2x - 3y + 8z &= -2 \\-x - 2y + (19 - k^2)z &= -k\end{aligned}$$

Tentukan semua nilai  $k$  sehingga SPL

- a. Tidak mempunyai solusi
  - b. Mempunyai tak berhingga banyaknya solusi
  - c. Hanya mempunyai 1 solusi
2. Diberikan persamaan garis di  $R^2$ :  $y + 2x = 0$
- a. Tentukan himpunan solusi dari  $y + 2x = 0$
  - b. Dengan menerapkan hasil kali titik, gunakan hasil (a) untuk menentukan semua vektor satuan yang ortogonal terhadap garis  $y + 2x = 0$
3. Periksa apakah himpunan berikut berbruang dari  $R^3$  (berikan penjelasan atas jawaban Anda)
- a.  $S = \{(x, y, z) \mid x + 2y - 3z = 1\}$
  - b.  $T = \{(0,0,0) \mid (1,2,1)\}$
4. Periksa apakah  $\begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}$  di ruang kolom matriks  $\begin{pmatrix} 1 & 2 & 1 \\ 2 & 3 & 2 \\ 2 & 3 & 2 \end{pmatrix}$
5. Misalkan  $R^3$  adalah ruang vektor yang dilengkapi dengan hasil kali dalam  $\langle u, v \rangle = u_1v_1 + 2u_2v_2 + 4u_3v_3$  untuk setiap  $u = (u_1, u_2, u_3)$ ,  $v = (v_1, v_2, v_3)$ . Gunakan proses Gram – Schmidt untuk mengubah basis  $\left\{(1,2,1), (0,-1,1), (0,0,\frac{1}{2})\right\}$  menjadi sebuah basis orthogonal.

## UTS SEMESTER II – 2008/2009

5 Desember 2008

90 menit

1. Diberikan

$$f(x) = xy - y^3 \text{ dan } r(t) = t^2 i - (2t - 1) j$$

- a. Tentukan  $r(t) = t^2 i - (2t - 1) j$  agar  $r(0) = 4i - 3j$
- b. Hitung turunan berarah dari  $f$  dalam arah yang sama dengan  $r'(t)$  pada  $(4, -3)$

2. Hitung kerja (integral garis) oleh medan gaya
- $r(t) = t^2 i - (2t - 1) j$
- sepanjang lintasan

$$C : r(t) = 3ti - tj + t^2 k \text{ dalam selang waktu } 0 \leq t \leq 1$$

3. Diberikan medan vektor
- $F(x, y) = (\cos \pi y + 2xy)i + (x^2 - 3 - \pi x \sin \pi y)j$

- a. Tunjukkan bahwa
- $F$
- merupakan medan vektor konservatif

b. Tentukan medan skalar  $f$  dan  $f(0,1) = -2$   
yang memenuhi  $F = \nabla f$

4. Diketahui medan kecepatan

$$r(t) = t^2 i - (2t - 1) j$$

- a. Tentukan divergensi dari

$$V(x, y)$$

- b. Hitung fluks dari
- $V(x, y)$
- yang menembus lintasan tertutup
- $(x + 1)^2 + (y - 3)^2 = 4$

UTS SEMESTER II – 2009/2010

5 Desember 2009

100 menit

1. Hitung turunan berarah  $f(x, y) = xy$  sepanjang garis  $3y = x - 1$  pada saat  $x = 4$  dan  $y = 1$
2. Sebuah partikel bergerak dengan posisi setiap saat dinyatakan oleh

$$R(t) = \sin 2t\mathbf{i} - 2 \cos t\mathbf{j} + t\mathbf{k}$$

Tentukan percepatan partikel setiap saat selalu tegak lurus terhadap kecepatannya.

3. Hitung kerja yang dilakukan oleh gaya

$$F(x, y) = x\mathbf{i} + \sin y\mathbf{j}$$

Untuk memindahkan partikel sepanjang lintasan berbentuk seperempat lingkaran dari titik  $(0, -\pi)$  ke titik  $(\pi, 0)$  dengan arah berlawanan perputaran jarum jam

4. a) Tuliskan teorema Green

- b) Hitung  $\int_C (y - \sqrt{x})dx - (x + \sin y)dy$  dengan  $C$  lintasan yang merupakan batas segitiga yang mempunyai titik sudut  $(0,0)$ ,  $(10,0)$  dan  $(4,5)$ .

## UTS SEMESTER II – 2010/2011

20 November 2010

100 menit

1. Sebuah partikel bergerak setiap saat mengikuti lintasan

$$C : \vec{r}(t) = (1-t)\mathbf{i} + t\mathbf{j} + (3t^2 - t)\mathbf{k}$$

- a. Berapa kecepatan partikel pada saat berada pada posisi  $(0, 1, 2)$  dalam koordinat Kartesius?
  - b. Pernahkah partikel berhenti? Beri alasan.
2. Temperatur di dalam ruang pada koordinat  $(x,y,z)$  dinyatakan dalam

$$T(x, y, z) = \frac{z}{x^2 + y^2}$$

Tentukan vektor  $\vec{u}$  agar turunan dalam arah tersebut bernilai terkecil pada titik  $(0, 1, 2)$ .

3. Diberikan medan vektor  $\vec{F}(x, y) = e^y \mathbf{i} + (xe^y - 2y)\mathbf{j}$ 
  - a. Tunjukkan  $\vec{F}$  medan vektor konservatif
  - b. Tentukan medan skalar  $u(x, y)$  yang memenuhi  $\vec{F} = \nabla u$
4. Hitung

$$\int_C ydx - xdy - zdz$$

Sepanjang lintasan  $C : \vec{r}(t) = \cos t\mathbf{i} + \sin t\mathbf{j} + 2t\mathbf{k}, 0 \leq t \leq 2\pi$

5. Hitung kerja untuk menggerakkan partikel sepanjang batas/tепи trapezium dalam satu putaran penuh, dengan titik sudut  $(0,0), (1,0), (2,2)$  dan  $(0,2)$  menggunakan gaya

$$\vec{F}(x, y) = \left( 2xy + \frac{1}{2}y^2 - 3y \right) \mathbf{i} + (x^2 + xy) \mathbf{j}$$

## UTS SEMESTER II – 2011/2012

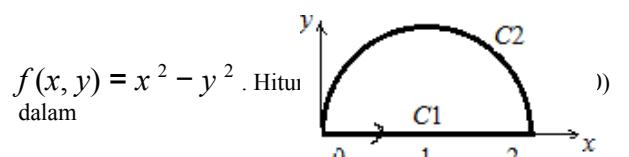
26 November 2011

100 menit

Kelompok A

1. Nyatakan fungsi vektor untuk lintasan setengah lingkaran  $C_1 \cup C_2$  yang berawal dan berakhir di  $(0,0)$ .

2. Diberikan



arah  $v = (2,1)$ .

3. Diberikan

$$f(x, y, z) = x^2 + y^2 + z^2. \text{ Hitung } \nabla f \text{ dan } \operatorname{curl}(\nabla f).$$

4. Tentukan  $g(u, v)$

dengan

$$u = x - 2y,$$

$$v = x + y. (R$$

sehingga  $\iint f($

$$x, y) dx dy = \iint$$

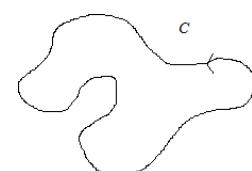
$$g(u, v) du dv$$

 $R$  $R$ 

adalah daerah pengintegralan hasil transformasi).

5.  $\vec{F}$  merupakan medan vektor di bidang dan  $f$  medan skalar yang memenuhi  $\vec{F} = \nabla f$ .

Jika  $C$  adalah lintasan seperti pada gambar, berapa nilai  $\int \vec{F} \cdot d\vec{r}$  dan jelaskan jawaban

 $C$ 

Saudara.

Kelompok B

1. Diberikan medan vektor  $\vec{F}(x, y, z) = \cos(yz)\vec{i} + xz \sin(yz)\vec{j} - xy \sin(yz)\vec{k}$ . Tunjukkan bahwa  $\vec{F}$  medan vektor konservatif, kemudian tentukan medan skalar  $f(x, y, z)$  agar berlaku  $\vec{F} = \nabla f$ .

2. Hitung

$$\int_C (e^{\cos x} - y)dx + (2x + \ln y)dy$$

*C*

Sepanjang lintasan  $C : r(t) = \begin{pmatrix} \cos 2t \\ \sin 2t \end{pmatrix}, 0 \leq t \leq \pi$

UTS SEMESTER I – 2012/2013  
 MA-2072 Matematika Teknik  
 I 12 Oktober 2012  
 Waktu 100 menit

**BAGIAN A (Bobot 5 per soal)**

1. Diberikan matriks

$$A = \begin{bmatrix} 1 & 3 & -1 & -1 \\ 2 & 6 & -2 & 4 \\ 0 & 0 & 1 & -3 \\ 0 & -2 & 1 & 1 \end{bmatrix}$$

- a. Gunakan Operasi Baris Elementer untuk membentuk segitiga atas.
- b. Hitung determinan  $A$ .

2.  $V$  ruang vektor yang dibangun oleh

$$S = \left( u_1 = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, u_2 = \begin{pmatrix} 0 \\ -1 \\ 1 \end{pmatrix}, u_3 = \begin{pmatrix} 2 \\ -2 \\ 4 \end{pmatrix} \right)$$

Nyatakan vektor  $w = \begin{pmatrix} -5 \\ 2 \\ 7 \end{pmatrix}$  sebagai kombinasi linear dari  $u_1, u_2, u_3$ , jika mungkin.

3. Diberikan

$$S = \left( u_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, u_2 = \begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix}, u_3 = \begin{pmatrix} -1 \\ -2 \\ 1 \end{pmatrix} \right)$$

Periksa apakah  $S$  bebas linear.

4. Dari 3 vektor berikut

$$u = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix}, v = \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix}, w = \begin{pmatrix} -3 \\ 1 \\ 1 \end{pmatrix}$$

hitung  $w \cdot (u \times v)$ .

5. Tentukan semua vektor  $u$  yang memenuhi

$$\begin{pmatrix} 2 & 1 \\ -2 & -1 \end{pmatrix}u = u$$

**BAGIAN B (Bobot 10 per soal)**

1. Perhatikan sistem persamaan linear

$$A \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} 2 \\ -2 \\ 1 \end{pmatrix} \text{ dengan } A = \begin{pmatrix} 1 & 2 & 0 \\ -1 & -3 & -1 \\ 0 & 2 & 1 \end{pmatrix}$$

- a. Tentukan invers dari  $A$ .
- b. Tentukan  $x, y$ , dan  $z$  yang memenuhi sistem persamaan di atas.

2.  $T : R^3 \rightarrow R^2$  transformasi linear yang memenuhi

$$T \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, T \begin{pmatrix} -1 \\ -1 \\ 2 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}, T \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \end{pmatrix}$$

Hitung  $T \begin{pmatrix} 3 \\ 1 \\ 4 \end{pmatrix}$

## UTS SEMESTER I – 2014/2015

IF2123 Aljabar  
Geometri Jumat, 24  
Oktober 2014  
Waktu 120 menit

1. Tunjukkan bahwa vektor  $(1, 1, -1)^T$ ,  $(3, 1, 2)^T$ , dan  $(-2, 1, 2)^T$  saling independen dan tunjukkan bahwa vektor  $(-3, 2, -2)^T$  dapat dinyatakan sebagai kombinasi linier dari vektor-vektor tersebut.

2. Hitunglah determinan dari matriks berikut:

$$\begin{pmatrix} 1 & x & x^2 \\ 1 & y & y^2 \\ 1 & z & z^2 \end{pmatrix}$$

Tuliskan dalam bentuk paling sederhana.

3. Misalkan

$$S = \{(x_1, x_2, x_3)^T : x_1 = x_2 + x_3\}$$

Tunjukkan bahwa S adalah ruang bagian dari  $\mathbb{R}^3$ .

4. Misalkan transformasi linier  $T : \mathbb{R}^3 \rightarrow \mathbb{R}^3$  didefinisikan :

$$T(x_1, x_2, x_3)^T = (x_1 + x_2 + x_3, 2x_2, x_1 + x_3)^T$$

- a. Tentukan matriks transformasi linier di atas.
- b. Tentukan basis dari *null space* dan basis dari *range*.

UTS SEMESTER I – 2015/2016  
 IF2123 Aljabar  
 Geometri Jumat, 23  
 Oktober 2015  
 Waktu 100 menit

12. Berapakah *rank* dari matriks di bawah ini?

$$A = \begin{pmatrix} 1 & -1 & 2 & -1 \\ 2 & 1 & -2 & -2 \\ -1 & 2 & -4 & 1 \\ 3 & 0 & 0 & -3 \end{pmatrix} \quad (10)$$

13. Kondisi-kondisi apakah yang harus dipenuhi oleh  $b_1$ ,  $b_2$ , dan  $b_3$  agar SPL berikut konsisten?

$$\begin{aligned} x_1 - x_2 + 3x_3 &= b_1 \\ 3x_1 - 3x_2 + 9x_3 &= b_2 \\ -2x_1 + 2x_2 - 6x_3 &= b_3 \end{aligned}$$

(jawaban anda kira-kira seperti ini:  $b_1 = 2b_2$ ,  $b_3 = -b_1$ , dll) (10)

$$14 \left| \begin{array}{ccc|c} & a & b & c \\ & d & e & f = 5 \\ & g & h & i \end{array} \right. \text{ Jika } \left| \begin{array}{ccc|c} -a & -b & -c \\ 2d & 2e & 2f \\ -g & -h & -i \end{array} \right. \text{ berapakah } \quad (10)$$

15. Buktikan kesamaan  $\| \mathbf{u} + \mathbf{v} \| + \| \mathbf{u} - \mathbf{v} \| \geq 2\|\mathbf{u}\| + 2\|\mathbf{v}\|$  untuk vektor-vektor di  $R^n$ . (15)

16. Untuk nilai riel  $\lambda$  yang manakah vektor-vektor  $\mathbf{u} = (\lambda, -1/2, -1/2)$ ,  $\mathbf{v} = (-1/2, \lambda, -1/2)$ ,  $\mathbf{w} = (-1/2, -1/2, \lambda)$  membentuk himpunan yang tak bebas lanjar (*linearly dependent*) di dalam  $R^3$ ? (15)

6. Diketahui dua buah vektor di  $R^3$  yaitu  $\mathbf{v} = (-1, 3, 2)$ ,  $\mathbf{w} = (1, 1, -1)$ .
- Carilah vektor  $\mathbf{x}$  sedemikian sehingga  $\mathbf{x} = \mathbf{v} \times \mathbf{w}$ .
  - Hitunglah norma vektor  $\mathbf{x}$
  - Tentukan jenis sudut yang dibentuk oleh vektor  $\mathbf{v}$  dan  $\mathbf{w}$ , yaitu apakah lancip, tumpul, atau siku-siku?
  - Misalkan  $\theta$  adalah sudut yang dibentuk oleh  $\mathbf{v}$  dan  $\mathbf{x}$ , tentukan  $\cos \theta$  (20)

Tinjaulah basis  $S = \{ \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3 \}$  untuk  $R^3$  yang dalam hal ini  $\mathbf{v}_1 = (1, 2, 3)$ ,  $\mathbf{v}_2 = (2, 5, 3)$ , dan  $\mathbf{v}_3 = (1, 0, 10)$ . Carilah sebuah rumus untuk transformasi lanjar  $T : R^3 \rightarrow R^2$  sehingga  $T(\mathbf{v}_1) = (1, 0)$ ,  $T(\mathbf{v}_2) = (1, 0)$ , dan  $T(\mathbf{v}_3) = (0, 1)$ , lalu hitunglah  $T(1, 1, 1)$ . (20)

**UTS SEMESTER I – 2016/2017**  
Mata kuliah: Aljabar Geometri (IF2123)  
Hari/tanggal: Selasa, 11 Oktober 2016  
Waktu: 90 menit

---

1. Tentukan nilai dari bilangan kompleks berikut dalam bentuk  $z = x + iy$ 
  - a.  $\sqrt[4]{-8 + 6i}$
  - b.  $\log(-100i)$
  - c.  $(2 + i)^i$

Jika quaternion  $q_1 = 1 + i2 + j3 + k4$ , dan  $q_2 = 2 - i + j5 - k2$ , hitunglah:

- d.  $q_1 q_2$
  - e.  $q_1 q_2^{-1}$
2. Selidiki apakah vektor **a** merupakan kombinasi linier dari vektor **b**, **c**, dan **d**, di mana
$$\mathbf{a} = \begin{pmatrix} 4 \\ 5 \\ 3 \end{pmatrix}, \mathbf{b} = \begin{pmatrix} 2 \\ 3 \\ 1 \end{pmatrix}, \mathbf{c} = \begin{pmatrix} 4 \\ -1 \\ -5 \end{pmatrix}, \text{ dan } \mathbf{d} = \begin{pmatrix} -3 \\ 1 \\ 4 \end{pmatrix}$$
3. Diberikan titik-titik A(-4,4,-2), B(8,-4,-2), dan C(1,2,-1).
  - a. Tentukan persamaan garis AB dan AC
  - b. Jika  $a = \overrightarrow{AB}$  dan  $b = \overrightarrow{AC}$ , hitunglah  $(a \wedge b)$ .
  - c. Tentukan luas  $\Delta ABC$ .
4. Tentukan letak titik P(2,-3,4) jika diputar sebesar  $120^\circ$  dengan sumbu putarnya adalah  $\vec{u} = -2\hat{i} - 4\hat{j} + 4\hat{k}$ .

**UTS SEMESTER I – 2017/2018**  
Mata kuliah: Aljabar Geometri (IF2123)  
Hari/tanggal: Rabu 11 Oktober 2016  
Waktu: 90 menit

---

1. Diberikan tiga buah matrik A, B, C sebagai berikut :

$$A = \begin{bmatrix} 3 & 0 \\ -1 & 2 \\ 1 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 4 & -1 \\ 0 & 2 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 4 & 2 \\ 3 & 1 & 5 \end{bmatrix}$$

a). Hitunglah  $(3A^T - 2C)^T$

b). Hitunglah  $(C^T B) \cdot A^T$

2. Diberikan matriks A berukuran  $n \times n$  sebagai berikut :

$$A = \begin{bmatrix} a_{11} & 0 & \dots & 0 \\ 0 & a_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & a_{nn} \end{bmatrix}, \quad \text{dimana } a_{11}, a_{22}, \dots, a_{nn} \neq 0$$

a). Tentukan matriks balikan dari matriks A

b). Dengan menggunakan matriks balikan dari jawaban a), tentukan solusi  $Ax = b$  jika  $b = (a_{11}, a_{22}, \dots, a_{nn})^T$

3. Dengan menggunakan reduksi baris dan ekspansi ko-faktor, tentukan determinan matriks berikut :

$$\begin{bmatrix} 1 & 3 & 1 & 5 & 3 \\ -2 & -7 & 0 & -4 & 2 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 2 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

4. Diberikan dua persamaan bidang sebagai berikut :

$$3x - 4y + z = 1$$

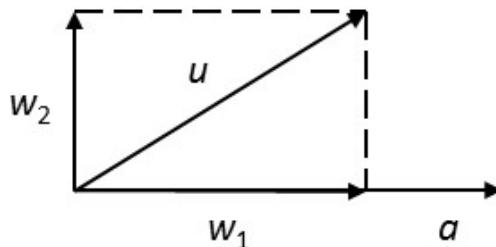
$$6x - 8y + 2z =$$

3

a). Periksa apakah dua bidang tersebut parallel atau berpotongan.

b). Jika bidang tersebut parallel tentukan jarak antara keduanya.

5. Perhatikan gambar berikut



$w_1$  adalah projeksi vektor  $u=(2,1,1,2)$  pada vektor  $a=(4,-4,2,-2)$ , sedangkan  $w_2$  adalah vektor orthogonal dengan vektor  $a$ . Jika vektor  $u$  dinyatakan sebagai  $w_1 + w_2$ , tentukan  $w_1$  dan  $w_2$ .

6. Tentukan normal dari bidang yang melewati tiga titik P(9,0,4), Q(-1,4,3), dan R(0,6,-2), kemudian tentukan persamaan bidangnya.



**UTS SEMESTER I – 2019/2020**

Mata kuliah: Aljabar Geometri (IF2123)

Hari/tanggal: Rabu, 9 Oktober 2019

Waktu: 120 menit

1. Diberikan sebuah SPL dalam bentuk matriks *augmented* sbb:

$$\begin{bmatrix} 1 & 2 & -3 & 4 \\ 3 & -1 & 5 & 2 \\ 4 & 1 & a^2 - 14 & a + 2 \end{bmatrix}$$

Tentukan nilai  $a$  demikian sehingga SPL tersebut:

- a). Mempunyai solusi banyak (tak hingga)
  - b). Tidak mempunyai solusi
  - c). Mempunyai solusi tunggal, dan tentukan solusinya
2. Diketahui matriks:

$$A = \begin{bmatrix} 2 & 1 & 2 \\ 0 & 3 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 2 & 2 & 0 & 1 \\ 1 & -1 & -2 & 2 \\ -1 & 0 & 0 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & -3 \\ -1 & -2 & 1 \end{bmatrix}$$

Tentukan nilai:

- a).  $((A + C)B)^T$
  - b).  $((AA^T)(CC^T))^{-1}$
  - c).  $\begin{bmatrix} A \\ C \end{bmatrix}B$
3. Diberikan SPL dengan empat variabel sebagai berikut:

$$\begin{aligned} x + y + z + 2w &= 4 \\ 3x + 2y - z + w &= 3 \\ 2x + 2y + z - w &= 10 \\ 4x - 3y + z - 2w &= 3 \end{aligned}$$

- a). Gunakan aturan *Cramer* untuk menghitung  $y$  saja.
  - b). Tentukan matrik *echelon* dari SPL tsb.
  - c). Tentukan nilai  $x, z, w$  dengan eliminasi Gauss.
4. . Diketahui matriks  $A$  sebagai berikut:

$$A = \begin{bmatrix} 2 & 5 & 5 \\ -1 & -1 & 0 \\ 2 & 4 & 3 \end{bmatrix}$$

- a). Tunjukkan matriks  $A$  invertible (tanpa menghitung  $A^{-1}$ )
  - b). Tentukan matrik kofaktor dari matriks  $A$
  - c). Hitunglah  $A^{-1}$
5. Diketahui tiga buah vektor  $u=(2,-6,2)$ ,  $v=(0,4,-2)$ ,  $w=(2,2,-4)$ .
- a). Perlihatkan apakah  $\{u,v \text{ dan } w\}$  merupakan himpunan orthogonal
  - b). Tentukan panjang vektor proyeksi orthogonal  $u$  pada vektor  $w$
  - c). Tentukan luas parallelogram yang dibentuk vektor  $v$  dan  $w$
  - d). Tentukan volume parallelepiped yang dibentuk vektor  $u$ ,  $v$ , dan  $w$
6. Diberikan 4 buah titik di ruang yakni,  $A(0,1,-1)$ ;  $B(1,1,2)$ ;  $C(2,2,1)$ ,  $P(3,3,3)$
- a). Tentukan persamaan bidang yang melewati titik  $A,B$ , dan  $C$  dalam bentuk vektor.
  - b). Pertanyaan sama dengan a) dengan menggunakan normal bidang
  - c). Hitunglah jarak titik  $P$  ke bidang tersebut.
  - d). Hitunglah luas segitiga  $ABC$ .

PENILAIAN TIAP NOMOR: 15-15-15-15-20-20

## UTS ALGEO 2020/2021

### MATRIKS

1. Jika A dan B adalah matriks yang bisa dilakukan inverse, maka  $\text{inverse}(AB)$  adalah

a. Q sama dengan  $\text{inverse}(BA)$

**b. sama dengan  $\text{inverse}(B).\text{inverse}(A)$**

c. sama dengan  $\text{inverse}(\text{transpose}(BA))$

d. sama dengan  $\text{inverse}(\text{transpose}(A), \text{Transpose}(B))$

e. Semua jawaban salah.

2. Diketahui matriks a sebagai berikut, manakah dari pernyataan berikut yang benar?

a. Kolom matriks tidak bebas linear

b. Matriks A memiliki nilai determinan -1

c. Matrik A tidak memiliki inverse/balikan

d. Transpose Matrik A memiliki nilai determinan -1

e. Semua jawaban di atas benar

**f. Semua jawaban di atas salah**

3. Asumsikan bahwa B adalah matriks  $3 \times 3$  dengan properti/karakteristik  $B^2 = B$ . Manakah dari pernyataan berikut tentang matriks B HARUS benar:

a. Matriks B memiliki inverse

b.  $\det(B) = 0$

c.  $\det(B) = \det(-B)$

**d.  $\det(B^5) = \det(B)$**

e. Semua jawaban di atas benar

f. Semua jawaban di atas salah

4. Diketahui matriks A dan matriks B yang keduanya berukuran  $n \times n$ . Hubungan antara balikan (inverse), determinan, dan transpose matriks adalah sebagai berikut

a.  $\text{Transpose}(AB) = \text{Transpose}(B) \times \text{Transpose}(A)$

b.  $\text{Determinan}(\text{Transpose}(A)) = (-1) \times \text{Determinan}(A)$

c.  $\text{Determinan}(\text{Inverse}(A)) = 1/\text{Determinan}(A)$

d. Jawaban A, B, dan C benar

**e. Hanya jawaban A dan C benar**

f. Semua jawaban salah

### SISTEM PERSAMAAN LINEAR

5. Diketahui sistem persamaan linear sebagai berikut :

$$y_1 + 6y_2 + 4y_3 = 1$$

$$2y_1 + 4y_2 - y_3 = 2$$

$$-y_1 + 2y_2 + 5y_3 = 3$$

a.  $y_1 = 20$

b.  $y_2 = 19$

c.  $y_3 = 19$

d.  $y_3 = 20$

e.  $y_1 = 19$

**f. Semua jawaban salah**

6. Diketahui sistem persamaan linear sebagai berikut :

$$x_1 + x_3 = 3$$

$$x_1 - x_2 - x_3 = 1$$

$$-x_1 + x_2 = 4$$

a. Tidak konsisten tanpa solusi

b. Tidak konsisten dengan banyak solusi

c. Konsisten dengan banyak solusi

**d. Konsisten dengan solusi unik**

e. Semua jawaban di atas benar

f. Semua jawaban di atas salah

7. Pada sistem persamaan linier (SPL) homogen  $Ax = 0$ , jika balikan matriks A tidak ada, maka itu artinya

a. SPL tidak memiliki solusi

b. SPL memiliki solusi non-trivial

c.  $\det(A) = 0$

d. SPL memiliki banyak solusi

**e. Jawaban B, C, D benar**

- f. Jawaban A, C benar
8. Diberikan sistem persamaan linier berikut. Solusinya adalah :
- $z = -t$ ,  $t$  elemen R
  - $x = 2 - t$ ,  $t$  elemen R
  - $y = 1 + t$ ,  $t$  elemen R
  - Hanya A dan B benar
  - Hanya B dan C benar
  - Semua jawaban tidak ada yang benar
9. Diketahui SPL dengan matriks augmented sebagai berikut. Nilai  $a$  dan  $b$  menentukan apakah SPL memiliki solusi unik, solusi banyak (dalam bentuk parametrik), atau tidak ada solusi. Jawaban mana yang benar?
- Mempunyai solusi unik jika  $a$  tidak sama dengan 0 dan  $b$  sama dengan 2
  - Mempunyai solusi dengan satu parameter jika  $a$  tidak sama dengan 0 dan  $b$  tidak sama dengan 2
  - Mempunyai solusi dengan dua parameter jika  $a$  sama dengan 0 dan  $b$  sama dengan 2
  - Tidak mempunyai solusi jika  $a$  sama dengan 0 dan  $b$  sama dengan 2
  - Semua jawaban di atas benar
  - Semua jawaban di atas salah
- DETERMINAN**
10. Diketahui matriks di bawah ini. Pilihlah satu jawaban yang paling benar terkait dari matriks di bawah ini :
- $$A = \begin{bmatrix} 2 & -1 & 3 \\ 1 & 2 & 4 \\ 5 & -3 & 6 \end{bmatrix}$$
- determinan matriks ini adalah -5
  - Determinan matriks ini adalah 5
  - Salah satu adjoint dari matriks  $A$  tersebut adalah 5
  - Jawaban pilihan kedua dan ketiga di atas adalah benar
  - Jawaban kesatu dan ketiga di atas adalah benar
  - Semua jawaban salah.
11. Jika  $A$  adalah matriks yang mempunyai invers berukuran  $n \times n$ , maka pernyataan berikut yang benar adalah:
- $Q(A)$ . Jika barisnya dipertukarkan (swap) sebanyak  $n$  kali (menjadi  $A'$ ), maka  $\det(A') = -\det(A)$
  - Jika dilakukan OBE saja sampai terbentuk matriks eselon, maka  $\det(A)$  tidak berubah
  - Jika baris ke- $n$  di kalikan dengan  $k$  maka determinant  $A = \det(kA)$
  - Jawab A dan B yang benar
  - Jawab B dan C yang benar
12. Berapa nilainya?
- $$\begin{vmatrix} 1 & 3 & 1 & 5 & 3 \\ -2 & -7 & 0 & -4 & 2 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 2 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \end{vmatrix}$$
- 1
  - 2
  - 3
  - 2/3
  - 1/2
  - Semua jawaban salah
13. Diberikan matriks  $A$  sebagai berikut. Agar  $\det(A) = 0$ , maka nilai-nilai  $a$  yang memenuhi adalah:
- 2
  - 5
  - 4
  - 4
  - 5
  - Semua jawaban salah
- $$A = \begin{bmatrix} a-4 & 4 & 0 \\ -1 & a & 0 \\ 0 & 0 & a-5 \end{bmatrix}$$

### VEKTOR DI RUANG EUCLIDEAN

14. Diketahui dua buah bidang di  $R^3$  dengan persamaan: 1.  $2x+y-z=3$  dan 2.  $2x-y+z=1$ . Kedua bidang tersebut perpotongan di suatu garis. Sudut lancip yang terbentuk antara kedua bidang tersebut adalah:
- Kurang dari  $30^\circ$

- b. Antara  $30^\circ$  dan  $45^\circ$   
c. Antara  $45^\circ$  dan  $60^\circ$   
**d. Antara  $60^\circ$  dan  $90^\circ$**   
e.  $90^\circ$
15. Diketahui vektor-vektor sbb:  $U = (1, 2, 3)$   $V = (2, 3, 7)$  dan  $W = (0, 0, 0)$ . Manakah di bawah ini yang benar  
a. Vektor-vektor tersebut tidak bebas linear  
b. Vektor-vektor tersebut bebas linear  
c. Tidak bisa ditentukan apakah bebas linear atau tidak bebas linear  
d. Tidak terdefinisi  
e. semua jawaban benar  
f. semua jawaban salah
16. Dua gaya yang besarnya dalam perbandingan 3: 5 memberikan resultan 35N. Jika sudut kemiringannya  $60^\circ$ , hitung besarnya masing-masing gaya.  
a. 15N dan 35N  
**b. 15N dan 25N**  
c. 10N dan 15N  
d. 20N dan 5N  
e. 20N dan 15N  
f. 10N dan 25N
17. Jika sudut antara dua vektor (keduanya berskala bukan nol) lebih besar dari  $90^\circ$  dan lebih kecil dari  $270^\circ$ , maka produk skalar (perkalian titik) dari vektor-vektor ini adalah:  
a. Positif  
**b. Negatif**  
c. Positif bila sudut lebih kecil dari  $180^\circ$ , Negatif bila sudut lebih besar dari  $180^\circ$   
d. Tidak dapat diketahui  
e. Semua jawaban di atas salah  
f. Semua jawaban di atas benar
18. Diketahui tiga buah titik di R3, yaitu A(0, 1, -1), B(1, 1, 2), dan C(2, 2, 1). Misalkan vektor  $u = AB$  dan vektor  $v = AC$ . Vektor  $u$  diproyeksikan ke vektor  $v$ . Maka dapat disimpulkan bahwa  
a. Proyeksi  $u$  pada  $v$  adalah vektor  $(16/9, 8/9, -16/9)$   
**b. Panjang proyeksi  $u$  pada  $v$  adalah  $8/3$**   
c. Komponen  $u$  yang tegak lurus pada  $v$  adalah  $(-7/9, -8/9, -11/9)$  O D)  
d. Jawaban A, B, dan C benar  
e. Hanya jawaban B dan C benar  
f. Semua jawaban salah
19. Diketahui tiga buah titik di R3, yaitu A(0, 1, -1), B(1, 1, 2), dan C(2, 2, 1). Misalkan vektor  $u = AB$  dan vektor  $v = AC$ . Vektor yang ortogonal dengan vektor  $u$  dan  $v$  adalah  
a. (3, 4, -1)  
b. (-3, 4, -1)  
c. (-3, 4, 1)  
d. (-3, -4, 1)  
e. (-3, -4, -1)  
f. Semua jawaban salah
20. Diketahui tiga buah titik di R3, yaitu A(0, 1, -1), B(1, 1, 2), dan C(2, 2, 1). Misalkan vektor  $u = AB$  dan vektor  $v = AC$ . Persamaan bidang yang melalui titik A adalah  
**a.  $-3x + 4y + z - 3 = 0$**   
b.  $3x - 4y - z + 3 = 0$   
c.  $-3x + 4y - z - 3 = 0$   
d.  $3x - 4y + z + 3 = 0$   
e.  $-3x - 4y - z - 3 = 0$   
f. Semua jawaban di atas salah

21. Diketahui tiga buah titik di R<sup>3</sup>, yaitu A(0, 1, -1), B(1, 1, 2) dan C(2, 2, 1). Misalkan vektor  $u = AB$  dan vektor  $v = AC$ . Diketahui terdapat suatu persamaan bidang yang melalui titik A. Berapa jarak dari bidang yang melalui titik A tersebut dengan bidang paralel  $6x - 8y - 2z - 5 = 0$ ? (keterangan:  $\sqrt{}$  = square root atau akar pangkat dua)
- $8 / \sqrt{104}$
  - $9 / \sqrt{104}$
  - $10 / \sqrt{104}$
  - $11 / \sqrt{104}$
  - $12 / \sqrt{104}$
  - Semua jawaban salah

### RUANG VEKTOR UMUM

22. Misalkan  $V = \{(a,b) | a, b \in \mathbb{R}\}$ . Pada V didefinisikan operasi penjumlahan dan perkalian skalar:  $(a,b) + (c,d) = (a+c, b+d)$  dan  $k(a,b) = (k.a, k.b)$ . Pilihlah satu jawaban yang benar diantara pilihan di bawah ini.
- V adalah ruang vektor real
  - V adalah bukan ruang vektor real**
  - V sebagian ruang vektor real dan sebagian lagi bukan ruang vektor real
  - Tidak bisa ditentukan secara pasti karena ada sebagian operasi tidak diketahui
  - Semua jawaban salah
  - Semua jawaban benar
23. Pilihlah jawaban yang berkaitan dengan matriks transisi
- Jika B<sub>1</sub> dan B<sub>2</sub> adalah basis untuk ruang vektor V, maka ada matriks transisi dari B<sub>1</sub> ke B<sub>2</sub>
  - Jika B adalah basis untuk ruang vektor R<sup>n</sup>, maka PB<sub>1</sub>->B adalah matriks identitas
  - Jika PB<sub>1</sub>->B<sub>2</sub> adalah matriks diagonal, maka setiap vektor dari B<sub>2</sub> adalah kelipatan dari B<sub>1</sub>
  - Hanya A dan B saja yang benar
  - Jawaban A, B, C semua benar**
  - Jawaban A, B, C semua salah
24. Diberikan sistem persamaan linier di bawah ini. Solusi SPL membentuk ruang vektor. Maka dapat \* disimpulkan bahwa:
- Basis ruang vektor adalah  $\{(4, 1, 0, 0), (-3, 0, 1, 0)\}$
  - Dimensi = 2
  - Basis ruang vektor  $\{(4, 0, 1, 1), (-3, 1, 0, 0)\}$
  - A dan B benar
  - B dan C benar
  - Hanya B yang benar, A dan C salah
  - Tidak ada yang benar**
25. Diketahui basis ruang vektor polinom  $B = \{(6+3x)(10+2x)\}$  dan  $B' = \{(2+x)(3+2x)\}$ , Jika  $p = -3-3x$ , maka [p]B dan matriks transisi dari B->B' adalah (Notasi matriks : {(baris1); {baris2}})
- (-2,1) dan  $\{\{-3\ 9\}; (0\ -3)\}$
  - (2,1) dan  $\{(3\ 9\); (0\ 9)\}$
  - (-2,1) dan  $\{\{3\ 9\}; (0\ -3)\}$
  - (2,1) dan  $\{(-3\ 9\); (0\ 3)\}$
  - Tidak ada jawaban yang benar**

# UTS ALGEO 2021/2022

## MATRIKS DAN DETERMINAN

1. Jika diketahui matriks A sebagai berikut, tentukan inversnya.

$$A = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix}$$

- A).  $\begin{pmatrix} -\sin \theta & \cos \theta \\ \cos \theta & \sin \theta \end{pmatrix}$   A
- B).  $\begin{pmatrix} \cos \theta & -\sin \theta \\ -\sin \theta & \cos \theta \end{pmatrix}$   B
- C).  $\begin{pmatrix} -\cos \theta & \sin \theta \\ -\sin \theta & -\cos \theta \end{pmatrix}$   C
- D).  $\begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix}$   D
- E). Tidak ada jawaban yang benar  E

2. Jika matrik I dan A berlaku hubungan sebagai berikut, maka matrik A yang mungkin adalah

$$(I + 2A)^{-1} = 1/9 \begin{bmatrix} -5 & 4 \\ 6 & -3 \end{bmatrix}$$

- A).  $\begin{bmatrix} 1 & 2 \\ 4 & 8 \end{bmatrix}$   A
- B).  $\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$   B
- C).  $\begin{bmatrix} 1 & 2 \\ 4 & 4 \end{bmatrix}$   C
- D).  $\begin{bmatrix} 1 & 2 \\ 3 & 2 \end{bmatrix}$   D
- E). Tidak ada jawaban yang benar  E

3. Perhatikan matriks B sebagai berikut. Maka dapat disimpulkan

$$B = \begin{bmatrix} -1 & 2 & 7 & 0 \\ 3 & 5 & -8 & 4 \\ 1 & 2 & 7 & -3 \\ 4 & -2 & 1 & 0 \end{bmatrix}$$

- A)  $\text{Trace}(B) = \text{Trace}(B^T)$  (Ket:  $B^T$  adalah matriks transpose dari B)
- B)  $\text{Trace}(B) = -\text{Trace}(B^T)$
- C)  $\text{Trace}(B) + \text{Trace}(B^T) = 22$
- D)  $\text{Trace}(B) + \text{Trace}(B^T) = 0$
- Jawaban A dan C benar
- Jawaban B dan D benar
- Semua jawaban di atas salah

4. Diketahui A adalah determinan dari matrik pertama dan B adalah determinan matrik kedua, maka berlaku

$$A = \begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix} \quad B = \begin{vmatrix} a_1 + b_1 & a_1 - b_1 & c_1 + b_1 \\ a_2 + b_2 & a_2 - b_2 & c_2 + b_2 \\ a_3 + b_3 & a_3 - b_3 & c_3 + b_3 \end{vmatrix}$$

- A.  $B = A$   
 B.  $B = -A$   
 C.  $B = -2A$   
 D.  $B = -3A$   
 E. Tidak ada jawaban yang benar

5. Diberikan matriks A sebagai berikut. Jika  $\det(A) = -5$  dan  $k = 3$ , maka jawaban yang SALAH adalah:

$$A = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

- A.  $\det(A^t) = -5$  (ket: t = transpose)  
 B.  $\det(A^{-1}) = -1/5$  (ket:  $A^{-1}$  = balikan atau invers A)  
 C.  $\det(3A) = -15$   
 D.  $\det(AI) = -5$  (ket: I = matriks identitas)  
 E.  $\det(A^2) = 25$  (ket:  $A^2 = A \times A$ )  
 F. Semua jawaban di atas tidak ada yang benar

### SISTEM PERSAMAAN LINEAR

6. Diberikan sistem persamaan linier (SPL) berikut. Solusi sistem persamaan linier tersebut bergantung pada nilai a

$$\begin{aligned} x + 2y + z &= 2 \\ 2x - 2y + 3z &= 1 \\ x + 2y - (a^2 - 3)z &= a \end{aligned}$$

- A) Jika  $a = -2$ , maka SPL tidak memiliki solusi  
 B) Jika  $a = 2$ , maka SPL memiliki banyak solusi  
 C) Jika a tidak sama dengan +2 maka SPL memiliki solusi unik (tunggal)  
 D) Untuk nilai a selain +2 dan -2, solusi SPL adalah  $z = 1/(a + 2)$   
 E) Semua jawaban A, B, C, dan D benar  
 F) Hanya B, C dan D yang benar  
 G) Hanya A, C, dan D yang benar  
 H) Hanya A, B, dan C yang benar  
 I) Semua jawaban salah

Pak Sukardi mempunyai uang Rp150.000,00 yang terdiri atas  $a$  lembar uang lima ribuan,  $b$  lembar uang sepuluh ribuan, dan  $c$  lembar uang dua puluh ribuan. Pak Sintan mempunyai uang Rp330.000,00 yang terdiri atas  $b$  lembar uang dua puluh ribuan dan  $c$  lembar uang lima puluh ribuan. Pak Ridwan mempunyai uang Rp600.000,00 yang terdiri atas  $a$  lembar uang lima puluh ribuan dan  $c$  lembar uang seratus ribuan. Jika Pak Akwila hanya mempunyai  $c$  lembar uang seratus ribuan, maka uang Pak Akwila sebanyak . . .

Tandai satu oval saja.

- Rp 400.000
- Rp. 500.000
- Rp. 700.000
- Rp. 1.000.000
- Rp. 1.500.000
- Semua jawaban salah

8.

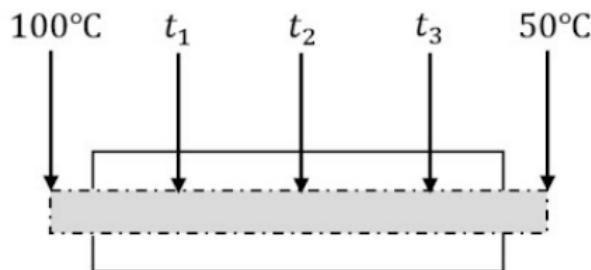
Tiga tukang cat bernama Joni, Deni, dan Ari biasanya bekerja secara bersama-sama. Mereka dapat mengecat eksterior (bagian luar) sebuah rumah dalam waktu 10 jam kerja. Deni dan Ari pernah bersama-sama mengecat rumah yang serupa dalam waktu 15 jam kerja. Suatu hari, ketiga tukang cat ini bekerja mengecat rumah serupa selama 4 jam kerja. Setelah itu, Ari pergi karena ada keperluan mendadak. Joni dan Doni memerlukan tambahan waktu 8 jam kerja lagi untuk menyelesaikan pengecatan rumah. Tentukan waktu yang dibutuhkan masing-masing tukang cat jika masing-masing bekerja sendirian.

Tandai satu oval saja.

- Joni 30 jam, Deni 24 jam, Ari 40 jam
- Joni 30 jam, Deni 40 jam, Ari 24 jam
- Joni 24 jam, Deni 30 jam, Ari 40 jam
- Joni 24 jam, Deni 40 jam, Ari 30 jam
- Joni 40 jam, Deni 30 jam, Ari 24 jam
- Joni 40 jam, Deni 24 jam, Ari 30 jam
- Semua Jawaban Salah

9.

Sebuah batang logam terisolasi dengan suhu pada masing-masing titik ditunjukkan oleh  $t_1$ ,  $t_2$ , dan  $t_3$  seperti tampak pada gambar.



Jika suhu pada titik-titik yang ditunjuk sama dengan rataan dua suhu di titik terdekat, tentukan:

Tandai satu oval saja.

- suhu di  $t_1$  adalah 80 derajat celcius
- suhu di  $t_1$  adalah 82,5 derajat celcius
- suhu di  $t_1$  adalah 85 derajat celcius
- suhu di  $t_1$  adalah 87,5 derajat celcius
- Semua jawaban salah

**VEKTOR DAN RUANG VEKTOR**

10. Dari keempat himpunan vektor di bawah ini, manakah himpunan yang ortogonal?

- (a)  $\mathbf{v}_1 = (2, 3), \mathbf{v}_2 = (-3, 2)$
- (b)  $\mathbf{v}_1 = (1, -2), \mathbf{v}_2 = (-2, 1)$
- (c)  $\mathbf{v}_1 = (1, 0, 1), \mathbf{v}_2 = (1, 1, 1), \mathbf{v}_3 = (-1, 0, 1)$
- (d)  $\mathbf{v}_1 = (2, -2, 1), \mathbf{v}_2 = (2, 1, -2), \mathbf{v}_3 = (1, 2, 2)$

Tandai satu oval saja.

- a dan b
- a dan c
- a dan d
- b dan c
- b dan d
- tidak ada jawaban yang benar

11. Jarak dari titik  $(3, 1, -2)$  ke bidang  $x + 2y - 2z = 4$  adalah

Tandai satu oval saja.

- $3/7$
- $4/5$
- $5/3$
- Opsi 4
- $5/11$
- Semua jawaban tidak ada yang benar

12. Jika diketahui tiga vektor  $u, v$  dan  $w$ , maka skalar triple product  $u \cdot (v \times w)$  menyatakan

- A. Vektor proyeksi dari  $u$  terhadap bidang  $v, w$
- B. Persamaan bidang yang sejajar bidang  $v, w$  dan mempunyai normal sejajar dengan  $u$
- C. Luas permukaan bidang dari bangun ruang yang dibentuk oleh tiga vektor tersebut
- D. Menyatakan volume bangun ruang yang dibentuk oleh tiga vektor tersebut
- E. Tidak ada jawaban yang benar

13..

Jika diberikan persamaan bidang P1:  $ax + by + cz + d = 0$ , dan P2:  $(x) = (x_0) + s(v_1) + t(v_2)$  maka pernyataan berikut yang benar adalah: catatan:  $(x)$  menyatakan vektor  $x^*$

Tandai satu oval saja.

- A. Persamaan P1 dan P2 tidak dapat diubah dari persamaan pertama ke kedua atau sebaliknya.
- B. Normal dari bidang P1 tidak dapat ditentukan dari persamaan bidang P2
- C. Bidang P1 dapat digambarkan dalam ruang dimensi 3, sedangkan P2 tidak dapat.
- D. Bidang P1 dan P2 sejajar jika vektor  $(a,b,c)$  sama dengan hasil kali silang  $v_1$  dengan  $v_2$
- E. Tidak ada jawaban yang benar

14. .

Jika diketahui tiga buah polinom  $p_1=2 + x + 4x^2$ ;  $p_2=1 - x + 3x^2$ ; dan  $p_3=3 + 2x + 5x^2$ , dan polinom  $q = 7 + 8x + 9x^2$  dapat dinyatakan sebagai kombinasi linier dari  $p_1, p_2$ , dan  $p_3$  maka  $k_1, k_2$ , dan  $k_3$  berturut-turut bernilai \*

Tandai satu oval saja.

- A. 0, -2, 3
- B. 0, 2, -3
- C. 2, 0, 3
- D. 3, -2, 0
- E. Tidak ada jawaban yang benar

## BASIS DAN DIMENSI

15. .

Diberikan empat buah vektor  $v_1=(1,0,1,1)$ ;  $v_2=(-3,3,7,-1)$ ;  $v_3=(-1,3,9,3)$ ;  $v_4=(-5,3,5,-1)$ . Diantara vektor-vektor tersebut yang merupakan basis dari sub-ruang vektor yang di-span adalah: \*

Tandai satu oval saja.

- A. V1 dan V2
- B. V1 dan V3
- C. V1, V2, dan V3
- D. V1,V2,V3, dan V4
- E. Tidak ada jawaban yang benar

16. .

Misalkan  $B_1 = \{u_1, u_2\}$  dan  $B_2 = \{v_1, v_2\}$  adalah basis-basis untuk ruang vektor  $R^2$ , yang dalam hal ini  $u_1 = (1, 2)$ ,  $u_2 = (2, 3)$ ,  $v_1 = (1, 3)$  dan  $v_2 = (1, 4)$ . Matriks transisi dari basis dari  $B_1$  ke  $B_2$  sudah dihitung seperti dibawah ini, maka: \*

$$P_{B_1 \rightarrow B_2} = \begin{bmatrix} 2 & 5 \\ -1 & -3 \end{bmatrix}$$

Tandai satu oval saja.

$$P_{B_2 \rightarrow B_1} = \begin{bmatrix} 3 & 5 \\ -1 & -2 \end{bmatrix}$$

\*Soal terpotong, pilihan tidak terlihat sampai akhir\*

17. Sebuah matriks A berukuran  $4 \times 4$ . Jika A memiliki balikan (invers), maka dapat disimpulkan bahwa:

- A)  $\text{Nullity}(A) = 4$ ,  $\text{Rank}(A) = 0$
- B)  $Ax = 0$  memiliki solusi trivial
- C)  $\det(A)$  tidak sama dengan 0
- D) Matriks eselon baris tereduksi dari A adalah matriks identitas I berukuran  $4 \times 4$
- E) Semua jawaban di atas benar
- F) Hanya B dan C yang benar
- G) Hanya jawaban B, C, dan D yang benar
- Semua jawaban di atas tidak ada yang benar

18. .

Sebuah matriks A berukuran  $4 \times 6$ . Jumlah parameter (seperti r, s, t, dst) di dalam solusi umum SPL homogen  $Ax = 0$  adalah 4 buah, maka dapat disimpulkan \*

Tandai satu oval saja.

- dimensi ruang baris = 4
- dimensi ruang kolom = 4
- jumlah baris tidak nol = 4
- $\text{rank}(A) = 4$
- jumlah variabel non-parameter pada solusi umum  $Ax = 0$  adalah 4
- semua jawaban di atas benar
- semua jawaban di atas salah

#### TRANSFORMASI LINEAR

19. Jika  $T: R^2 \rightarrow R^3$  transformasi linier yang didefinisikan sbb:  $T(x_1, x_2) = (x_1+3x_2, x_1-x_2, x_1)$ , maka rank dan nullity dari matrik T adalah

- A. 2, 1
- B. 3, 0
- C. 1, 2
- D. 2, 0
- E. Tidak ada jawaban yang benar

20. Sebuah vektor  $v = (2, -1)$  di  $R^2$  mula-mula dicerminkan terhadap  $y = x$ , selanjutnya hasilnya diregang (shear) dalam arah x dengan faktor  $k = 3$ . Maka, bayangan akhir v adalah

- $v' = (2, -5)$
- $v' = (5, 2)$
- $v' = (2, 5)$
- $v' = (5, -2)$
- $v' = (-5, 2)$
- $v' = (-2, 5)$
- Semua jawaban di atas tidak ada yang benar

21. .

Misalkan  $T$  merupakan suatu transformasi dari  $M_{2 \times 2}$  ke  $R$  yang didefinisikan oleh  $T(A) = \det(A)$ , untuk setiap  $A \in M_{2 \times 2}$ . Apakah  $T$  merupakan Transformasi Linear?

Tandai satu oval saja.

- T adalah bukan Transformasi linear  
 T adalah Transformasi Linear  
 Tidak bisa ditentukan  
 Semua jawaban salah

22. .

Diketahui basis dari polinom orde dua adalah  $\{1+x, -x+x^2, 1+x-x^2\}$

Jika  $T: P_2 \rightarrow R^3$  adalah transformasi linear dimana

$$T(1+x) = \begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix}; T(-x+x^2) = \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix}; T(1+x-x^2) = \begin{pmatrix} 2 \\ 1 \\ 0 \end{pmatrix}$$

Tentukan

$$T(1-x+x^2)$$

Tandai satu oval saja.

- Transpose dari (0,4,3)  
 Transpose dari (1,2,3)  
 Transpose dari (4,5,0)  
 Transpose dari (3,4,0)  
 Semua jawaban salah

23. .

Diketahui  $T: P_2$  (Polinom orde 2)  $\rightarrow R^2$ , dimana

$$T(a+bx+cx^2) = \begin{pmatrix} a-b \\ a-c \end{pmatrix}$$

Tandai satu oval saja.

- T adalah bukan Transformasi Linear  
 T adalah Transformasi linear  
 Tidak bisa ditentukan  
 Semua jawaban salah

24. .

Diketahui transformasi linear  $T: R^4 \rightarrow R^3$  didefinisikan oleh:

$$T \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} = \begin{pmatrix} a+b \\ c-2d \\ -a-b+c-2d \end{pmatrix}$$

Tandai satu oval saja.

- nulitasnya adalah 2
- nulitasnya adalah 1
- nulitasnya adalah 3
- nulitasnya adalah 4
- nulitasnya adalah 5
- Tidak ada jawaban yang benar

25. .

Misalkan

$$\left\{ \overrightarrow{v_1} = \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}, \overrightarrow{v_2} = \begin{pmatrix} 0 \\ 1 \\ -1 \end{pmatrix}, \overrightarrow{v_3} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right\} \text{ adalah basis bagi } R^3.$$

$T: R^3 \rightarrow P_1$  Transformasi linear didefinisikan  $T(\overrightarrow{v_i}) = A\overrightarrow{v_i} = p_i$  untuk setiap  $i = 1, 2, 3$ .

Jika

$$p_1 = 1 - x; p_2 = 1; p_3 = 2x$$

Tentukan hasil transformasi dari  $T \begin{pmatrix} 1 \\ -1 \\ 2 \end{pmatrix}$

Tandai satu oval saja.

- $1 + x$
- $-1 + x$
- $1 - x$
- $-1 - x$
- Semua jawaban salah

## UAS ALJABAR LINEAR DAN GEOMETRI

UAS SEMESTER PENDEK 2006/2007

3 Agustus 2007

120 menit

1. Misalkan  $T : R^3 \rightarrow R^3$  pemetaan linear didefinisikan sebagai berikut

$$T(x,y,z) = \begin{vmatrix} x - y + 2z \\ 2x + y \\ -x - 2y + 2z \end{vmatrix}$$

- a. Tentukan matriks penyajian  $T$  terhadap basis baku dari  $R^3$
  - b. Tentukan basis bagi ruang Peta( $T$ ) dan basis bagi ruang Inti ( $T$ )
2. Misalkan

$$A = \begin{vmatrix} 0 & 0 \\ 3 & 3 \\ 0 & 0 \end{vmatrix}$$

Apakah  $A$  dapat didiagonalkan secara ortogonal? Jelaskan jawaban saudara!

Jika  $A$  dapat didiagonalkan secara ortogonal, tentukan matriks tak singular  $P$  yang mendiagonalkan matriks  $A$  secara ortogonal

3. Misalkan  $V_1 = (\lambda, -1, -1)$ ,  $V_2 = (-1, \lambda, -1)$ ,  $V_3 = (-1, -1, \lambda)$  vektor-vektor di ruang  $R^3$ . Tentukan semua

bilangan riil  $\lambda$  supaya  $(V_1, V_2, V_3)$  bergantung linier

4. Misalkan  $B = (b_1, b_2, b_3)$  dan  $C = (c_1, c_2, c_3)$  adalah dua buah basis dari suatu ruang vektor  $V$ . Misalkan matriks transisi perubahan basis dari basis  $B$  ke basis  $C$  adalah

$$\begin{pmatrix} 2 & -1 & 3 \\ 1 & 1 & 4 \\ 0 & 1 & 2 \end{pmatrix}$$

- a. Tuliskan vektor-vektor di  $C$  sebagai kombinasi linier vektor-vektor di  $B$
- b. Tuliskan vektor-vektor di  $B$  sebagai kombinasi linier vektor-vektor di  $C$

## UAS SEMESTER II – 2007/2008

8 Januari 2008

120 menit

1. Diberikan  $T : R^3 \rightarrow R^3$  pemetaan linear:

$$T(x, y, z) = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 4 & 6 \\ -1 & -2 & -3 \end{pmatrix}$$

- a. Tentukan Inti  $T$  berikut basisnya
- b. Tentukan Peta  $T$  berikut basisnya
- c. Periksa apakah  $T$  bersifat satu-satu (Jelaskan jawaban Anda)

2. Misalkan  $A = \begin{pmatrix} 1 & -2 & -4 \\ 0 & 2 & 2 \\ 0 & -2 & -3 \end{pmatrix}$

- a. Tentukan Ruang karakteristik (Ruang eigen) dari  $A$  berikut basisnya
- b. Tentukan matriks  $P$  yang mempunyai invers sehingga  $P^{-1}AP$  diagonal
- c. Tentukan matriks  $P^{-1}AP$

3. Misalkan  $B = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}$ . Tentukan himpunan semua  $x$  di  $R^2$  sehingga  $Bx$  sejajar dengan  $x$ .

4. Misalkan  $C = \begin{pmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{pmatrix}$  matriks penyajian dari pemetaan linier  $F : R^3 \rightarrow R^3$  terhadap basis  $X = \left\{ \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \right\}$

dan  $Y = \left\{ \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \right\}$

- a. Tentukan  $\left[ T \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \right]_y$  dan  $\left[ T \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \right]_x$

- b. Tentukan  $T \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$  dan  $T \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$

- c. Tentukan pemetaan linear  $F(x, y)$  untuk setiap  $(x, y)$  di  $R^2$ .

## UAS SEMESTER II – 2008/2009

17 Desember 2008

120 menit

1. Tentukan semua jawab dari

$$\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 2 & 3 & 1 \\ -1 & 4 & 5 & 2 \\ -2 & 2 & 1 & 1 \end{bmatrix} \begin{bmatrix} r \\ s \\ t \\ u \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

2.  $W$  merupakan ruang bagian dari  $R^3$  yang dibangun oleh  $S = \{(0,-2,1), (-1,5,-3), (1,1,0)\}$ . Tentukan basis dari  $W$ .
3. Sebuah partikel bergerak sepanjang garis lurus dari titik  $(1,2,-3)$  ke  $(-3,-2,1)$  selama satu satuan waktu.
- Tentukan vektor kecepatan dari gerakan partikel
  - Tentukan koordinat (posisi) partikel pada setiap saat  $t$ , kemudian tentukan koordinatnya pada saat  $t = \frac{3}{4}$
4. Hitung  $\int_C \sqrt{x} - \frac{y}{3x} ds$  jika  $C : \vec{r}(t) = t^2 \vec{i} - 3t \vec{j}$ ,  $0 \leq t \leq 2$ , dan  $ds$  menyatakan elemen panjang kurva.
5. Diberikan  $\int_C xy dx - (x^2 - y^2) dy$  dengan  $C$  lintasan tertutup berorientasi positif yang merupakan batas segitiga dengan titik sudut  $(0,0), (1,0), (1,1)$ .
- Nyatakan integral di atas sebagai integral lipat.
  - Hitung integral di atas.

## UAS SEMESTER I – 2013/2014

Sekolah Teknik Elektro Informatika  
Institut Teknologi Bandung

**Ujian Akhir Semester I 2013/2014**  
**Mata kuliah : Aljabar Geometri (IF2123)**  
**Hari/tanggal : Senin, 16 Desember 2013**  
**Waktu : 120 menit**

I. Diberikan dua bilangan kompleks  $z_1 = 1 + \sqrt{3}i$  dan  $z_2 = \sqrt{3} + i$ , hitunglah:

$$1). z_1 z_2 = 4i \quad z_1 = 2e^{i\pi/3} \quad z_2 = 2e^{i\pi/6}$$

$$2). \frac{z_1}{z_2} = \frac{1}{2}i + \frac{1}{2}\sqrt{3}$$

$$3). z_1^8 = -128 + 128\sqrt{3}i$$

$$4). \ln z_2 = \ln 2 + i \cdot \frac{\pi}{6}$$

II. Diberikan dua quaternion  $q_1 = 1 + 2i + 3j + 4k$  dan  $q_2 = 1 - 2i + 3j - 2k$ , hitunglah:

$$1). q_1 + q_2 = 2 + \begin{bmatrix} 0 \\ 2 \\ 3 \\ 4 \end{bmatrix}$$

$$2). q_1 q_2 = 4 + \begin{bmatrix} -18 \\ 18 \\ 18 \\ 14 \end{bmatrix}$$

$$3). q_1 q_2^{-1} = \frac{1}{9} (-1 + \begin{bmatrix} 11 \\ -2 \\ -3 \end{bmatrix})$$

III. Diketahui tiga vektor :

$$\mathbf{a} = e_1 + 2e_2 - e_3$$

$$\mathbf{b} = 2e_1 - e_2 + 2e_3$$

$$\mathbf{c} = -e_1 + e_2 + e_3$$

$$\mathbf{b} \wedge \mathbf{c} = e_{12} + 4e_{13} - 3e_{23}$$

$$\mathbf{a} \wedge \mathbf{b} = -5e_{12} + 4e_{13} + 3e_{23}$$

Hitunglah :

$$1). \mathbf{a}(\mathbf{b} \wedge \mathbf{c}) = e_1 + e_2 - 2e_3 - 15e_{123}$$

$$2). (\mathbf{a} \wedge \mathbf{b})(\mathbf{b} \wedge \mathbf{c}) = 2 + 24e_{12} + 12e_{13} + 24e_{23}$$

## UAS SEMESTER I – 2014/2015

Sekolah Teknik Elektro Informatika  
Institut Teknologi Bandung

**Ujian Akhir Semester I 2014/2015**  
**Mata kuliah : Aljabar Geometri (IF2123)**  
**Hari/tanggal : Selasa, 09 Desember 2014**  
**Waktu : 100 menit**

- I. Diberikan dua quaternion  $q_1 = 2 + i + 2j + ik$  dan  $q_2 = 1 - 2i - j + 2k$ ,  
**Hitunglah :**

$$\begin{aligned} 1). q_1 - q_2 &\rightarrow 1+3i+3j-k \\ 2). q_1 q_2 &\rightarrow 4+2i-4j+8k \\ 3). q_1 q_2^{-1} &\rightarrow \frac{4}{5}i - \frac{3}{5}j - \frac{3}{5}k \end{aligned}$$

- II. Diketahui tiga vektor dan satu bivektor:

$$\begin{aligned} \mathbf{a} &= 2e_1 + e_2 + 2e_3 \\ \mathbf{b} &= e_1 + e_2 - 3e_3 \\ \mathbf{c} &= e_1 + 2e_2 + 2e_3 \\ B &= (\mathbf{b} \wedge \mathbf{c}) \end{aligned}$$

$$\begin{aligned} \mathbf{e}_1 \wedge \mathbf{e}_2 &= \cancel{\mathbf{e}_1} \\ \mathbf{e}_2 \wedge \mathbf{e}_3 &= \cancel{\mathbf{e}_2} \\ \mathbf{e}_1 \wedge \mathbf{e}_3 &= \cancel{\mathbf{e}_1} \\ \mathbf{e}_1 \wedge \mathbf{e}_2 + \mathbf{e}_1 \wedge \mathbf{e}_3 &= \mathbf{e}_1 \wedge \mathbf{e}_2 + \cancel{\mathbf{e}_1} \\ &= \mathbf{e}_1 \wedge \mathbf{e}_2 \\ &= \mathbf{e}_{12} \end{aligned}$$

Tentukan :

- $$\begin{array}{ll} 1). aB & 2). (a \cdot B) \\ 3). IA & 4). abc \end{array}$$

- III. Diketahui tiga vektor :

$$\begin{aligned} \mathbf{a} &= e_1 + 2e_2 + 2e_3 \\ \mathbf{b} &= 2e_1 + e_2 + e_3 \\ \mathbf{c} &= 2e_1 + 2e_2 + 2e_3 \end{aligned}$$

-13

$\mathbf{e}_{12}$

**Hitunglah :**

- 1). Luas parallelogram yang dibentuk oleh vektor  $\mathbf{a}$  dan  $\mathbf{b}$
- 2). Volume parallel pipedum yang dibentuk oleh 3 vektor diatas

$$\begin{aligned} & -(2i+j-2k) \\ & \times (e_1: e_2: e_3) \\ & = (2i+j-2k) \times (e_1: e_2: e_3) \end{aligned}$$

**UAS SEMESTER I – 2015/2016**

IF2123 Aljabar  
Geometri Rabu, 16  
Desember 2015

Waktu 100 menit

- I. Diketahui sebuah titik  $P = (1,1,1)$  diputar terhadap sumbu  $u = j + k$  sebesar  $180^\circ$ , tentukan koordinat titik  $P'$  yang merupakan hasil dari rotasi tersebut.

- II. Diberikan tiga vektor dan satu buah bivektor berikut:  $a = 2e_1 + e_2 + 2e_3$   
 $b = e_1 + e_2 - 3e_3$   
 $c = e_1 + 2e_2 + 2e_3$   
 $B = (b \wedge c)$

Hitunglah:

1.  $\frac{a}{B}$
2.  $\frac{ab}{c}$
3.  $(a \cdot B) \cdot (B \cdot a)$
4.  $I^3$

- III. Diketahui tiga vektor:

$$\begin{aligned}a &= e_1 + e_2 - 2e_3 \\b &= 2e_1 - e_2 + 3e_3 \\c &= 2e_1 + 2e_2 - e_3\end{aligned}$$

Tentukan:

1. Luas bayangan yang merupakan proyeksi dari bidang yang dibentuk oleh vektor  $a$  dan vektor  $b$  pada bidang  $(e_1 \wedge e_2)$
2. Perpotongan bidang yang dibentuk oleh vektor  $a$  dan vektor  $b$  dengan bidang  $(e_1 \wedge e_2)$
3. Volume parallel pipedum yang dibentuk oleh vektor  $a$ ,  $b$ , dan  $c$ .

Nilai: 20-40-40 (15;15;10)

Nilai SALAH 0; BENAR penuh

**UAS SEMESTER I – 2016/2017**  
**Mata kuliah:** Aljabar Geometri (IF2123)  
**Hari/tanggal:** Selasa, 13 Desember 2016  
**Waktu:** 120 menit  
**Sifat:** tutup buku

1. Diberikan dua vektor  $a = 2e_1 - 3e_3$ ,  $b = 3e_2 - 4e_3$ , dan bivektor  $B = e_{12}$ . Hitunglah
  - a.  $A = a \wedge b$
  - b.  $a \cdot B$
  - c.  $a \wedge B$
  - d.  $A \vee B$
2. Diberikan vektor  $b = 2e_1 + 3e_2 - e_3$ , dan bivektor  $B = 4e_{23} + e_{31} - 5e_{12}$ .
  - a. Tunjukkan bahwa  $b^{-1} = \frac{b}{\|b\|^2}$
  - b. Gunakan hasil di atas untuk menghitung vektor  $a$ , jika  $B = ab$ .
3. Jika diberikan vektor  $m = e_1 - e_3$ ,  $n = e_2 - e_3$ , dan  $a = 2e_1 - 2e_2 + e_3$ 
  - a. Tentukan bayangan  $a''$ , jika direfleksikan terhadap  $\hat{m}$  kemudian dilanjutkan terhadap  $\hat{n}$ .
  - b. Tentukan matriks rotasi  $[R]$ , kemudian gunakan untuk menghitung bayangan  $a''$ .
4. Diketahui persamaan bidang  $w: 2x - 3y + z = 2$  dan titik  $P(1, -2, 4)$ .
  - a. Hitunglah jarak titik  $P$  ke bidang tersebut
  - b. Jika garis  $\vec{x} = (3, 3, 4)^T + \lambda(2, -1, 1)^T$  memotong bidang tersebut, tentukan titik potongnya.

UAS SEMESTER I – 2017/2018  
**Mata Kuliah :** Aljabar Geometri (IF2123)  
**Hari/tanggal :** Rabu, 6 Desember 2017  
**Waktu:** 100 menit

---

I. Transformasi  $T : \mathbb{R}^4 \rightarrow \mathbb{R}^3$  didefinisikan :

$$T \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 7x_1 + 2x_2 - x_3 + x_4 \\ x_2 + x_3 \\ -x_1 + 2x_3 \end{pmatrix}$$

- a) Tentukan matriks transformasi  $T$ . (perlihatkan cara perhitungan dengan menggunakan vector basis satuan).
- b) Dengan menggunakan jawaban a), tentukan bayangan vector  $(3, -1, 4, 5)$ .
- c) Jika hasil dari langkah b) diregang (shear) dalam arah  $x$ , tentukan bayangan akhirnya.

II. Diberikan matriks:

$$A = \begin{bmatrix} 5 & 6 & 2 \\ 0 & -1 & -8 \\ 1 & 0 & -2 \end{bmatrix}$$

- a). Tentukan nilai eigen dari matriks A.
- b). Tentukan vektor-vektor eigen dari (1).
- c). Tentukan matriks  $A^2$ , (pangkat dua matriks A)
- d). Tentukan nilai eigen dari matriks  $A^2$ , apakah nilai eigen matriks  $A^2$  = nilai eigen (a) dikalikan?

III. Diberikan dua quaternion  $p = 2 + 2i + 3j + 4k$  dan  $q = 3 - i + 5j - 2k$ , Hitunglah :

- 1).  $pq$
- 2).  $P^{-1}$
- 3).  $P^{-1}p$ .

IV. Diberikan tiga buah vektor

$$\begin{aligned} \mathbf{a} &= 2e_1 + e_2 + e_3 \\ \mathbf{b} &= 3e_1 + 5e_2 - 2e_3 \\ \mathbf{c} &= -e_1 + 2e_2 - e_3 \end{aligned}$$

*Hitunglah :*

- 1).  $\mathbf{a} \wedge \mathbf{b}$
- 2).  $\|\mathbf{a} \wedge \mathbf{b}\|$
- 3).  $\mathbf{a}(\mathbf{bc})$

V. Diketahui tiga buah vektor :

$$\mathbf{a} = e_1 + e_2 - 2e_3; \quad \mathbf{b} = e_1 - e_2 + 2e_3; \quad \mathbf{c} = 2e_1 + e_2 - 2e_3$$

1. Tentukan luas bayangan yang merupakan proyeksi dari bidang yang dibentuk oleh vektor  $\mathbf{a}$  dan vektor  $\mathbf{b}$  pada bidang  $(e_1 \wedge e_2)$
2. Hitunglah volume parallelepipedum yang dibentuk oleh vektor  $\mathbf{a}$ ,  $\mathbf{b}$ , dan  $\mathbf{c}$ .

VI. Diketahui vektor  $v = e_1 + 2e_2$ .

1. Tuliskan rotor untuk memutar vektor tersebut sebesar  $\emptyset$
2. Dengan menggunakan rotor tersebut, tentukan bayangan vektor  $v$  ( $v'$ ) jika diputar sebesar  $30^\circ$  berlawanan arah jarum jam.
3. Tentukan bayangan vector  $v$  ( $v'$ ), jika diputar  $45^\circ$  searah dengan jarum jam.

UAS SEMESTER I – 2019/2020  
Mata Kuliah : Aljabar Geometri (IF2123)  
Hari/tanggal : Rabu, 11 Desember 2019  
Waktu: 120 menit

---

I. Ada banyak cara melakukan rotasi sebuah vektor dengan menggunakan rotor. Misalkan vektor  $\mathbf{v} = 2\mathbf{i} - 4\mathbf{j} + 5\mathbf{k}$  akan diputar sejauh  $\theta$  berlawanan arah jarum jam, tentukan  $\mathbf{v}'$  (bayangan  $\mathbf{v}$ ).

- Jika  $\mathbf{v}$  diputar dengan rotor quaternion pada sumbu  $\mathbf{u} = \mathbf{i} + \mathbf{j} + \mathbf{k}$  sejauh  $\theta = 120^\circ$
- Jika  $\mathbf{v}$  diputar dengan menggunakan pseudoscalar sejauh  $\theta = 120^\circ$ .

II. Diberikan tiga buah vektor sebagai berikut:

$$\begin{aligned}\mathbf{a} &= 2\mathbf{e}_1 + \mathbf{e}_2 - \mathbf{e}_3 \\ \mathbf{b} &= \mathbf{e}_1 - \mathbf{e}_2 - \mathbf{e}_3 \\ \mathbf{c} &= 2\mathbf{e}_1 + 2\mathbf{e}_2 - \mathbf{e}_3\end{aligned}$$

- Jika  $B$  adalah multivektor,  $B = ab$ , perlihatkan bahwa  $a = Bb^{-1}$
- Tentukan perpotongan bidang yang dibentuk oleh vektor  $\mathbf{b}$  dan  $\mathbf{c}$  dengan bidang  $(\mathbf{e}_2 \wedge \mathbf{e}_3)$

III. Jika diketahui tiga buah vektor:

$$\begin{aligned}\mathbf{a} &= 2\mathbf{e}_1 + 2\mathbf{e}_2 + \mathbf{e}_3 \\ \mathbf{b} &= 3\mathbf{e}_1 + 2\mathbf{e}_2 - 2\mathbf{e}_3 \\ \mathbf{c} &= \mathbf{e}_1 + 2\mathbf{e}_2 - \mathbf{e}_3\end{aligned}$$

Hitunglah :

- $(\mathbf{a} + \mathbf{b})\mathbf{c}$
- $(\mathbf{a} \wedge \mathbf{b})\mathbf{c}$
- $(\mathbf{a} + \mathbf{b}) \cdot \mathbf{c}$

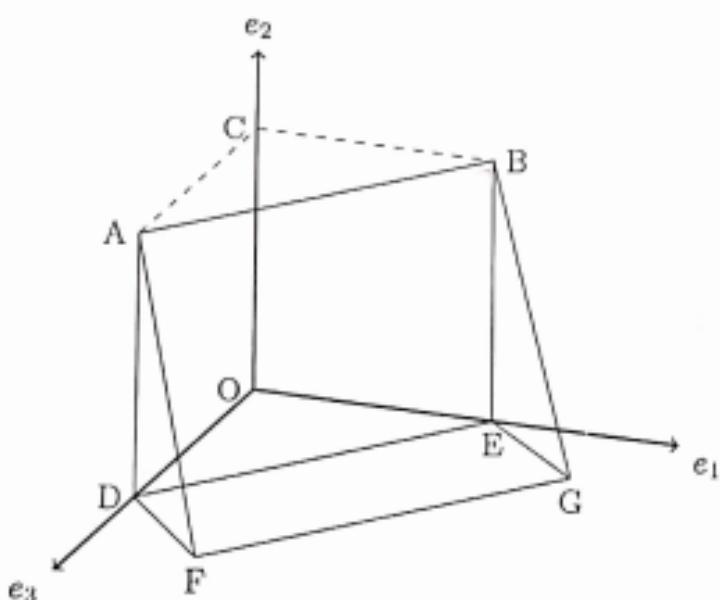
IV. Untuk tiga buah vektor yang sama seperti no III, hitunglah

- Luas jajaran genjang yang dibentuk oleh vektor  $\mathbf{a}$  dan  $\mathbf{b}$
- Luas bayangan jajaran genjang  $(\mathbf{a})$  pada bidang  $(\mathbf{e}_1 \wedge \mathbf{e}_2)$
- Volume parallelepiped yang dibentuk oleh ketiga vektor tersebut.

V. Jika  $I_n = e_{123\dots n}$ , adalah pseudoscalar di  $\mathbb{R}^n$ , tuliskan ekspresi berikut dalam bentuk yang paling sederhana:

- 1).  $I_1 I_2 I_3$
- 2).  $e_1 I_2 I_3 I_4 I_5$
- 3).  $(I_3)^4 (I_2)^2 I_3 I_2$

VI. Diberikan gambar bidang-bidang di  $\mathbb{R}^3$  sebagai berikut:



Jika bidang  $\overrightarrow{OE} \wedge \overrightarrow{OC} = e_1 \wedge e_2$ , bidang  $\overrightarrow{OC} \wedge \overrightarrow{OD} = e_2 \wedge e_3$ , bidang  $\overrightarrow{OD} \wedge \overrightarrow{OE} = e_3 \wedge e_1$ , dan luas bidang  $\overrightarrow{DF} \wedge \overrightarrow{DE}$  (dimana semua sudutnya siku-siku) sama dengan luas bidang  $e_3 \wedge e_1$ , nyatakan bidang-bidang berikut dalam ekspresi  $e_1, e_2$ , dan  $e_3$

- 1). Bidang  $\overrightarrow{AD} \wedge \overrightarrow{AB}$
- 2). Bidang  $\overrightarrow{DF} \wedge \overrightarrow{DE}$
- 3). Bidang  $\overrightarrow{AF} \wedge \overrightarrow{AB}$

## UAS ALGEO 2020/2021

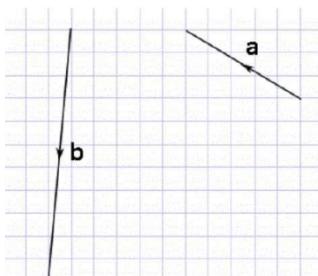
### VEKTOR DI RUANG UMUM

1. Diberikan matriks A seperti di bawah ini. Maka, dapat disimpulkan bahwa

$$\begin{bmatrix} 1 & 4 & 5 & 6 & 9 \\ 3 & -2 & 1 & 4 & -1 \\ -1 & 0 & -1 & -2 & -1 \\ 2 & 3 & 5 & 7 & 8 \end{bmatrix}$$

- a. rank(A) = 3, nullity(A)=1
- b. rank(A) = 2, nullity(A)= 2
- c. rank(A) = 1, nullity(A)= 3
- d. rank(A) = 4, nullity(A)= 0
- e. rank(A) = 0, nullity(A)= 4
- f. Semua jawaban di atas tidak ada yang benar

2. Diketahui dua buah vektor a dan b seperti pada gambar berikut. Berapa magnitude dari  $a + b$  ?



- a. 10
- b. 14,66
- c. 15,21
- d. 16,88
- e. 15
- f. 14,78

3. Diketahui suatu matrik Z berukuran  $p \times q$ . Pernyataan berikut yang benar adalah:

- a. Basis untuk ruang kolom dari matrik Z adalah semua kolom pada matrik Z.
- b. Basis untuk ruang baris dari matrik Z adalah semua vektor baris yang mengandung 1-utama pada matrik Z.
- c. Basis untuk ruang kolom dari matrik Z adalah semua vektor kolom dari matrik Z yang mengandung 1-utama
- d. Basis untuk ruang null adalah vektor-vektor yang membangun solusi SPL homogen  $Zx = 0$ .**
- e. Basis ruang kolom dan basis ruang baris matrik Z adalah sama

4. Vektor  $v = (4, -6)$  mula-mula diregang (shear) dengan faktor sebesar 3 dalam arah-x kemudian hasilnya dicerminkan terhadap sumbu  $y = x$

- a. Vektor bayangan akhirnya adalah  $V = (6, -14)$ (B) Matriks standard untuk peregangan lalu diikuti pencerminan memiliki determinan = -1
- b. Matriks standard untuk peregangan lalu diikuti pencerminan memiliki determinan = -1**
- c. Vektor bayangan oleh peregangan (shear) saja adalah  $V = (-14, 6)$
- d. Jawaban A, B, dan C benar
- e. Hanya jawaban A dan C saja yang benar
- f. Semua jawaban di atas tidak ada yang benar

5. Jika diketahui sistem linear  $Ax=b$ , dimana A adalah ukurannya  $3 \times 3$ , dan  $\text{rank}(A) = 3$ , dan  $\text{rank}[A|b] = 3$  , maka pernyataan yang benar di bawah ini adalah :

- a. Sistem Persamaan Linear (SPL) di atas tidak konsisten
- b. SPL di atas konsisten dan jumlah parameter di solusi umumnya adalah 0**
- c. SPL di atas konsisten dan jumlah parameter di solusi umumnya adalah 1
- d. SPL di atas konsisten dan jumlah parameter di solusi umumnya adalah 2
- e. SPL di atas konsisten dan jumlah parameter di solusi umumnya adalah 3
- f. Semua jawaban di atas tidak ada yang benar

6. Jika vektor-vektor baris dan vektor-vektor kolom matriks A adalah linearly independent , maka matriks A pastilah square matrix. Pernyataan ini adalah :

- a. Selalu salah
- b. Selalu benar**
- c. Kadang benar kadang salah
- d. Semua jawaban A,B,C benar
- e. Semua jawaban A,B,C salah
- f. Semua jawaban di atas tidak ada yang benar

**NILAI DAN VEKTOR EIGEN**

7. Diberikan sebuah matriks A di bawah ini. Setelah dihitung nilai eigen dan vektor eigennya, maka dapat disimpulkan bahwa:

$$A = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}$$

- a. Salah satu nilai eigennya adalah 2
- b. Nilai eigen yang lain adalah -1
- c. Salah satu vektor eigennya adalah  $(-1, -2)$
- d. Vektor eigen yang lainnya adalah  $(-1, -1)$
- e. Semua jawaban dari A sampai D benar
- f. Semua jawaban dari A sampai D tidak ada yang benar**

8. Misalkan A adalah matriks persegi  $n \times n$ , E adalah matriks yang mendiagonalisasi A, D adalah matriks diagonal yang memiliki kemiripan dengan A, maka pernyataan yang SALAH adalah

- a. A memiliki balikan (invers) jika dan hanya jika D memiliki balikan
- b.  $\text{Rank}(D) + \text{Nullity}(D) = n$
- c. Jika A tidak dapat didiagonalisasi, maka A tidak memiliki balikan (inverse)**
- d. Matriks A dan D memiliki trace yang sama
- e. Matriks A dan D selalu memiliki persamaan karakteristik yang sama
- f.  $A = EDE^{-1}$  (ket: Tanda A menyatakan perpangkatan)
- g. Tidak ada jawaban yang memenuhi

9. Diberikan matriks A di bawah ini. Nilai-nilai eigennya adalah

$$A = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 4 & 5 \\ 0 & 4 & 3 \end{bmatrix}$$

- a. 1, -2, 8
- b. -1, 2, -8
- c. 1, 2, -8
- d. -1, 2, 8**
- e. -1, -2, 8
- f. 1, -2, -8

10. Jika diketahui sebuah matrik persegi-Z berorde 3, memiliki nilai eigen a, b, dan c. Maka berapa nilai eigen dari matriks persegi-Y yang merupakan  $Z^T$  (Z-transpose)? \*

- a. a, b, c**
- b. -a, -b, -c
- c. a-b, b-a, c-a
- d. a/D, b/D, c/D; dengan D=  $\sqrt{a^2 + b^2 + c^2}$
- e. 1/a, 1/b, 1/c

11. Berapakah jumlah semua nilai eigen dari matriks

$$A = \begin{bmatrix} 3 & 6 & 7 \\ 5 & 4 & 2 \\ 7 & 9 & 1 \end{bmatrix}$$

- a. 6
- b. 7
- c. 8**
- d. 9
- e. 10

12. Jika A diagonalizable, maka ada matriks P yang unik dimana perkalian di bawah ini adalah menghasilkan matriks diagonal

$$P^{-1}AP$$

- a. Pernyataan di atas adalah salah**
- b. Pernyataan di atas adalah benar
- c. Tidak bisa ditentukan benar atau salahnya O
- d. Pernyataan di atas semua benar

13. Jika A diagonalizable, maka inverse dari matriks A juga pastilah diagonalizable

- a. Pernyataan di atas adalah salah
- b. Pernyataan di atas adalah benar**
- c. Tidak bisa ditentukan benar atau salahnya
- d. Pernyataan di atas semua benar

**ALJABAR QUATERNION**

14. Diberikan dua buah quaternion murni, yaitu  $a = 2i + 4j - k$  dan  $b = 3i - j + 3k$ . Maka dapat disimpulkan bahwa
- Perkalian dua buah quaternion murni tidak bersifat tertutup
  - Norma  $ab$  adalah  $\sqrt{399}$  (ket:  $\sqrt{\text{square root}} = \text{akar pangkat dua}$ )
  - Balikan (inverse)  $a$  adalah  $a' = (-2/21)i + (4/21)j + (1/21)k$
  - Norma( $a$ )  $\times$  Norma( $b$ ) = Norma( $ab$ )
  - Hanya jawaban C yang salah, jawaban lainnya benar
  - Hanya jawaban A dan C yang salah, jawaban lainnya benar**
  - Semua jawaban salah
15. Sebuah vektor  $p = 4i + 2j + 2k$  diputar berlawanan arah jarum jam sejauh  $a = 120$  derajat dengan sumbu rotasinya adalah  $u = i + j + k$ . Tentukan vektor bayangannya:
- $5i + j - 4k$
  - $5i - j + 4k$
  - $2i + 4i - 2j$
  - $-2 + 4i + 2j$
  - $-2i + 4j + 2k$
  - Tidak ada jawaban yang benar**
16. Titik  $P(1, 1, 1)$  diputar berlawanan arah jarum jam sejauh  $120$  derajat dengan sumbu rotasinya adalah  $u = i + j + k$ . Maka:
- Vektor  $q$  yang menjadi rotornya adalah  $q = 1/2 + (1/2)i + (1/2)j + (1/2)k$
  - Vektor  $q^{-1}$  adalah  $W(-1) = 1/2 - (1/2)i - (1/2)j - (1/2)k$
  - Bayangan  $P$  adalah  $P'(1/2, -1, 1)$
  - Semua jawaban di atas benar
  - Hanya jawaban A dan B yang benar**
  - Semua jawaban salah
17. Jika  $q_1$  dan  $q_2$  adalah unit quaternion, dan  $1q_2^{-1}$  tidak sama dengan 0, maka persamaan di bawah ini adalah :
- $$\frac{q_1}{q_2} = \frac{q_1 \bar{q}_2}{|q_2|^2}.$$
- Salah
  - Benar**
  - Tidak bisa ditentukan
  - Jawaban A, B, C benar
  - Jawaban A, B, C salah
18. Jika  $p$  dan  $q$  adalah quaternion, maka pernyataan yang benar di bawah ini adalah : \*
- $\text{Conjugate}(pq) = \text{Conjugate}(p) \times \text{Conjugate}(q)$
  - $\text{Conjugate}(pq) = \text{Conjugate}(q) \times \text{Conjugate}(p)$
  - $|pq| = |p| \times |q|$
  - $|pq| = |q| \times |p|$
  - Jawaban (A) dan (C) benar
  - Jawaban (B) dan (D) benar
  - Jawaban (A) dan (D) benar
  - Jawaban (B) dan (C) benar**

**ALJABAR GEOMETRI**

19. Diberikan tiga vektor  $a = e_1 - 2e_2 + 2e_3$ ;  $b = e_1 - e_2 - 2e_3$ ; dan  $c = 2e_1 + e_2 - e_3$ . Hitunglah  $2a + b - 3c$
- $-3e_1 + 8e_2 + 5e_3$
  - $-3e_1 - 8e_2 - 5e_3$
  - $3e_1 + 8e_2 - 5e_3$
  - $-3e_1 - 8e_2 + 5e_3$**
  - Tidak ada jawaban yang benar
20. Hitunglah volume parallelepipedum yang dibentuk oleh  $a, b$ , dan  $c$ ; dimana  $a = 2e_1$ ,  $b = e_1 + 2e_2$ , dan  $c = 3e_3$

- a. 17
- b. 12
- c. 7
- d. 8
- e. Tidak ada jawaban yang benar

21. Jika  $A=e_1Ae_3$  dan  $B=(e_1+e_3)Ae_2$ . Hitunglah  $A+B$  dalam bentuk satu bivektor, pilihlah dari jawaban berikut yang benar

- a.  $(e_1 + e_3)^*(-0.5e_1 + e_2 + 0.5e_3)$
- b.  $(e_1 + e_3)^*(-e_1 + 2e_2 + e_3)$
- c.  $(-0.5e_1 + e_2 + 0.5e_3)^*(e_1 + e_3)$
- d.  $(-e_1 + 2e_2 + e_3)^*(e_1 + e_3)$
- e. Tidak ada jawaban yang benar

22. Dari pernyataan berikut, mana yang paling benar tentang multivector?

- a. Multivector pasti mengandung trivector.
- b. Multivector minimal mengandung scalar dan bivector
- c. Multivector pasti mengandung vector
- d. Multivector pasti mengandung scalar, vektor, dan bivektor
- e. Tidak ada pernyataan yang benar

### PERKALIAN GEOMETRI

23. Diberikan tiga vektor  $a=e_1-2e_2+2e_3$ ;  $b=e_1-e_2-2e_3$ ; dan  $c=2e_1+e_2-e_3$ . Hitunglah  $a \cdot b$ , dimana  $B=b \wedge c$

- a.  $15e_1e_3e_2$
- b.  $5e_1e_2e_3$
- c.  $10e_1e_2e_3$
- d.  $15e_1e_2e_3$
- e. Tidak ada jawaban yang benar

24. Bentuk paling sederhana dari  $(e_1e_2e_3)^7$  adalah

- a. -1
- b. 1
- c.  $e_1e_2e_3$
- d.  $e_2e_1e_3$
- e. Tidak ada jawaban yang benar

25. Vektor yang merupakan perpotongan antara dua bidang  $A=(e_1-e_2)^*(e_3-e_2)$  dengan  $B=e_2 \wedge e_3$  adalah

- a.  $e_1$
- b.  $e_1-e_2$
- c.  $e_3-e_2$
- d.  $e_1+e_2+e_3$
- e. Tidak ada jawaban yang benar

**UAS ALGEO 2021/2022****VEKTOR DAN NILAI EIGEN**

1. Diberikan matriks A , maka elemen a<sub>12</sub> dan a<sub>22</sub> dari A pangkat 5 adalah

$$\begin{pmatrix} 2 & -1 \\ -2 & 1 \end{pmatrix}$$

*Tandai satu oval saja.*

- A). 162 dan -162
- B). 162 dan -81
- C). -162 dan 81
- D). -162 dan -81
- E). Tidak ada jawaban yang benar

2. Diberikan sebuah matriks sebagai berikut. Dari matriks tersebut dapat disimpulkan

$$A = \begin{bmatrix} 2 & 0 & -1 \\ 0 & 0 & 4 \\ 1 & 0 & 0 \end{bmatrix}$$

*Tandai satu oval saja.*

- A) Jumlah semua nilai eigen = 2
- B) Rank(A) = 3
- C) Salah satu vektor eigennya adalah x = (2, 8, 2)
- D) Matriks A tidak dapat didiagonalisasi
- E) Jawaban A, B, C, dan D benar
- F) Hanya B dan C yang benar
- G) Hanya C dan D yang benar
- H) Semua jawaban di atas salah

3. Misalkan A adalah matriks berukuran n x n. P adalah matriks yang mendiagonalisasi matriks A, sedemikian sehingga menghasilkan matriks diagonal D. Hubungan antara A, P, dan D adalah

- A) D = PAP<sup>-1</sup> (Ket: <sup>A</sup> menyatakan tanda pangkat)
- B) trace(A) = trace(D) = trace(P)
- C) Ukuran (dimensi) matriks A dan P sama
- D) Nilai-nilai eigen A dan D sama
- E) rank(A) + nullity(A) + rank(D) + nullity(D) = 2n
- F) Semua jawaban B, C, D, dan E benar
- G) Hanya jawaban C dan D yang benar
- H) Hanya jawaban C, D, dan E yang benar
- I) Semua jawaban A sampai E benar
- J) Semua jawaban A sampai E tidak ada yang benar

4. Kata eigen berasal dari bahasa

- Belanda
- Jerman
- Inggris
- Rusia
- Semua jawaban benar
- Semua jawaban salah

5. Arti kata eigen adalah

- A. nama orang
- B. asli
- C. karakteristik
- D. vektor
- E. nilai
- Jawaban D dan E benar
- Jawaban B dan C benar
- Semua jawaban salah
- Semua jawaban benar.

### SVD DEKOMPOSISI

6. Diberikan matriks berukuran  $2 \times 3$  sebagai berikut. Matriks U dan Sigma hasil dekomposisi A dengan metode SVD adalah:

$$A = \begin{bmatrix} -2 & -1 & 2 \\ 2 & 1 & -2 \end{bmatrix}$$

$$A) \Sigma = \begin{bmatrix} 3\sqrt{2} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \underline{U} = \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{-\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{2}{2} & \frac{2}{2} \end{bmatrix}$$

$$B) \Sigma = \begin{bmatrix} 3\sqrt{2} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \underline{U} = \begin{bmatrix} \frac{-\sqrt{2}}{2} & \frac{-\sqrt{2}}{2} \\ \frac{2}{2} & \frac{2}{2} \\ \frac{-\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{2}{2} & \frac{2}{2} \end{bmatrix}$$

$$C) \Sigma = \begin{bmatrix} 3\sqrt{2} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \underline{U} = \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{2}{2} & \frac{2}{2} \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{2}{2} & \frac{2}{2} \end{bmatrix}$$

$$D) \Sigma = \begin{bmatrix} 3\sqrt{2} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \underline{U} = \begin{bmatrix} \frac{\sqrt{2}}{2} & \frac{-\sqrt{2}}{2} \\ \frac{2}{2} & \frac{2}{2} \\ \frac{\sqrt{2}}{2} & \frac{-\sqrt{2}}{2} \\ \frac{2}{2} & \frac{2}{2} \end{bmatrix}$$

$$E) \Sigma = \begin{bmatrix} 3\sqrt{2} & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \underline{U} = \begin{bmatrix} \frac{-\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{2}{2} & \frac{2}{2} \\ \frac{-\sqrt{2}}{2} & \frac{\sqrt{2}}{2} \\ \frac{2}{2} & \frac{2}{2} \end{bmatrix}$$

Jawaban A

Jawaban B

Jawaban C

Jawaban D

Jawaban E

Tidak ada jawaban yang benar

7. Pemfaktoran sebuah matriks menjadi perkalian matriks segitiga atas dan matriks segitiga bawahnya disebut dengan

Dekomposisi Triangle

Dekomposisi Orthogonal

Dekomposisi LU

Dekomposisi eigen

Dekomposisi geometri

Dekomposisi QR

Singular Value Decomposition

Semua jawaban benar

Semua jawaban salah

8. Pemfaktoran sebuah matriks menjadi perkalian vektor unit orthogonal dan matriks diagonal atasnya disebut dengan

Dekomposisi Triangle

Dekomposisi diagonal

Dekomposisi LU

Dekomposisi QR

Dekomposisi eigen

Singular Value Decomposition

Semua jawaban benar

Semua jawaban salah

9. Dari formula SVD di bawah ini, dimanakah unsur yang mengandung singular nya ?

$$A = U \Sigma V^T$$

U

Sigma

V

T

semua jawaban salah

semua jawaban benar

10. Berikut ini adalah aplikasi dari SVD di bidang informatika, kecuali

- Video kompresi
- Machine learning
- Computer vision
- Penjadwalan proses dalam Sistem Operasi
- Digital watermarking
- Semua jawaban benar
- Semua jawaban tidak benar.

11. Di bawah ini adalah nama-nama dekomposisi yang ada di Aljabar Geometri, kecuali :

- Dekomposisi LU
- Dekomposisi QR
- Dekomposisi nilai singular
- SVD
- Dekomposisi substitusi
- Semua jawaban benar
- Semua jawaban salah

## QUATERNION

12. Jika diberikan dua quaternion  $p=1 + 2i - j + 2k$  dan  $q=2 - i + 2j - 3k$ , maka nilai dari  $2p-q$  adalah

- A).  $3 + i + j - k$
- B).  $-1 + 3i - 3j - 5k$
- C).  $5i - 4j + 7k$
- D).  $4 + 3i + k$
- E). Tidak ada jawaban yang benar

13. Diketahui dua buah quaternion p dan q, yaitu  $p = 3 + 2i - j + 2k$  dan  $q = 2 + 2i - j + k$ . Maka, hasil  $p/q$  adalah (dibulatkan):

- 1.3 - 0.1i + 0.3j + 0.1k
- 1.3 - 0.3i - 0.1j - 0.1k
- 1.3 + 0.3i - 0.1j + 0.1k
- 1.3 + 0.1i - 0.3j + 0.1k
- 1.3 - 0.3i - 0.1j + 0.1k
- Semua jawaban di atas tidak ada yang benar

14. Sifat-sifat operasi quaternion yang SALAH adalah

- A) Jika  $p$  adalah quaternion, maka  $pp^{-1} = p^{-1}p$  (ket: tanda  $\wedge$  adalah pangkat)
- B) Perkalian dua buah quaternion murni tidak selalu menghasilkan quaternion murni
- C) Jika  $p$  dan  $q$  adalah quaternion, maka  $pq \neq qp$  (ket: tanda  $\neq$  berarti tidak sama)
- D) Jika  $p$ ,  $q$ , dan  $r$  adalah quaternion, maka  $(pq)r = p(qr)$
- E) Jika  $p$  quaternion dan  $p'$  adalah conjugate (seawan) dari  $p$ , maka  $pp' = p'p$
- F) jawaban B dan E
- G) Jawaban B, D, dan E
- Tidak ada jawaban yang memenuhi

15. Bilangan Quaternion ditemukan oleh

- Sir Robert Charles Quaternion
- Sir William Rowan Hamilton
- Sir Duke Anton Wales
- Anonim (tidak diketahui penemunya)
- Semua jawaban benar
- Semua jawaban salah

#### ALJABAR GEOMETRI DAN APLIKASINYA

16. Diberikan tiga buah vektor,  $a = -2e_1 + 3e_2$ ,  $b = 5e_1 - 2e_2$ , dan  $c = 4e_1 + 7e_2$ . Titik ujung vektor  $a$  adalah A, titik ujung vektor  $b$  adalah B, dan titik ujung vektor  $c$  adalah C. Luas segitiga ABC adalah

- 26
- 58
- 52
- 29
- 45
- 22,5
- Tidak ada jawaban yang benar

17. Jika  $I = e_1 e_2 e_3 e_4 e_5$  adalah pseudoskalar di  $R^5$ , maka nilai dari  $I^3$  adalah

- A). 1
- B). - 1
- C). I
- D). - I
- E). Tidak ada jawaban benar

18. Diketahui dua buah vektor di  $R^3$ ,  $a = 2e_1 + e_2 + 2e_3$ , dan  $b = e_1 - 3e_2 + 2e_3$ . Jika bivktor  $B = a \wedge b$  diproyeksikan terhadap bidang  $e_1 \wedge e_2$ , maka bayangannya adalah

- A).  $7(e_2 \wedge e_1)$   
 B).  $5(e_1 \wedge e_2)$   
 C).  $-1(e_2 \wedge e_1)$   
 D).  $-1(e_1 \wedge e_2)$   
 E). Tidak ada jawaban yang benar

19. Jika  $a$  sebuah vektor dan  $B$  adalah sebuah bivektor, maka pernyataan yang benar adalah

- A).  $a \cdot B$  dan  $a \wedge B$  bersifat komutatif  
 B).  $a \cdot B$  dan  $a \wedge B$  bersifat antikomutatif  
 C).  $a \cdot B$  bersifat komutatif dan  $a \wedge B$  bersifat antikomutatif  
 D).  $a \cdot B$  bersifat antikomutatif dan  $a \wedge B$  bersifat komutatif  
 E). Tidak ada jawaban yang benar

20. Jika diberikan bivektor  $B_1 = 3(e_1 \wedge e_2)$  dan bivektor  $B_2 = 2(e_3 \wedge e_1)$ , maka nilai dari  $B_1 - B_2$  adalah:

- A).  $e_1 \wedge (3e_2 + 2e_3)$   
 B).  $e_2 \wedge (3e_1 + 2e_3)$   
 C).  $e_3 \wedge (3e_1 - 2e_2)$   
 D).  $e_1 \wedge (3e_2 - 2e_3)$   
 E). Tidak ada jawaban yang benar

#### PERKALIAN GEOMETRI DAN APLIKASINYA

21. Misalkan  $P$  adalah multivektor,  $q$  dan  $r$  adalah vektor-vektor di  $R^2$  sedemikian sehingga  $P = qr$ . Jika diketahui  $P = 7e_1 - 4$  dan  $r = (e_1 + 2e_2)$ , maka

- A)  $q = -3e_1 + 2e_2$   
 B) Balikan (invers)  $q$  adalah  $(1/13)(-3e_1 + 2e_2)$   
 C)  $q = 2e_1 - 3e_2$   
 D) Balikan (invers)  $q$  adalah  $(1/13)(-2e_1 + 3e_2)$   
 E) A dan B benar  
 F) C dan D benar  
 G) Semua jawaban di atas salah

22. Ada banyak cara memutar (rotasi) vektor sejauh tetha derajat berlawanan arah jarum jam (sudut tetha sembarang). Misalkan  $v = ae_1 + be_2$  adalah vektor di  $R^2$ , maka cara merotasi  $v$  sejauh tetha derajat berlawanan arah jarum jam sehingga bayangannya adalah  $v'$  adalah dengan menggunakan persamaan:

A)  $v' = vI$  ( $I$  = pseudoscalar)

B)  $v' = ve^{i\phi}$  ( $I$  = pseudoscalar)

C)  $v' = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix}$

D)  $v' = qvq^{-1}$  ( $q$  dan  $q^{-1}$  adalah quaternion,  
 $q = \cos(\phi/2) + \sin(\phi/2)\hat{u}$ ,  $q^{-1} = \cos(\phi/2) - \sin(\phi/2)\hat{u}$   
 $\hat{u}$  = sumbu putar)

A, B, C, dan D

A dan B saja

B dan C saja

C dan D saja

C saja

B, C, dan D saja

Semua jawaban tidak ada yang benar

23. Diberikan dua buah vektor sebagai berikut:  $a = 2e_1 + e_2 - e_3$ , dan  $b = e_1 - e_2 - e_3$ . Perpotongan bidang yang dibentuk oleh vektor  $a$  dan  $b$  dengan bidang ( $e_2$  wedge  $e_3$ ) adalah:

$e_1 + 2e_2$

$e_1 - 2e_2$

$2e_1 - e_2$

$-e_1 - 2e_2$

$-e_1 + 2e_2$

$-2e_1 - e_2$

Tidak ada jawaban yang benar

24. Diketahui bivektor  $B=2(e_1 \wedge e_3)$ , jika  $B$  akan diubah bentuknya (re-shape) menjadi  $B'=D \wedge (3e_2+e_3)$ , maka bivektor  $D$  yang benar adalah

A).  $1/5(3e_3 + e_2)$

B).  $1/5(-3e_3 + e_2)$

C).  $1/5(3e_2 - e_2)$

D).  $1/5(-3e_3 - e_2)$

E). Tidak ada jawaban yang benar

25. Volume benda ruang (parallelepiped) yang dibentuk oleh tiga buah vektor  $a=e_1+2e_2+e_3$ ,  $b=2e_1-e_2+e_3$ , dan  $c=e_1+e_2-2e_3$  adalah

- A). 16
- B). 14
- C). 12
- D). 10
- E). Tidak ada jawaban yang benar



IF2130

Organisasi dan  
Arsitektur  
Komputer

## UTS ORGANISASI DAN ARSITEKTUR KOMPUTER

UTS SEMESTER I – 2002/2003

1. Pengertian dasar:
  - a. apa yang saudara ketahui tentang sistem komputasi?
  - b. apakah arsitektur komputer=organisasi komputer?
  - c. ada 2 jenis representasi informasi yang internal dan eksternal, jelaskan?
  - d. mengapa representasi dalam komputer tidak berbentuk desimal?
  - e. gambar diagram umum perangkat komputer, jelaskan dengan singkat fungsi dari setiap subsistem utamanya
2. Secara abstrak sistem komputer dapat dibagi enam tingkatan berdasarkan pada enam sudut pandang berbeda dan dengan masing-masing komponen di dalam tingkatan tersebut. Uraikan dengan lengkap dan jelas enam tingkatan tersebut.
3. Representasi informasi:
  - a. konversi representasi bilangan tidak bertanda bisa dilakukan dengan jalan pintas, coba buktikan representasi 1982 dalam oktal dan heksa.
  - b. jika  $m = 15$ , tuliskan -1999 dalam S\,M, OC, dan TC
4. Betapa jenius ide "OS" John Von Neumann sehingga mempermudah pekerjaan end-user, programmer, dan desainer OS. Jelaskan dengan singkat fungsi utama sistem komputer dan sudut pandang masing-masing dari ketiga pemakai terhadap fungsi tersebut.
5. Logika digital:
  - a. tuliskan de morgan's law dan buat truth tablenya
  - b. bila  $M = A \cdot B \cdot C + \sim A \cdot B \cdot C + A \cdot \sim B \cdot C + A \cdot B \cdot \sim C$ , buat truth table dan rangkaian sirkuitnya.
6. Memori
  - a. gambarkan hierarki tempat penyimpanan data, tuliskan 3 alasan penggambaran tersebut.
  - b. apa beda antara SRAM dan DRAM?
  - c. apa yang dimaksud dengan first fit, best fit, dan worst fit.
7. CPU:
  - a. apa yang saudara ketahui tentang register?
  - b. Mengapa format instruksi 4-address dan 3-address tidak pernah diimplementasikan

## UTS SEMESTER II – 2002/2003

1. Jawab cukup 1 baris, jelaskan pengertian saudara tentang:  
Loop, label, mask, monitor, nop, flog, address, nibble, pop, op code
2. Arsitektur SAP-3:
  - A) apa beda antara SBI dan SUI? Apa yang terjadi bila 990(10)-550(10)?
  - B) tulis register-register dari SAP-3 dengan kegunaannya.
  - C) apa yang dimaksud dengan PSW, bagaimana formatnya dan apa hubungan PSW dengan POP dan PUSH?
  - D) tuliskan 2 perintah untuk mengendalikan CY dan bagaimana perintah utk reset CY [agar CY=0]?
3. Arsitektur SAP-1:
  - A) berapa jumlah perintah yang digunakan?
  - B) apa yang dimaksud dengan LSB dan MSB, apa kaitannya dengan format instruksi?
  - C) apa hasilan dari CU?
  - D) tulis program [op code dan machine language] untuk aritmatika desimal:  $15 + 111 - 46 - 80 + 1$
4. Arsitektur SAP-2:
  - A) apa saja kelebihan arsitektur SAP-2 dari arsitektur SAP-1?
  - B) jumlah instruksi SAP-2 > SAP-1, tuliskan op code dari instruksi-instruksi SAP-2 yang mirip dengan instruksi-instruksi SAP-1.
  - C) apa yang dimaksud dengan handshaking?
  - D) misalkan ada program:

|       |     |        |
|-------|-----|--------|
| 3EH   | MVI | A, 00H |
| 06H   | MVI | B, 13H |
| 0EH   | MVI | C, 03H |
| 80H   | ... | ...    |
| 0DH   | ... | ...    |
| C2H   | ... | ...    |
| ... H | HLT |        |

- a) tulis dengan lengkap ["assembly + comment" dan "hand-made assemble"-nya]
  - b) program tersebut untuk apa?
  - c) berapa hasil akhirnya?
5. Apa yang dimaksud dengan microprocessor dan microprogram? Gambarkan diagram blok komputer lengkap dengan microprocessor?

UTS SEMESTER I – 2004/2005

1. Bila anda seorang pemrogram, jelaskan pemahaman anda tentang arsitektur komputer, organisasi komputer, dan instruction set architecture.
2. Komputer modern selain dituntut price/performance, juga customized design. Apa maksud price/performance dan mengapa costumized design?
3. Dalam memahami kinerja komputer, apa elapsed time = cycle time?
4. Hari Rabu besok akan diingat Bangsa Indonesia, karena Presiden pilihan rakyat akan dilantik.
  - A) konversi (tanggal + tahun) Rabu besok ke biner, oktal, dan heksa.
  - B) konversi (1852-2004) ke S/M, TC, OC, dan BCD bila m = bulan ini.
  - C) bila ascii untuk A = 101 (oktal), tulis biner dari akronim Presiden terpilih.
5. Sebuah truth table mempunyai keluaran 1 untuk masukan-masukan 0011, 0011, 1010, 1011, 1111; dan mempunyai keluaran X untuk masukan-masukan 0000, 0101, 0110, 1100, 1101, 1110. Buat karnaugh map, tulis ekspresi dan sederhanakan, dan gambar logic gatesnya.
6. Sederhanakan ekspresi boolean berikut, kemudian gambar diagram switching algebra, logic gates, dan truth table.  $\sim A \cdot \sim B + B \cdot C \cdot \sim D + A \cdot \sim C \cdot D + \sim B \cdot \sim C \cdot \sim D$

## UTS SEMESTER I – 2005/2006

1. Tuliskan 5 orang pakar yang berjasa dalam pengembangan komputer digital, lengkap dengan komputer yang diciptakannya.
2. Gambarkan 3 jenis konsep arsitektur komputer yang telah dikembangkan sampai saat ini.
3. Pada setiap baris, isikan bilangan yang merepresentasikan bilangan dengan baris yang sudah ditentukan:

| Base 2            | Base 10 | Base 16 |
|-------------------|---------|---------|
| a. 0010 0111      |         |         |
| b.                | 1300    |         |
| c.                |         | A0B     |
| d. 0001 0010 0011 |         |         |

4. Bila  $m=10$  konversi (1945 - 2005) ke sign mangitude, one's complement, two's complement, dan BCD.
5. Jika  $m = 4$ , kerjakan dengan two's complement :  $(-2) - (-5)$ .
6. Normalisasikan bilangan floating point berikut :

| Desimal | Base 2 | Base 8 | Base 16 |
|---------|--------|--------|---------|
| a 20,75 |        |        |         |
| b 16,4  |        |        |         |

7. Pada komputer hipotesis SAP-1:
  - A) tulis hand assemble menggunakan menggunakan mnemonic (LDA, ADD, SUB, OUT, HLT) yang mendisplay hasil komputasi (9-2+25-4)
  - B) apa isi register IR (instruction register) setelah selesai mengeksekusi instruksi pertama(LDA)
  - C) tuliskan 6 instruksi mikro untuk mengeksekusi (makro) instruksi ADD.
8. Misalkan diketahui....
  - A) buat tabel seperti di bawah ini dan isi-isinya kolom kosongnya:
  - B) buat K-Map dari tabel di atas
  - C) tuliskan bentuk canonical SOP dan canonical POS
  - D) gambar gerbang logikanya.

## UTS SEMESTER I – 2006/2007

1. Tuliskan bersambungan tanggal-bulan-tahun hari ini dalam desimal, kemudian kurangi NIM saudara. Hasilnya bagi dengan tahun ini, bila hasil akhir pecahan bulatkan ke atas bila lebih besar sama dengan 0.5; untuk m=12 dari hasil akhir tersebut kerjakan unsigned integer, signed magnitude, one's complement, dan two's complement.
2. Bila oktak ASCII: 0 = 060, a = 101 tulis representasi no parity dan with parity dari 28Okt06.
3. Tulis mnemonic dan machine language SAP-1 untuk aritmatika:  $11 - 21 + 2 + 60$ .
4. Tulis minter dan maxterm, tulis bentuk canonical SOM dan POM, serta gambar combinational logic circuit bila diketahui truth table berikut:

| A | B | C | D | Y |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |

## UTS SEMESTER I – 2007/2008

1. Bila diketahui satu karnaugh map(KM)

|   |   |   |   |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 |

Yang saudara kerjakan adalah:

- A) buat tabel lengkap(truth table) yang akan membentuk KM tersebut
  - B) tuliskan bentuk SOM dan POM
  - C) sederhanakan persamaan boolean dari SOM
  - D) gambar [combinational] logic circuit
2. Bila oktal ASCII : 0 = 060, a = 141, dan A = 101 tulis representasi *with parity* dari 25Okt07
  3. Tuliskan bersambungan tanggal-bulan-tahun hari ini dalam desimal, kemudian kurangi dengan NIM saudara. Hasilnya bagi dengan 9999, bila hasil akhirnya real dan lebih besar sama dengan 0,5 maka bulatkan ke atas; untuk m = 12 dari hasil akhir tersebut tuliskan *unsigned integer, signed magnitude, one's complement, dan two's complement.*
  4. Apa yang saudara ketahui tentang: teknologi, sistem, definisi arsitektur komputer Amdahl, filosofi RISC, price/performance.
  5. Apa yang menjadi dasar pemilihan representasi data?

UTS SEMESTER II – 2009/2010

1. Apa perbedaan antara periferal paralel dengan serial
2. Mengapa skasi lebih baik bila dibandingkan dengan IDE dan EIDE?
3. Kapan sistem diklasifikasi, oleh siapa, berapa jenis dan definisikan dengan ringkas tiap jenis dari klasifikasi tersebut.
4. Representasi data dapat dipandang dari tiga sudut pandang, tuliskan dan berikan contohnya
5. Kinerja tinggi komputer adalah dambaan para pemakai profesional, tergantung pada konteksnya aspek - aspek apa saja yang dianggap sebagai tolak ukurnya?
6. Tuliskan sepuluh aplikasi yang membutuhkan system komputer berkinerja tinggi.
7. Jumlahkan numerik dari NIM saudara dan tanggal-bulan-tahun hari peringatan proklamasi RI tahun ini, hasilnya masing-masing ubah ke biner dan lakukan aritmatika tambah dan kali.
8. Gambarkan diagram High Performance Bus.
9. Apa perbedaan antara big endian dengan little endian? Berikan tiga sistem contoh masing-masing implementasinya.
10. Tuliskan lima dasar kategori memori.

UTS SEMESTER II – 2010/2011

14 Maret 2011

90 menit

1. Apa yang dimaksud dengan sistem tertutup dan contohnya dalam sistem manusia?
2. Apa definisi arsitektur dan organisasi komputer menurut W. Stallings?
3. Gambarkan dan uraikan dengan singkat cara/aliran kerja sistem komputer.
4. Jumlahkan angka NIM saudara, kurangi dengan jumlah tanggal-bulan-tahun hari ini, kemudian kalikan hasilnya dengan hari kemerdekaan RI. Lakukan operasi aritmatika dalam representasi *unsigned integer*.
5. Apa fungsi dan bagaimana cara kerja DMA? Gambar konfigurasi-3 DMA dan uraikan keunggulannya dibanding dengan konfigurasi lain.
6. Tulis cara kerja cache dan gambar organisasi *three level cache*.
7. Uraikan dengan singkat tentang: *fetch cycle*, *execute cycle*, dan *interrupt cycle*.

UTS SEMESTER II – 2011/2012  
EL2010 - Organisasi & Arsitektur Komputer

Hari: Senin, 12 Maret 2012

Waktu: 90 menit

1. Apa yang dimaksud dengan sistem tertutup dan terbuka? Berikan contoh masing-masing!
2. Gambarkan dan uraikan dengan singkat cara/aliran kerja sistem komputer!
3. Lakukan operasi aritmatika dalam representasi *unsigned integer* untuk menjumlahkan nomor urut NIM saudara, kurangi dengan tanggal hari ini kemudian kalikan hasilnya dengan tanggal hari kemarin!
4. Apa yang dimaksud dengan teknologi informasi?
5. Tulis enam hasil yang dinyatakan sebagai Moore's Law!
6. Buat program SAP-1 untuk menjumlahkan nomor urut Program Studi IF, kurangi dengan tanggal hari ini kemudian tambah dengan tanggal hari kemerdekaan!
7. Apa manfaat kuliah ini bagi saudara?

UTS SEMESTER II – 2012/2013

**EL2010 – ORGANISASI & ARSITEKTUR KOMPUTER**

Hari: Senin, 18 Maret 2013

Waktu: 120 menit

Sifat: TUTUP Buku/Catatan, matikan HP, TANPA: Laptop/Notebook/Kalkulator

---

1. Jumlahkan angka-angka tanggal-bulan-tahun hari UTS ini, kemudian kalikan hasilnya dengan tanggal (tidak termasuk bulan dan tahun) dari hari kemerdekaan RI, lalu hasilnya ditambah dengan hasil penjumlahan angka-angka pada NIM Anda. Lakukan operasi aritmatika dalam representasi unsigned integer.
2. Sebuah representasi bilangan floating point menggunakan 6 bit, dengan pembagian 3 bit untuk eksponen dan 3 bit untuk fraction. Representasi ini tidak memiliki sign bit, sehingga hanya digunakan untuk merepresentasikan bilangan non-negatif. Hitunglah representasi bilangan untuk bilangan bulat berikut ini:
  - a.  $3/32$
  - b.  $8 \frac{1}{2}$
  - c.  $15/2$
  - d. Berapakah nilai terkecil (selain 0) yang dapat direpresentasikan?
  - e. Berapakah nilai terbesar (selain infinity) yang dapat direpresentasikan?
3. Jika  $x$  dan  $y$  bertipe `int`, tuliskan  $x$  dan  $y$  untuk ekspresi berikut yang menghasilkan nilai false. Tuliskan ‘Tidak ada’ jika untuk semua nilai  $x$  dan  $y$ , ekspresi tersebut menghasilkan nilai true.
  - a.  $((x \wedge y) < 0)$
  - b.  $((\sim(x \mid (\sim x + 1)) \gg 31) \& 0x1) == !x$
  - c.  $(x \wedge (x \gg 31)) - (x \gg 31) > 0$
4. Pasangan kode bahasa C dan kode assembly di bawah ini merupakan pasangan kode yang secara logika sama. Lengkapilah kode bahasa C dan kode assembly di bawah ini berdasarkan kesamaan logika tersebut.

Kode C:

```
int swit (int i, int j)
{
 int result;

 result = _____; // soal
 (a) switch (j) {
 case 1:
 _____; // soal
 (b) break;
 case 2:
 case 3:
 _____; // soal
 (c) case 4:
 _____; // soal
 (d) break;
 case 5:
 default
 :
 result = 0;
 }
```

---

```
 return result;
}
```

Kode Assembly:

```
_swit:
 pushl %ebp
 m
 o
 v
 l
 %
 e
 s
 p
 ,
 %
 e
 b
 p
 m
L9:
 o
 v
 l
 8
 (
 %
 e
 b
 p
)
L5:
 ,
 %
 e
 a
 x
 m
 o
 c
 l
 1
 2
 (
 %
 e
 b
 p
)
 ,
 %
 e
 d
 x
 l
 e
 a
 l
 (
 %
 e
 a
 x
L2:
 ,
 %
 e
 a
 x
 m
 o
 c
 l
 1
 2
 (
 %
 e
 b
 p
)
 ,
 %
 e
 d
 x
 l
 e
 a
 l
 (
 %
 e
 a
 x
```

,  
%  
e  
a  
x  
,2  
)  
,%  
e  
a  
x  
a  
d  
d  
1  
%  
e  
a  
x  
,%  
e  
a  
x  
c  
m  
p  
l  
\$  
3  
,%  
e  
d  
x  
j  
g  
  
L  
9  
c  
m  
p  
l  
\$  
2  
,%  
e  
d  
x  
j  
g

e  
L  
5  
1  
e  
a  
l  
(  
%  
e  
a  
x  
,%  
e  
a  
x  
,2  
)  
,%  
e  
a  
x  
c  
m  
p  
l  
\$  
1  
,%  
e  
d  
x  
j e  
j mp  
  
L2  
L8  
  
c  
m  
p  
l  
\$  
4  
,%  
e  
d  
x  
j  
e  
L

```

6
jmp L8
addl $4, %eax

s

o

a

l

(

e

)

p

o

p

l

%

e

b

p

ret

```

Soal (f): Berdasarkan pasangan kode di atas, lengkapi pasangan nilai case dalam kode bahasa C dengan urutan label yang dieksekusi untuk case tersebut dalam kode assembly.

| Nilai case | Urutan label yang dieksekusi |
|------------|------------------------------|
| Default    | L8 -> L2                     |
| 1          |                              |
| 2          |                              |
| 3          |                              |
| 4          |                              |
| 5          |                              |

5. Diketahui kode assembly sebagai berikut:

```

__test:
 p
 u
 s
 h
 l

 %

 e
 b
 p
 m
 o
 v
 l
 %

 e
 s
 p
 ,
L1:

```

```
%ebsubl$4,%esp)calltесьleal(%eax,%eax),%edx
movl%edx,%eax1eaxleavcals(%esp)
movl%eax,%eax
ret
```

Berdasarkan kode di atas:

- a. Lengkapilah kode C di bawah ini:

```
int test (unsigned x)
{
 if (_____) {
 i. _____;
 }
 else {
 i. _____;
 ii. _____;
 }
}
```

- b. Apa yang dilakukan oleh fungsi tersebut?  
c. Gambarkan perubahan isi stack yang terjadi selama eksekusi fungsi tersebut jika x = 2!
6. Diberikan kode C berikut:

```
static int bunny(int l, int r, int *A)
{
 int x = _____;
 int i = _____;
 int j = _____;
 while (_____)
 {
 do j--; while (_____);
 do i++; while (_____);
 if (_) {
 int t =
 A[i];
 A[i] = A[j];
 A[j] = t;
 }
 }
 return _____;
}
```

Hasil assembly-nya adalah sebagai berikut:

bunny:

```
pushl %ebp
movl %esp, %ebp
pushl %edi
pushl %esi
pushl %ebx
movl 8(%ebp), %eax
movl 16(%ebp), %esi
movl (%esi,%eax,4),
 %edi
leal -1(%eax), %ecx
movl 12(%ebp), %ebx
incl %ebx
cmpl %ebx, %ecx
jge .L3
.L16:
decl %ebx
cmpl %edi,
 (%esi,%ebx,4)
jg .L16
.L7:
incl %ecx
cmpl %edi,
 (%esi,%ecx,4)
```

```
jl .L7
cmpl %ebx, %ecx
jge .L3
movl (%esi,%ecx,4),
 %edx
movl (%esi,%ebx,4),
 %eax
movl %eax,
 (%esi,%ecx,4)
movl %edx,
 (%esi,%ebx,4)
jmp .L16
```

.L3:

ret

m  
o  
v  
l  
%  
e  
b  
x  
,,  
%  
e  
a  
x  
p  
o  
p  
l  
%  
e  
b  
x  
p  
o  
p  
l  
%  
e  
s  
i  
p  
o  
p  
l  
%  
e  
d  
i  
p  
o  
p  
l  
%  
e  
b  
p

- a. Isilah daftar tabel penggunaan register berikut (gunakan nama variabel pada kode C di atas)!

| Register | Variabel |
|----------|----------|
| %eax     |          |
| %ebx     |          |
| %ecx     |          |
| %edx     |          |
| %esi     |          |
| %edi     |          |
| %esp     |          |
| %ebp     |          |

- b. Lengkapi kode C yang belum terisi!

UTS SEMESTER I – 2013/2014  
**IF2130 Organisasi dan Arsitektur**  
Komputer Waktu: 120 menit.

**1. Representasi Bilangan - Integer (bobot 25)**

Tentukan representasi 8 bit two's complement dari

- $-1_{10}$
- $-104_{10}$
- Negatif dari  $01101111_2$
- Negatif dari  $0x58$
- Hasil dari operasi aritmatika  $0x53 + 0x3A$
- Hasil dari operasi aritmatika  $0x3A - 0x53$

**2. Representasi Bilangan - Floating Point (bobot 25)**

Representasi bilangan floating point menggunakan 7 bit, dengan 3 bit exponent dan 4 bit fraction (tidak ada nilai negatif). Lengkapilah tabel berikut:

| Nilai            | Normalized/<br>Denormalized | M x $2^E$       |   | Rep.<br>hexa | Perhitungan   |
|------------------|-----------------------------|-----------------|---|--------------|---------------|
|                  |                             | M               | E |              |               |
| 7                | Normalized                  | 1 $\frac{3}{4}$ | 2 | 0x5A         | exp:<br>frac: |
| $\frac{1}{4}$    |                             |                 |   |              |               |
| $24 \frac{1}{2}$ |                             |                 |   |              |               |
| $25 \frac{1}{2}$ |                             |                 |   |              |               |
| $\frac{1}{5}$    |                             |                 |   |              |               |
| $\frac{1}{8}$    |                             |                 |   |              |               |

**3. Bahasa Mesin - Addressing Mode (bobot 25)**

Pada sebuah komputer x86, diketahui isi alamat memori sebagai berikut:

| Alamat      | Nilai |
|-------------|-------|
| 0x1000 0000 | 0xAE  |
| 0x1000 0001 | 0x05  |
| 0x1000 0002 | 0x00  |
| 0x1000 0003 | 0x00  |
| 0x1000 0004 | 0x10  |
| 0x1000 0005 | 0xFE  |
| 0x1000 0006 | 0x18  |
| 0x1000 0007 | ,     |
| 0x1000 0008 | 0x22  |
| 0x1000 0009 | 0xE3  |
| 0x1000 000A | 0x77  |
|             | 0x21  |

Dan diketahui isi register sebagai berikut:

| Alamat | Nilai       |
|--------|-------------|
| %eax   | 0x1000 0004 |
| %ecx   | 0x0000 0003 |
| %edx   | 0x0000 0002 |
| %ebp   | 0x1000 0001 |
| %esp   | 0x1000 0005 |

Isilah tabel berikut sesuai dengan nama register dan alamat memori mana saja yang terupdate beserta isinya:

|           |                                                     |
|-----------|-----------------------------------------------------|
| Instruksi | Register dan alamat memori terupdate beserta isinya |
|-----------|-----------------------------------------------------|

|                            |  |
|----------------------------|--|
| addl %ecx, (%ebp, %edx, 2) |  |
|----------------------------|--|

|                                                                     |  |
|---------------------------------------------------------------------|--|
| movl 2(%ebp), %edx                                                  |  |
| leal 2(%ebp), %edx                                                  |  |
| leal (%eax, %edx, 2), %ecx<br>pushl %ecx                            |  |
| movl 4(%ebp), %edx<br>leal<br>0x4(%ebp, 2), %eax<br>subl %eax, %edx |  |

## 4. Bahasa Mesin - Control (bobot

25) Diberikan kode assembly

berikut:

```

loop :
pushl %ebp
movl %esp, %ebp
movl 8(%ebp), %ecx
movl 12(%ebp), %edx
xorl %eax, %eax
cmpl %edx, %ecx
jle .L4
.L6:
decl %ecx
incl %edx
incl %eax
cmpl %edx, %ecx
jg .L6
.L4:
incl %eax
movl %ebp, %esp
popl %ebp
ret

```

lengkapilah kode C berikut:

```

int loop (int x, int y) {
 int result;
 for(_____; _____; result++) {
 _____;
 _____;
 }
 _____;
 return result;
}

```

UTS SEMESTER I – 2014/2015  
 IF2130 Organisasi dan Arsitektur Komputer  
 Waktu: 120 menit

**1. Representasi Bilangan - Integer (bobot 25)**

Lengkapilah tabel konversi dari representasi 8 bit ke representasi 16 bit berikut ini.

| Representasi 8 bit |     |     | Representasi 16 bit |       |     |
|--------------------|-----|-----|---------------------|-------|-----|
| Biner              | B2U | B2T | Hexadecimal         | H2U   | H2T |
| 0000 1111          |     |     | 0x000f              |       |     |
|                    | 253 |     | 0xffffd             |       |     |
| 1000 0100          |     |     |                     | 65412 |     |
|                    | MAX |     |                     | 65535 |     |
|                    |     | MAX |                     | 127   |     |

Keterangan:

B2U : Binary to unsigned

B2T : Binary to two's

complement H2U : Hexadecimal to unsigned

H2T : Hexadecimal to two's complement

**2. Representasi Bilangan -Floating Point (bobot 25)**

Program berikut menerima sebuah bilangan single-precision IEEE floating point (32 bit, 1 bit sign, 8 bit exp dan 23 bit frac), dan menampilkan informasi kategori bilangan floating point tersebut. Lengkapilah kode yang masih kosong.

```
void classify_float(float f) {
 unsigned u = *(unsigned *) &f; // u memiliki bit pattern
 // sama dengan
 if sign = (u >> 31) &0x1; // sign bit
 int exp = _____; // exp field
 int frac = _____; // frac
 field
 // isilah ekspresi if di bawah dengan menggunakan nilai
 // sign, exp dan frac
 if

 printf("f bernilai 0 (positif/negatif)\n");
 else if _____
 printf("Non zero,
denormalized\n");
 else if _____
 printf("Infinity
(positif/negatif)\n");
 else if

 printf("NaN\n");
}
```

---

```
else if _____
 printf("Lebih besar dari -1.0 dan kurang dari
1.0\n");
else if _____
 printf("Lebih kecil dari atau sama dengan
-1.0\n");
else
 printf("Lebih besar atau sama dengan 1.0\n");
```

### 3. Bahasa Mesin – Addressing Mode (bobot 25)

Diketahui alamat suatu memori dan register berisi nilai sebagai berikut :

| Alamat | Nilai |
|--------|-------|
| 0x11C  | 0xAA  |
| 0x120  | 0xFC  |
| 0x124  | 0x22  |
| 0x128  | 0x33  |

| Register | Nilai |
|----------|-------|
| %eax     | 0x120 |
| %ebx     | 0x110 |
| %ecx     | 0x3   |
| %esp     | 0x101 |

Isi tabel berikut yang memperlihatkan akibat bila instruksi-instruksi tersebut dieksekusi, baik berupa tujuan tempat data hasil eksekusi disimpan, serta nilai yang diperoleh. Catatan: instruksi tidak saling berkaitan (setiap instruksi menggunakan kondisi awal pada tabel di atas). Diketahui lebar bit untuk nilai adalah **16 bit**.

| Instruksi                    | Tujuan | Nilai |
|------------------------------|--------|-------|
| movl %ebx, %ecx              |        |       |
| leal 5(%eax,%ecx), %eax      |        |       |
| movl 5(%eax,%ecx), %eax      |        |       |
| leal 17(,%esp,16), 0x4(%eax) |        |       |
| movl (%ebx,%ecx,4), 8(%eax)  |        |       |
| addl %esp,12(%ebx)           |        |       |
| subl \$6, (%ebx,%ecx,4)      |        |       |
| pushl %esp                   |        |       |
| negl 0x6(%ebx,%ecx,4)        |        |       |
| notl 0x4(%eax)               |        |       |

### 4. Bahasa Mesin – Control (bobot 25)

Diberikan kode assembly berikut:

```
foo:
pushl %ebp
movl
%esp,%ebp
pushl %ebx
movl 8(%ebp),%ebx
leal
2(%ebx),%edx
xorl %ecx,%ecx
cmpl %ebx,%ecx
jge .L4
```

```
.L6:
leal
5(%ecx,%edx),%edx
leal 3(%ecx),%eax
imull %eax,%edx incl
%ecx
cmpl
%ebx,%ecx jl
.L6
.L4:
```

```
movl %edx,%eax
popl %ebx
movl %ebp,%esp
popl %ebp
ret
```

**lengkapilah** kode C berikut:

```
int foo(int a)
{
 int i;
 int result = __a + 2_____;
 for(____i=0____; __i < a_____; i++) {
 _____result += (i + 5)_____;
 _____result *= (i + 3)_____;
 }
 return result;
}
```

UTS SEMESTER I – 2015/2016  
 IF2130 Organisasi dan Arsitektur Komputer  
 Waktu: 120 menit

**3. Representasi Bilangan - Integer (bobot 10)**

Jika Y bertipe int (signed 32 bit), dan MAX\_INT adalah nilai maksimum int, dan MIN\_INT adalah nilai minimum int, jelaskan nilai yang mungkin dari ekspresi berikut:

- a.  $Y \wedge (\text{MAX\_INT} + \text{MIN\_INT}) =$
- b.  $\sim ((Y >> 31) << 1) =$

**4. Representasi Bilangan – Integer (bobot 15)**

Diberikan kode berikut, apakah hasil yang ditampilkan pada layar? (hint: pada printf, %d untuk menampilkan nilai signed integer, dan %u untuk menampilkan nilai unsigned integer)

```
int x = 0x15213F10 >> 4;
char y = (char) x;
unsigned char z = (unsigned char)
x; printf("%d, %u", y, z);
```

Jawaban:

**5. Representasi Bilangan –Floating Point (bobot 25)**

Representasi bilangan floating point menggunakan 5 bit, dengan 3 bit exponent dan 2 bit fraction (tidak ada nilai negatif). Lengkapilah tabel berikut, dengan

M: Mantissa, dengan nilai  $0 \leq M < 1$  untuk representasi denormal, dan  $1 \leq M < 2$  untuk representasi normal.

| Nilai awal | Normalized/<br>Denormalized | $M \times 2^E$ |    | Rep.<br>Biner | Nilai akhir (setelah dibulatkan) |
|------------|-----------------------------|----------------|----|---------------|----------------------------------|
|            |                             | M              | E  |               |                                  |
| 9/32       | Normalized                  | 1              | -2 | 001 00        | ¼                                |
| 2          |                             |                |    |               |                                  |
| 13         |                             |                |    |               |                                  |
| 1/8        |                             |                |    |               |                                  |
| 1/5        |                             |                |    |               |                                  |
| 1/8        |                             |                |    |               |                                  |
| 15/4       |                             |                |    |               |                                  |

6. **Bahasa Mesin – Control** (bobot 25) Diberikan kode assembly berikut:

foo:

```
p
u
s
h
l
%
e
.L7: b
p
m
o
v
.L3: l
%
e
s
p
,
%
e
b
p
mo
vl
8(
%e
bp
),
%e
cx
```

```
//
a
mo
vl
16
(%
eb
p)
,%
ed
x
```

```
//
va
l
mo
vl
12
(%
eb
p)
,%
```

```

ea v
x 1
// %
i e
= b
n p
de ,
cl %
%e e
ax s
// i--
js .L3 // jump if negative
 p
 p
 o
 p
 l
 %
 e
 b
 p
 ret
m
o

```

**lengkapilah** kode C berikut:

```

int foo(int *a, int n, int val) {
 int i;
 for (i = _____; _____; _____) {
 ;
 }
 return i;
}

```

## 7. Bahasa Mesin – Prosedur (25)

Kode assembly berikut ini:

```

mystery1:
 pushl %ebp
 movl %esp,
 %ebp
 subl $8, %esp // allocate stack cmp $0, 8(%ebp) // cmp 0,
 n jne .L2 // if n != 0 jump L2
 movl $1, -4(%ebp) // tmp =
 1 jmp .L3

.L2: movl 8(%ebp), %eax // eax =
 n shr1 %eax // n >>
 1 movl %eax, (%esp) // set
 n >> 1 as param call
.L3: mystery1 addl $1, %eax
 // mystery1 + 1
 movl %eax,
 -4(%ebp)

 movl
 -4(%ebp)

```

```
, %eax
return
result
leave
ret
```

merupakan representasi dari kode C berikut:

```
unsigned mystery1(unsigned n)
{ if (_____
 return 1;
else
 return 1 + mystery1(_____
}
Lengkapilah bagian yang kosong pada kode C di atas, sesuai dengan deskripsi kode assembly di atas.
```

**1. Pemrograman Integer**

Open with ▾

Diketahui suatu kode menggunakan bahasa C sebagai berikut.

```
#define M /* konstanta bilangan pertama */
#define N /* konstanta bilangan kedua */
int arith(int x, int y)
{
 int result = 0;
 result = x*M + y/N;
 return result;
}
```

Kode ini dikompilasi untuk suatu nilai konstanta M dan N tertentu. Compiler melakukan optimisasi operasi perkalian dan pembagian. Ketika kode mesin hasil kompilasi diterjemahkan kembali ke bahasa C, diperoleh kode C sebagai berikut.

```
/* Translation of assembly code for arith */
int optarith(int x, int y)
{
 int t = x;
 x = x << 4;
 x = x - t;
 y = y >> 2; /* Arithmetic shift */
 return x+y;
}
```

Berapakah nilai M dan N tersebut?

| Const | Nilai | Perhitungan/Penjelasan |
|-------|-------|------------------------|
| M     |       |                        |
| N     |       |                        |

**2. Operasi Integer**

Palte 7 b.7.3

Untuk representasi biner bilangan x dan y berikut, tentukan masing-masing hasil operasi x dan y dalam bilangan biner, unsigned (desimal), dan two's complement (desimal).

| X (biner) | Y (biner) | Operasi    | Biner | Unsigned | Two's comp |
|-----------|-----------|------------|-------|----------|------------|
| 1010      | 0101      | (x+y)      |       |          |            |
| 1111      | 0100      | (y&x) + ~x |       |          |            |
| 1001      | 1000      | (x^x)   y  |       |          |            |
| 0110      | 1001      | x & !y     |       |          |            |
| 0110      | 0011      | x && ~y    |       |          |            |

### 3. Representasi Floating Point

Untuk merepresentasikan floating point digunakan tiruan format IEEE sepanjang 7 bit yang terdiri dari 1 bit tanda (sign), 3 bit eksponen, dan 3 bit fractional.

Lengkapilah tabel berikut:

Catatan: untuk mantissa ( $M$ ) bulatkan hingga 3 digit di belakang koma

| Deskripsi/<br>Nilai           | Representasi<br>Numerik |                                |   | Normalized/<br>Denormalized | Representasi<br>Floating Point |                  |                  | Representasi<br>Hexadesimal |
|-------------------------------|-------------------------|--------------------------------|---|-----------------------------|--------------------------------|------------------|------------------|-----------------------------|
|                               | +/-                     | M                              | E |                             | s                              | exp              | frac             |                             |
| +0                            | +                       | 0. <sup>000</sup> <sub>2</sub> | 0 | Denormalized                | 0 <sub>2</sub>                 | 000 <sub>2</sub> | 000 <sub>2</sub> | 0x00                        |
| $2^{34}$                      |                         |                                |   |                             |                                |                  |                  |                             |
| -0.225                        |                         |                                |   |                             |                                |                  |                  |                             |
| $31/16$                       |                         |                                |   |                             |                                |                  |                  |                             |
| Largest<br>Positive           |                         |                                |   |                             |                                |                  |                  |                             |
|                               |                         |                                |   |                             | 0 <sub>2</sub>                 | 000 <sub>2</sub> | 110 <sub>2</sub> |                             |
| Negative<br>Infinity          |                         |                                |   |                             |                                |                  |                  |                             |
|                               |                         |                                |   |                             |                                |                  |                  | 0x7A                        |
| Closest<br>negative<br>to 0.0 |                         |                                |   |                             |                                |                  |                  |                             |

### 4. Basic Assembly & Addressing Mode

Diberikan suatu prototipe fungsi :

```
int decode(int x, int y, int z);
```

Jika dikompilasi menjadi kode assembly, kode utamanya menjadi sebagai berikut :

1. movl 16(%ebp), %eax
2. movl 12(%ebp), %edx
3. subl %eax, %edx
4. movl %edx, %eax
5. imull 8(%ebp), %edx
6. sall \$31, %eax
7. sarl \$31, %eax
8. xorl %edx, %eax

Parameter x, y dan z disimpan di suatu lokasi memori dengan offset 8, 12 dan 16 relatif terhadap alamat pada register %ebp. Kode tsb menyimpan return value pada reg. %eax. Terjemahkan code assembly di atas menjadi implementasi bahasa C dari fungsi decode.

```
int decode(int x, int y, int z)
{
 // Your code here
}
```

## 5. Basic Assembly & Addressing Mode

Buatlah versi C dari kode assembly di bawah ini dengan hanya menggunakan 1 statement dengan maksimum 2 operator saja.

a. fool:

```
pushl %ebp
movl %esp,%ebp
movl 8(%ebp),%eax
sal l $4,%eax
subl 8(%ebp),%eax
movl %ebp,%esp
popl %ebp
ret
```

```
int fool(int x) {
 return _____;
}
```

b. foo2:

```
pushl %ebp
movl %esp,%ebp
movl 8(%ebp),%eax
shrl $31,%eax
movl %ebp,%esp
popl %ebp
ret
```

```
int foo2(int x) {
 return _____;
}
```

## UTS SEMESTER I 2019/2020

### Soal 1: Representasi Integer

Untuk representasi biner bilangan x dan y berikut, tentukan masing-masing hasil operasi x dan y dalam bilangan biner, heksadesimal, unsigned (desimal), dan two's complement (desimal). Baik x maupun y diinisiasi sebagai signed integer.

| x    | y    | Op                   | Biner | Hex | Unsigned | 2's comp. |
|------|------|----------------------|-------|-----|----------|-----------|
| 0110 | 1100 | (x & y)              |       |     |          |           |
| 1110 | 1111 | (x + y)              |       |     |          |           |
| 1000 | 0011 | (x >> 2A) ^ ~y       |       |     |          |           |
| 0010 | 1100 | (y && (y >> 2L)) - x |       |     |          |           |
| 0101 | 0011 | (y * x) & y          |       |     |          |           |

### Soal 2: Representasi Floating point

Sebuah representasi bilangan floating point menggunakan 8 bit, dengan pembagian 4 bit untuk exp dan 4 bit untuk frac. Representasi ini tidak memiliki sign bit, sehingga hanya digunakan untuk merepresentasikan bilangan non negatif. Hitunglah representasi bilangan untuk bilangan berikut ini:

- a. 3/4
- b. 9/32
- c. 7 1/2
- d. Berapakah nilai terkecil (selain 0) yang dapat direpresentasikan?
- e. Berapakah nilai terbesar (selain infinity) yang dapat direpresentasikan?

### Soal 3: Bahasa Assembly (Register & Memory Addressing)

Diketahui alamat suatu memori dan register berisi nilai sebagai berikut:

| Alamat | Nilai |
|--------|-------|
| 0x128  | 0xFF  |
| 0x12C  | 0x100 |
| 0x130  | 0x13  |
| 0x134  | 0x11  |

| Register | Nilai |
|----------|-------|
| %eax     | 0x128 |
| %ebx     | 0x1   |
| %ecx     | 0x3   |
| %edx     | 0x2   |

Isilah tabel berikut yang memperlihatkan akibat bila instruksi-instruksi tersebut dieksekusi, baik berupa tujuan penyimpanan hasil eksekusi, serta nilai yang diperoleh/disimpan.

Catatan: instruksi tidak saling berkaitan (setiap instruksi menggunakan kondisi awal pada tabel di atas).

| Instruksi                    | Tujuan | Nilai |
|------------------------------|--------|-------|
| movl %eax,%ebx               |        |       |
| movl \$0x120, 5(%eax,%ecx)   |        |       |
| movb %al, %bh                |        |       |
| movl 2(%eax,%edx,3),(%eax)   |        |       |
| leal (%eax,%edx,4),%eax      |        |       |
| movl (%eax,%edx,4),%eax      |        |       |
| leal 0x120(,%ecx,4), 8(%eax) |        |       |

|                     |  |  |
|---------------------|--|--|
| addl 12(%eax), %eax |  |  |
| negl (%eax)         |  |  |
| notl %ebx           |  |  |

**Soal 4: Bahasa Assembly (Mathematic & Logic Operator)****Bagian A**

Diketahui suatu fungsi dengan prototipe :

```
int hitung(int x, int y, int z);
```

Setelah fungsi tersebut dikompilasi menjadi kode assembly, dihasilkan :

```

1 movl 8(%ebp), %ecx
2 leal (%ecx,%ecx,2), %eax
3 sall $2, %eax
4 addl 16(%ebp), %eax
5 movl 12(%ebp), %edx
6 xorl $0xc, %edx
7 addl %ecx, %edx
8 imull %edx, %eax

```

Parameter x menempati lokasi stack %ebp+8, parameter y pada %ebp+12, dan parameter z pada %ebp+16.

Tuliskan fungsi hitung dalam kode bahasa pemrograman C yang menurut anda kira-kira akan menghasilkan kode seperti itu.

Tulis makna dari setiap baris assembly di atas

| Baris | Penjelasan |
|-------|------------|
| 1     |            |
| 2     |            |
| 3     |            |
| 4     |            |
| 5     |            |
| 6     |            |
| 7     |            |
| 8     |            |

Tuliskan fungsi hitung dalam kode bahasa pemrograman C yang menurut anda kira-kira akan menghasilkan kode tersebut (gunakan variabel lokal antara jika diperlukan).

**Bagian B**

Diberikan kode assembly berikut:

```

pushl %ebp
movl %esp,%ebp
movl 8(%ebp),%eax
sall $4,%eax
addl 8(%ebp),%eax
addl %eax,%eax
movl %ebp,%esp
popl %ebp
ret

```

Manakah kode C berikut (f4, f5 atau f6) yang sesuai dengan kode di atas?

*s*  
int f4(int a) {  
 return a \* 30;  
}

F5  
int f5(int a) {  
 return a \* 34;  
}

F6  
int f6(int a) {  
 return a \* 18;  
}

# UTS SEMESTER I 2020/2021

## Soal 1: Representasi Integer dan Word Ordering

Diberikan suatu spesifikasi representasi integer dalam bentuk 16 digit bit. Representasi ini digunakan dalam sebuah sistem dengan pengelompokan 4 bit per-*address space*.

- A) Berapa nilai tertinggi serta terendah yang dapat direpresentasikan spesifikasi tersebut baik sebagai **unsigned** maupun **2's-complement**?
- B) Ambil nilai digit kedua dan ketiga, kemudian gabungkan dengan tiga digit terakhir dari NIM Anda (misal: 13507021 → 35021). Nyatakan bilangan tersebut sebagai suatu bilangan desimal.
- i. Konversi bilangan ini sesuai dengan spesifikasi di atas, kemudian tunjukkan nilainya pada *address space* baik sebagai *big-endian* maupun *little-endian*!

*Big-endian*

| address+0 | address+1 | address+2 | address+3 |
|-----------|-----------|-----------|-----------|
|           |           |           |           |

*Little-endian*

| address+0 | address+1 | address+2 | address+3 |
|-----------|-----------|-----------|-----------|
|           |           |           |           |

- ii. Ambil **least significant byte** dari nilai di atas dan simpan sebagai suatu variabel A, kemudian lakukan perhitungan di bawah ini. Notasi operator diberikan sesuai penamaan dalam bahasa C. Hasil memuat 8-bit bilangan biner. Nyatakan interpretasi hasilnya dalam bentuk biner, heksadesimal, *unsigned*, serta *2's complement*!

| X         | y         | Operasi              | Binary | Hexa | Unsigned | Signed 2C |
|-----------|-----------|----------------------|--------|------|----------|-----------|
| 1010 0101 | 1110 1111 | (A & y) ^ x          |        |      |          |           |
| 1101 1011 | 1111 0011 | (~(y >> 4) && x) + A |        |      |          |           |
| 0010 1111 | 1010 0101 | ((x >> 2)   y) * A   |        |      |          |           |

## Soal 2: Floating Point

Soal di bawah ini merupakan 10 bit IEEE Floating Point format dengan alokasi bit sebagai berikut: s = 1 bit, exp = 3 bit dan frac = 6 bit. Dengan s adalah *sign*, M adalah Mantissa, E adalah eksponen yang memenuhi persamaan  $(-1)^s \times M \times 2^E$ .

Lengkapilah seluruh tabel berikut dengan teliti.

Lengkapilah seluruh tabel berikut dengan teliti.

| No     | Dekripsi/Nilai     | Representasi Numerik |         |    | Normalized / Denormalized | Representasi IEEE Floating Point |     |        | Nilai Desimal (Pecahan) |
|--------|--------------------|----------------------|---------|----|---------------------------|----------------------------------|-----|--------|-------------------------|
|        |                    | +/-                  | M       | E  |                           | s                                | exp | frac   |                         |
| Contoh | -12 3/4            | -                    | 1 19/32 | 3  | Normalized                | 1                                | 110 | 100110 | 12.75                   |
| 1      | +0                 | +                    |         |    |                           |                                  |     |        |                         |
| 2      | 4                  |                      |         |    |                           |                                  |     |        |                         |
| 3      | -8                 |                      |         |    |                           |                                  |     |        | -8.0                    |
| 4      | Closest to +0      | +                    |         |    | Denormalized              | 0                                | 000 | 000001 |                         |
| 5      | Closest to -0      | -                    |         |    | Denormalized              | 1                                | 000 | 000001 |                         |
| 6      | Largest Denorm.    | +                    | 63/64   | -2 | Denormalized              |                                  |     |        |                         |
| 7      | Smallest Norm.     | +                    | 1       |    | Normalized                |                                  |     |        |                         |
| 8      | Closest to 1 below | +                    | 1 63/64 | -1 | Normalized                |                                  |     |        |                         |
| 9      | 1                  | +                    |         |    |                           |                                  |     |        |                         |
| 10     | Closest to 1 above | +                    | 1 1/64  | 0  | Normalized                |                                  |     |        |                         |
| 11     | Largest Norm.      | +                    | 1 63/64 | 3  | Normalized                |                                  |     |        |                         |
| 12     | Infinity           | +                    | *       | *  | Special                   |                                  |     |        | +∞                      |
| 13     | NaN                | *                    | *       | *  | Special                   |                                  |     | 011001 | NaN                     |
| 14     | 7 7/8              |                      |         |    |                           |                                  |     |        |                         |
| 15     | -1 55/64           |                      |         |    |                           |                                  |     |        |                         |
| 16     | -3/16              |                      |         |    |                           |                                  |     |        |                         |
| 17     | 5/16               |                      |         |    |                           |                                  |     |        |                         |
| 18     | 15/64              |                      |         |    |                           |                                  |     |        |                         |
| 19     | -16/64             |                      |         |    |                           |                                  |     |        |                         |
| 20     | -17/64             |                      |         |    |                           |                                  |     |        |                         |

\*Tidak ada representasinya. Closest to 0: Mendekati 0. Closest to 1: Mendekati 1.

### Soal 3: Assembly

Diketahui alamat suatu memori dan register berisi nilai sebagai berikut :

| Alamat    | Nilai     |
|-----------|-----------|
| 0x12<br>8 | 0xFF      |
| 0x12<br>C | 0x10<br>0 |
| 0x13<br>0 | 0x13      |
| 0x13<br>4 | 0x11      |

| Register | Nilai |
|----------|-------|
| %rax     | 0x128 |
| %rbx     | 0x1   |
| %rcx     | 0x3   |
| %rdx     | 0x2   |

Isi tabel berikut yang memperlihatkan akibat bila instruksi-instruksi tersebut dieksekusi, baik berupa lokasi tujuan dimana hasil eksekusi disimpan (alamat memori ataukah register), serta nilai yang diperoleh.

Catatan: **setiap instruksi tidak saling berkaitan** (jalankan setiap instruksi menggunakan kondisi awal pada tabel di atas). Diketahui lebar bit untuk nilai adalah **16 bit**.

| Instruksi                    | Lokasi tujuan | Nilai |
|------------------------------|---------------|-------|
| movq %rax, %rbx              |               |       |
| movq \$0x120, 5(%rax,%rcx)   |               |       |
| movq 2(%rax,%rdx,3), (%rax)  |               |       |
| leaq (%rax,%rdx,4), %rax     |               |       |
| leaq 0x120(,%rcx,4), 8(%rax) |               |       |

| Instruksi                | Lokasi tujuan | Nilai |
|--------------------------|---------------|-------|
| addq %rcx, (%rax)        |               |       |
| subq %rdx, 4(%rax)       |               |       |
| negq %rcx                |               |       |
| imulq \$8, (%rax,%rcx,4) |               |       |
| decq 4(%rax)             |               |       |

### Soal 4: Assembly - Loop

Diberikan kode assembly sebagai berikut:

```

moo:
 pushl %ebp
 movl %esp,%ebp
 movl 12(%ebp),%ecx
 xorl %eax,%eax
 movl 8(%ebp),%edx
 cmpl %ecx,%edx
 jle .L3
 .align 4
.L5:
 addl %edx,%eax
 decl %edx
 cmpl %ecx,%edx
 jg .L5
.L3:
 leave
 ret
}

```

Berdasarkan kode assembly di atas, lengkapilah bagian yang kosong pada kode C yang di bawah ini:

```

int moo(int x, int y)
{
 int i, result=0;

 for (i=_____; _____; _____) {
 _____;
 }
 return result;
}

```

## UAS ORGANISASI DAN ARSITEKTUR KOMPUTER

UAS SEMESTER II – 2010/2011

10 Mei 2011

120 menit

Open  
book

Misalkan MX adalah suatu mesin 8 bit dengan 16 register memiliki instruksi sebagai berikut

| Kode instruksi                | keterangan                    | Waktu eksekusi (cycle) |
|-------------------------------|-------------------------------|------------------------|
| LOAD RDEST, RSEGMENT, ROFFSET | RDEST = MEM[RSEGMENT+ROFFSET] | 1                      |
| STORE RsRC, RSEGMENT, ROFFSET | MEM[RSEGMENT+ROFFSET]=RsRC    | 1                      |
| ADD RDEST, R1, R2             | RDEST=R1+R2                   | 2                      |
| SUB RDEST, R1, R2             | RDEST=R1-R2                   | 3                      |
| MUL RDEST, R1, R2             | (*)RDEST=R1*R2                | 5                      |
| INC R, DEC R                  | R = R+1, R = R-1              | 1                      |
| ZERO R, NEG R                 | R =0, R=-R                    | 1                      |
| BR RADDR / BZ R, RADDR        | PC=RADDR / IF R=0 THEN PC=R   | 1                      |
| BE R1, R2, RADDR              | IF R1=R2 THEN PC=RADDR        | 3                      |
| BL R1, R2, RADDR              | IF R1<R2 THEN PC=RADDR        | 3                      |
| BG R1, R2, RADDR              | IF R1>R2 THEN PC=RADDR        | 3                      |
| CALL R,RET                    | PC=R                          | 3,2                    |

(\*)nilai hasil disimpan di register dengan indeks DEST dan DEST+1. MSB disimpan di RDEST.  
Setiap register diindeks menggunakan angka 1-16 dengan panduan alokasi sebagai berikut:

R<sub>0</sub>-R<sub>9</sub> : bebas

R<sub>15</sub> : alamat segmen data di memori

R<sub>14</sub> : alamat offset data di memori

R<sub>10</sub>-R<sub>11</sub>, R<sub>12</sub>-R<sub>13</sub> : penampung hasil perkalian

Pengaksesan memori pada instruksi LOAD/STORE menggunakan kombinasi dua register yaitu

[RSEGMENT+ROFFSET]

```
#define MAX(a,b) (a>b) ? a:b
#define ABS(x) (x<0) ?-x:x
typedef struct {char w,h; char *data;} Image;
void AbsDiffM(Image I, Image M) {
 byte t,u,v,w,x,y;
 for (y=1; y<I.h-1; ++y)
 for (x=1; x<I.w-1; ++x) {
 t = ABS(I.data[(y-1)*I.w+(x+1)]-I.data[(y+1)*I.w+(x-1)]);
 u = ABS(I.data[(y+1)*I.w+(x+1)]-I.data[(y-1)*I.w+(x-1)]);
 v = ABS(I.data[y*I.w+(x+1)]-I.data[y*I.w+(x-1)]);
 w = ABS(I.data[(y+1)*I.w+x]-I.data[(y-1)*I.w+x]);
 M.data[y*M.w+x] = MAX(t, MAX(u, MAX(v, w)));
 }
}
```

Soal

1. Buatlah kode dalam bahasa assembly mesin MX untuk program tersebut
2. Buatlah versi kode yang lebih efisien dari kode jawaban soal nomor 1
3. Adakah dependency pada kode tersebut? Gambarkan dan jelaskan!
4. Jelaskan saran Anda terhadap arsitektur mesin MX untuk peningkatan kinerja program tersebut!

Saran

Buatlah dan jelaskan semua asumsi yang dibutuhkan untuk menjawab soal-soal di atas.

## UTS SEMESTER I 2021/2022

1. 0x12AB34CD adalah bilangan hexadesimal. Pada mesin Big Endian, tentukan isi alamat memory 0x01, 0x02, 0x03, 0x04 secara berturut tanpa 0x dan tanpa spasi atau koma. Contoh jawaban: 1234ABCD
2. manakah kode c yang akan memberikan hasil yang sama dengan leal 0x10(%rax, %rcx, 4), %rax?
  - a.  $\text{rax} = 16 + \text{rax} + 4 * \text{rcx}$
  - b.  $\text{rax} = *(16 + \text{rax} + 4 * \text{rcx})$
  - c.  $\text{rax} = 16 + *(\text{rax} + 4 * \text{rcx})$
  - d.  $*(16 + \text{rcx} + 4 * \text{rax}) = \text{rax}$
  - e.  $\text{rax} = 16 + 4 * \text{rax} + \text{rcx}$

3. suatu encoding IEEE floating point 7-bit memiliki format:

|            |                            |                              |
|------------|----------------------------|------------------------------|
| 1 sign bit | 4 bit eksponen ( $k = 4$ ) | 2 bit fraksional ( $n = 2$ ) |
|------------|----------------------------|------------------------------|

hitung bias:

- a. 4
  - b. 7
  - c. 8
  - d. 16
4. Suatu representasi IEEE floating point biner 7-bit memiliki format: 1-bit tanda (sign bit). 4-bit eksponen ( $k = 4$ ) dan 2-bit pecahan/fraksional ( $n = 2$ ). dengan aturan tersebut, encoding dari -INFINITY adalah
    - a. 0111100
    - b. 0111111
    - c. 1111100
    - d. 1111111
  5. Manakah perintah berikut yang dapat digunakan untuk menjumlahkan 2 register dan menyimpan hasilnya tanpa menimpa nilai awal kedua register tersebut
    - a. mov
    - b. lea
    - c. add
    - d. bukan salah satu di atas
  6. diketahui kode IEEE floating point biner 7-bit bernilai 0x44 (biner: 1000100) (format: 1-bit tanda (sign bit), 4-bit ekponen ( $k = 4$ ) dan 2-bit pecahan/fraksional ( $n = 2$ ). Nilai pecahan dari bilangan floating point tersebut (dalam basis 10 atau angka decimal ):
    - a.  $-1/64$
    - b.  $-1$
    - c.  $1/64$
    - d.  $1/128$
  7. Consider word size  $w = 4$  bits. The following table indicating the representation of  $U_{\text{Max}_w}$ ,  $T_{\text{Max}_w}$ ,  $T_{\text{Min}_w}$ ,  $-1$  dan  $0$  in hexadecimal and its numeric value

|                              | Hexadecimal | Numeric value (decimal) |
|------------------------------|-------------|-------------------------|
| U <sub>max<sub>4</sub></sub> | A1 = ...    | A2 = ...                |

|              |              |              |
|--------------|--------------|--------------|
| $T_{\max_4}$ | $B1 = \dots$ | $B2 = \dots$ |
| $T_{\min_4}$ | $C1 = \dots$ | $C2 = \dots$ |
| -1           | $D1 = \dots$ | $D2 = \dots$ |
| 0            | $E1 = \dots$ | $E2 = \dots$ |

$E2 = \dots ?$

Example Answer: -15, -8, -1, 0, 1, 8, 10, 15, etc

$C1 = \dots ?$

Example Answer: 0x1, 0x8, 0xA, 0xF, etc

$D1 = \dots ?$

Example Answer: 0x1, 0x8, 0xA, 0xF, etc

$A2 = \dots ?$

Example Answer: -15, -8, -1, 0, 1, 8, 10, 15, etc

$B1 = \dots ?$

Example Answer: 0x1, 0x8, 0xA, 0xF, etc

$A1 = \dots ?$

Example Answer: 0x1, 0x8, 0xA, 0xF, etc

$E1 = \dots ?$

Example Answer: 0x1, 0x8, 0xA, 0xF, etc

$C2 = \dots ?$

Example Answer: -15, -8, -1, 0, 1, 8, 10, 15, etc

$D2 = \dots ?$

Example Answer: -15, -8, -1, 0, 1, 8, 10, 15, etc

$E2 = \dots ?$

Example Answer: -15, -8, -1, 0, 1, 8, 10, 15, etc

8. Register Manakah yang digunakan untuk menyimpan parameter pertama pada pemanggilan fungsi
  - a. edi
  - b. esi
  - c. eax
  - d. bukan salah satu di atas
9. Suatu representasi IEEE floating point biner 7-bit memiliki format: 1-bit tanda (sign bit). 4-bit eksponen ( $k = 4$ ) dan 2-bit pecahan/fraksional ( $n = 2$ ). dengan aturan tersebut, encoding dari -NaN adalah
  - a. 0111100
  - b. 0111111
  - c. 1111100
  - d. 1111111
10. Consider word size  $w = 4$  bits. there are two constants. A and B in a single expression with the value of  $A = 7U$  and  $B = -7-1$ . What will be the relation and evaluation of those constants?
  - a.  $A > B$ , signed
  - b.  $A < B$ , signed
  - c.  $A > B$ , unsigned
  - d.  $A < B$ , unsigned
11. Konversikan bilangan biner berikut ke dalam bentuk decimal 10101111:

12. Manakah yang berikut ini benar
- tidak ada representasi bilangan floating point IEEE yang bernilai tepat 0
  - ada 1 representasi bilangan floating point IEEE yang bernilai tepat 0
  - ada 2 representasi bilangan floating point IEEE yang bernilai tepat 0
  - ada banyak representasi bilangan floating point IEEE yang bernilai tepat 0
13. diketahui kode IEEE floating point biner 7-bit bernilai 0x7C (biner: 1111100) (format: 1-bit tanda (sign bit), 4-bit eksponen (k = 4) dan 2-bit pecahan/fraksional (n = 2). Nilai bilangan floating point tersebut:
- 0
  - 1
  - INFINITY
  - NAN
14. Suatu representasi IEEE floating point biner 7-bit memiliki format: 1-bit tanda (sign bit). 4-bit eksponen (k = 4) dan 2-bit pecahan/fraksional (n = 2). dengan aturan tersebut, encoding dari +NaN adalah
- 0111100
  - 0111111
  - 1111100
  - 1111111
15. Apakah perbedaan antara arithmetic right shift dengan logical right shift
- arithmetic right shift digunakan pada C, logical right shift digunakan pada java
  - logical shift untuk data 32 bit, arithmetic untuk data 64 bit
  - keduanya akan mengisi bit pada bagian kiri dengan cara yang berbeda
  - sama saja
16. diketahui suatu bilangan pecahan decimal bernilai 1.5. Tentukan kode IEEE floating point biner 7-bit bilangan tersebut (format: 1-bit tanda (sign bit), 4-bit eksponen (k = 4) dan 2-bit pecahan/fraksional (n = 2)).
- 0x1E
  - 0x22
  - 0x5E
  - 0xA2
17. pada x86 64, addl %ebx, %eax mempunyai efek berikut
- %eax berisi %eax + %ebx, high-order 32 bits pada %rax diisi 0
  - %eax berisi %eax + %ebx, %rax tidak berubah
  - %eax berisi %eax + %ebx, high-order 32 bits dari %rax diisi sign-extended
  - %rax berisi %eax + %ebx
18. Berikut adalah format dari sebuah IEEE floating point; s = 1 bit, exp = 4 bit, dan frac = 2 bit. Nilai terkecil dari normalized (smallest normalized) dalam bentuk fractional binary number (direct conversion/non-IEEE floating point) adalah
- 0.0001000
  - 0.0000100
  - 0.0000010
  - 0.0000001
19. processor x86\_64 menggunakan

- a. big endian
  - b. little endian
  - c. no endian
  - d. bergantung system operasi
20. Suatu representasi IEEE floating point biner 7-bit memiliki format: 1-bit tanda (sign bit). 4-bit eksponen ( $k = 4$ ) dan 2-bit pecahan/fraksional ( $n = 2$ ). dengan aturan tersebut, encoding dari \_INFINITY adalah
- a. 0111100
  - b. 0111111
  - c. 1111100
  - d. 1111111
21. Konversikan bilangan hexa 0xABC ke dalam bentuk biner 12 bit. contoh jawaban: 000000111111
22. Argumen untuk fungsi pada x86 diletakkan pada
- a. stack
  - b. register
  - c. memori fisik
  - d. pada .text section
  - e. kombinasi stack dan register
23. pada program c berikut:
- ```
char z = -1;  
unsigned char r;  
r = z;
```
- tuliskan nilai r dalam decimal
24. Berikut ini adalah bagian dari hierarki memori: 0: Main Memory(DRAM), 1: L2 Cache, 2: Remote Storage, 3: Register, 4 : L1 Cache, 5: Local Storage, 6: L3 Cache. Urutkan hirarki memori dengan akses terlambat hingga tercepat. contoh jawaban: 0123456.
25. Berikut adalah format dari sebuah IEEE floating point; $s = 1$ bit, $exp = 4$ bit, dan $frac = 2$ bit. Nilai terbesar dari denormalized (biggest denormalized) dalam bentuk fractional binary number (direct conversion/non-IEEE floating point) adalah
- a. 0.0000000110
 - b. 0.0000001100
 - c. 0.0000011000
 - d. 0.0000110000
26. Manakah kode assembly yang akan memberikan hasil yang sama dengan $rcx = ((int *)rax)[rcx]?$
- a. mov (%rax,%rcx,4),%rcx
 - b. lea (%rax,%rcx,4),%rcx
 - c. lea (%rax,4,%rcx),%rcx
 - d. mov (%rax,4,%rcx),%rcx
27. jika %esp bernilai 0xBFFF0000 sebelum pemanggilan instruksi CALL, maka nilai %esp setelah pemanggilan instruksi CALL adalah
- a. 0xBFFEFFFC
 - b. 0xBFFF0004

-
- c. 0xBFFF0000
 - d. alamat instruksi setelah instruksi CALL
28. diketahui suatu bilangan pecahan decimal bernilai 10. Tentukan kode IEEE floating point biner 7-bit bilangan tersebut (format: 1-bit tanda (sign bit), 4-bit eksponen ($k = 4$) dan 2-bit pecahan/fraksional ($n = 2$)).
- a. 0x19
 - b. 0x29
 - c. 0x69
 - d. tidak ada representasi eksak
29. Berapakah ukuran register terpanjang pada arsitektur x86-64 Linux?
- a. 8 byte
 - b. 16 bit
 - c. 32 bit
 - d. 64 byte
30. A dan B bertipe data char, dimana $A = 10000001$ dan $B = 00000010$ dalam biner, C bertipe data unsigned char. Tuliskan nilai $C = A * B$ dalam decimal
31. isi %rsp adalah 0xdeadbeefdeadd0d0. Berapakah isi %rsp setelah instruksi ini dijalankan: pushq %rbx
- a. 0xdeadbeefdeadd0d4
 - b. 0xdeadbeefdeadd0d8
 - c. 0xdeadbeefdeadd0cc
 - d. 0xdeadbeefdeadd0c8
32. diberikan deklarasi bitop sbb:
- ```
int bitop(int x);
```
- hasil kompilasi pada mesin x86 adalah sbb:
- ```
bitop:
```
- ```
 push %ebp
 mov %esp, %ebp
 sub $0x8, %esp
 mov 0x8(%ebp), %eax
 not %eax
 inc %eax
 leave
 ret.
```
- manakah kose c yang sesuai dengan fungsi tersebut
- a.  $!(x+1)$
  - b.  $*(1-x)$
  - c.  $-x$
  - d.  $(x > 0 ? -x : -x + 1)$
33. A dan B bertipe data char, dimana  $A = 10000001$  dan  $B = 10001110$  dalam biner, C bertipe data unsigned char. Tuliskan nilai  $C = A + B$  dalam decimal
34. pada prosessor x86, posisi stack berada pada
- a. memori bagian bawah

- 
- b. tumbuh ke arah alamat yang lebih kecil
  - c. tumbuh ke alamat yang lebih besar
  - d. terletak pada heap
35. manakah instruksi berikut yang invalid pada x86
- a. pop %eip
  - b. pop %ebp
  - c. lea 0x10(%esp),%ebp
  - d. mob (%esp),%ebp
36. diketahui suatu bilangan pecahan decimal bernilai 6. Tentukan kode IEEE floating point biner 7-bit bilangan tersebut (format: 1-bit tanda (sign bit), 4-bit eksponen ( $k = 4$ ) dan 2-bit pecahan/fraksional ( $n = 2$ )).
- a. 0x18
  - b. 0x26
  - c. 0x5B
  - d. 0x66
37. dua bilangan biner  $A = 0110$  dan  $B = 0101$ .  $\sim$  adalah operasi inversi,  $\&$  adalah operasi AND, dan  $|$  adalah operasi OR. carilah nilai  $C = (\sim A | B) \& (A | B) \& A$  dalam bentuk desimal



UAS SEMESTER II – 2011/2012  
EL2010 - Organisasi & Arsitektur Komputer  
Hari: Rabu, 16 Mei 2012  
Waktu: 60 menit

1. Jelaskan yang dimaksud dengan *cache coherency*!
2. Jelaskan yang dimaksud dengan *direct memory access* (DMA)!
3. Apakah sebaiknya *cache* diimplementasi menggunakan RAM dinamik atau statik?

Jelaskan jawaban anda!

4. Pada kasus *Interaction Set* yang seperti apa kebijakan *write-back* menjadi kurang efektif?
  5. DMA dapat diimplementasi dalam bentuk salah satu dari tiga model konfigurasi yang dianggap baku/standar. Gambarkan diagram konfigurasi model-2, jelaskan jawaban anda!
  6. Apa perbedaan memory-mapped I/O dan isolated I/O?
  7. Scanner mengkonversi gambar/objek ke kode digital, jelaskan tiga jenis scanner yang dapat digunakan dalam berbagai kegiatan sehari-hari!
  8. Apa yang anda ketahui *Small Computer System Interface*?
  9. Gambar Struktur pohon taksonomi Flynn!
10. Raid:
- a. Singkatan dari : .....
  - b. Apa maksud/tujuan pengembangan teknik tersebut?
  - c. Jelaskan RAID level-1 dan RAID level-5!

UAS SEMESTER II – 2012/2013

## EL2010 – ORGANISASI &amp; ARSITEKTUR KOMPUTER

Hari: Jum'at 17 Mei 2013

Waktu: 120 menit

Sifat: TUTUP Buku/Catatan, matikan HP, TANPA: Laptop/Notebook/Kalkulator

- Diberikan kode C sebagai berikut :

```
#include <stdio.h>

int
overflow(void);
int one = 1;

/* main - NukeJr's main routine
 */
int main() {
 int val = overflow();

 val += one;
 if (val != 15213)
 printf("Boom!\n");
 ;
 else
 printf("Curses! You've defused NukeJr!\n");
 _exit(0); /* syscall version of exit that doesn't need %ebp */
}

/* overflow - writes to stack buffer and returns 15213
 */
int overflow() {
 char buf[4];
 int val,
 i=0;

 while (scanf("%x", &val) != EOF) buf[i++] = (char)val;
 return 15213;
}
```

Hasil kompilasinya pada Linux/x86 adalah sbb:

```
08048560 <main>:
8048560: 55 pushl %ebp
8048561: 89 e5 movl %esp,%ebp
8048563: 83 ec 08 subl $0x8,%esp
8048566: e8 31 00 00 00 call 804859c <overflow>
804856b: 03 05 90 96 04 addl 0x8049690,%eax #val += one;
8048570: 08
8048571: 3d 6d 3b 00 00 cmpl $0x3b6d,%eax #val == 15213?
8048576: 74 0a je 8048582 <main+0x22>
8048578: 83 c4 f4 addl $0xfffffffff4,%esp
804857b: 68 40 86 04 08 pushl $0x8048640
8048580: eb 08 jmp 804858a <main+0x2a>
8048582: 83 c4 f4 addl $0xfffffffff4,%esp
8048585: 68 60 86 04 08 pushl $0x8048660
804858a: e8 75 fe ff ff call 8048404 < init+0x44>#call printf
804858f: 83 c4 10 addl $0x10,%esp
```

---

```
8048592: 83 c4 f4 addl $0xffffffff4,%esp
8048595: 6a 00 pushl $0x0
8048597: e8 b8 fe ff ff call 8048454 < init+0x94> #call exit

0804859c <overflow>:
804859c: 55 pushl %ebp
```

```

804859d: 89 e5 movl %esp,%ebp
804859f: 83 ec 10 subl $0x10,%esp
80485a2: 56 pushl %esi
80485a3: 53 pushl %ebx
80485a4: 31 f6 xorl %esi,%esi
80485a6: 8d 5d f8 leal 0xffffffff(%ebp),%ebx
80485a9: eb 0d jmp 80485b8 <overflow+0x1c>
80485ab: 90 nop
80485ac: 8d 74 26 00 leal 0x0(%esi,1),%esi
80485b0: 8a 45 f8 movb 0xffffffff(%ebp),%al #L1: loop start
80485b3: 88 44 2e fc movb %al,0xfffffff(%esi,%ebp,1)
80485b7: 46 incl %esi
80485b8: 83 c4 f8 addl $0xffffffff,%esp
80485bb: 53 pushl %ebx
80485bc: 68 80 86 04 08 pushl $0x8048680
80485c1: e8 6e fe ff ff call 8048434 <init+0x74> #call scanf
80485c6: 83 c4 10 addl $0x10,%esp
80485c9: 83 f8 ff cmpl $0xffffffff,%eax
80485cc: 75 e2 jne 80485b0 <overflow+0x14> #goto L1
80485ce: b8 6d 3b 00 00 movl $0x3b6d,%eax
80485d3: 8d 65 e8 leal 0xffffffe8(%ebp),%esp
80485d6: 5b popl %ebx
80485d7: 5e popl %esi
80485d8: 89 ec movl %ebp,%esp
80485da: 5d popl %ebp
80485db: c3 ret

```

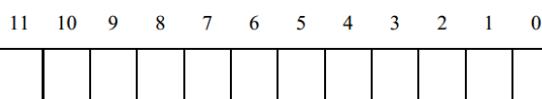
Berikut informasi tambahan yang diberikan:

- Mesin Linux/x86 menggunakan Little Endian.
  - Fungsi `scanf("%x", &val)` membaca sekuens karakter yang merepresentasikan hexa integer dari `stdin`, mengonversi menjadi 32-bit int, dan menyimpan hasilnya pada `val`. `scanf` mengembalikan 1 jika berhasil membaca nilai, dan EOF jika tidak ada nilai yang dibaca. Contoh : pemanggilan 4 kali `scanf` terhadap input string “0 a ff” akan menghasilkan sebagai berikut:
    - Pemanggilan `scanf` ke-1: `val = 0x0`, `scanf` mengembalikan 1.
    - Pemanggilan `scanf` ke-2: `val = 0xa`, `scanf` mengembalikan 1.
    - Pemanggilan `scanf` ke-3: `val = 0xff`, `scanf` mengembalikan 1.
    - Pemanggilan `scanf` ke-4: `val = ?, scanf` mengembalikan EOF.
- a. Tuliskan alamat beberapa objek yang terdapat pada stack saat instruksi `subl` pada fungsi overflow (alamat `0x804859f`) selesai dieksekusi sebagai offset dari `buf[0]`, dengan mengisi tabel berikut.
- | Stack object                  | Address of stack object      |
|-------------------------------|------------------------------|
| <code>return adrddress</code> | <code>&amp;buf[0] +</code>   |
| <code>old %ebpp</code>        | <code>&amp;buf[0] +</code>   |
| <code>buf[3]</code>           | <code>&amp;buf[0] +</code>   |
| <code>buf[2]</code>           | <code>&amp;buf[0] +</code>   |
| <code>buf[1]</code>           | <code>&amp;buf[0] + 1</code> |
| <code>buf[0]</code>           | <code>&amp;buf[0] + 0</code> |
- b. Tuliskan input string yang dapat mengakibatkan pemanggilan overflow kembali ke alamat `0x8048571` (bukan ke `0x804856b`).
2. Pada cache dengan aturan berikut:
- Memori akses berbasis byte (*byte addressable*)
  - Memory akses 1 byte word
  - Physical address 12 bit
  - Cache 4-way set associative, 2 byte line size, dan 32 total lines Isi cache adalah sebagai berikut:

| 4-way Set Associative Cache |     |       |        |        |     |       |        |        |     |       |        |        |     |       |        |        |
|-----------------------------|-----|-------|--------|--------|-----|-------|--------|--------|-----|-------|--------|--------|-----|-------|--------|--------|
| Index                       | Tag | Valid | Byte 0 | Byte 1 | Tag | Valid | Byte 0 | Byte 1 | Tag | Valid | Byte 0 | Byte 1 | Tag | Valid | Byte 0 | Byte 1 |
| 0                           | 29  | 0     | 34     | 29     | 87  | 0     | 39     | AE     | 7D  | 1     | 68     | F2     | 8B  | 1     | 64     | 38     |
| 1                           | F3  | 1     | 0D     | 8F     | 3D  | 1     | 0C     | 3A     | 4A  | 1     | A4     | DB     | D9  | 1     | A5     | 3C     |
| 2                           | A7  | 1     | E2     | 04     | AB  | 1     | D2     | 04     | E3  | 0     | 3C     | A4     | 01  | 0     | EE     | 05     |
| 3                           | 3B  | 0     | AC     | 1F     | E0  | 0     | B5     | 70     | 3B  | 1     | 66     | 95     | 37  | 1     | 49     | F3     |
| 4                           | 80  | 1     | 60     | 35     | 2B  | 0     | 19     | 57     | 49  | 1     | 8D     | 0E     | 00  | 0     | 70     | AB     |
| 5                           | EA  | 1     | B4     | 17     | CC  | 1     | 67     | DB     | 8A  | 0     | DE     | AA     | 18  | 1     | 2C     | D3     |
| 6                           | 1C  | 0     | 3F     | A4     | 01  | 0     | 3A     | C1     | F0  | 0     | 20     | 13     | 7F  | 1     | DF     | 05     |
| 7                           | 0F  | 0     | 00     | FF     | AF  | 1     | B1     | 5F     | 99  | 0     | AC     | 96     | 3A  | 1     | 22     | 79     |

- a. Box berikut merupakan format address fisik. Tuliskan dengan kode di bawah bit yang digunakan untuk:

- C0 – Block offset dalam cache line
- CI – Cache index
- CT – Cache tag



- b. Jika alamat yang diakses adalah 3B6, tuliskan entry cache yang diakses dan nilainya dalam hexadesimal dengan mengisi tabel di bawah.

- Address fisik:

|                      |                      |                      |                      |                      |                      |                      |                      |                      |                      |                      |                      |                      |                      |                      |                      |                      |
|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|
| 11                   | 10                   | 9                    | 8                    | 7                    | 6                    | 5                    | 4                    | 3                    | 2                    | 1                    | 0                    |                      |                      |                      |                      |                      |
| <input type="text"/> |

- Akses memori fisik:

| Parameter           | Value |
|---------------------|-------|
| Cache Offset (CO)   | 0x    |
| Cache Index (CI)    | 0x    |
| Cache Tag (CT)      | 0x    |
| Cache Hit? (Y/N)    |       |
| Cache Byte returned | 0x    |

3. Sebuah mesin memiliki cache direct mapped berukuran 64K, dengan block size 16 byte, dengan write-back dan write-allocate. Jika variabel lokal menggunakan register, dan isi awal cache kosong, hitunglah miss rate untuk kode berikut:

```
void copy_matrix (int dest[ROWS][COLS], int src[ROWS][COLS]) {
 int i, j;
 for (i = 0; i < ROWS; i++) {
 for (j = 0; j < COLS; j++) {
 dest[i][j] = src[i][j];
 }
 }
}
```

- a. Hitunglah cache miss rate jika ROWS = 128 dan COLS = 128
- b. Hitunglah cache miss rate jika ROWS = 128 dan COLS = 192
- c. Hitunglah cache miss rate jika ROWS = 128 dan COLS = 256

4. Diberikan kode berikut.

```
int counter = 0;
void handler(int sig) {
 counter++;
}

int main() {
 int i;
```

```
signal(SIGCHLD, handler);
for (i = 0; i < 5; i++) {
```

```
if (fork() == 0) {
 exit(0);
}
while (wait(NULL) != -1);
printf ("counter = %d\n", counter);
return 0;
}
```

- a. Apakah kode tersebut akan menghasilkan output yang sama setiap kali dijalankan?  
b. Jika jawabannya YA, tuliskan output yang ditampilkan. Jika jawabannya TIDAK, tuliskan semua kemungkinan output yang ditampilkan.
5. Lengkapi dengan benar paragraf berikut ini:  
RAM is traditionally packaged as \_\_\_\_\_. Basic storage unit is normally a \_\_\_\_\_. Multiple RAM chips form a memory. Each cell in \_\_\_\_\_ stores a bit with a \_\_\_\_\_. transistor circuit. Each cell in \_\_\_\_\_ stores a bit with a \_\_\_\_\_. One transistor is used for access. SDRAM uses a conventional clock signal instead of \_\_\_\_\_. By 2010, DDR SDRAM is standard for most \_\_\_\_\_ and \_\_\_\_\_ systems. DRAM and SRAM are \_\_\_\_\_. The memory hierarchy creates a \_\_\_\_\_ of storage that costs as much as the \_\_\_\_\_ storage near the \_\_\_\_\_, but that at the rate of the \_\_\_\_\_ near the \_\_\_\_\_ serves data to \_\_\_\_\_.
6. Linker
  - a. Jelaskan perbedaan *static linking* dan *dynamic linking*!
  - b. Mengapa *command-line orders* perlu diperhatikan pada saat *linking*?
  - c. Perubahan apa saja yang dilakukan saat *object files* dikonversi menjadi *executable files*?
7. Memori
  - a. Mengapa SRAM memiliki kinerja lebih cepat jika dibandingkan dengan DRAM, dan mengapa teknologi SRAM tidak umum digunakan dalam *main memory* komputer yang ada saat ini?
  - b. Jika prosesor ingin menjalankan operasi untuk menyimpan *value* yang disimpan oleh register EDX ke dalam alamat memori 0x0F3EA21,
    - i. Perintah apa yang sedang dijalankan oleh prosesor?
    - ii. Jelaskan secara rinci apa saja yang terjadi pada komponen sistem komputer saat pelaksanaan operasi tersebut!

**UAS SEMESTER I – 2013/2014**

1. Pada cache dengan aturan berikut:
- Akses memori berbasis byte (byte addressable)
  - Akses memori 1 byte word
  - Alamat fisik 12 bit
  - Cache 4-way set associative, 2 byte line size, 32 total lines
- Isi cache adalah sebagai berikut:

| 4-way set associative |     |   |       |    |     |   |       |    |     |   |       |    |     |   |       |    |
|-----------------------|-----|---|-------|----|-----|---|-------|----|-----|---|-------|----|-----|---|-------|----|
| Idx                   | Tag | V | Bytes |    |
|                       |     |   | 0     | 1  |     |   | 0     | 1  |     |   | 0     | 1  |     |   | 0     | 1  |
| 0                     | F0  | 1 | ED    | 32 | 8A  | 1 | BF    | 80 | 14  | 1 | EF    | 09 | BC  | 0 | 25    | 44 |
| 1                     | BC  | 0 | 03    | 3E | A0  | 0 | 16    | 7B | BC  | 1 | 8E    | 4C | E4  | 1 | FB    | B7 |
| 2                     | BC  | 1 | 54    | 9E | B6  | 1 | DC    | 81 | 00  | 0 | B6    | 1F | 74  | 0 | 10    | F5 |
| 3                     | BE  | 0 | 2F    | 7E | C0  | 1 | 27    | 95 | C4  | 0 | 07    | 11 | BC  | 0 | C7    | B7 |
| 4                     | 7E  | 1 | 32    | 21 | 8A  | 1 | 22    | C2 | BC  | 1 | BA    | DD | DC  | 0 | E7    | A2 |
| 5                     | 98  | 0 | A9    | 76 | 54  | 0 | BC    | 91 | 98  | 1 | 80    | BA | BC  | 1 | 48    | 16 |
| 6                     | 38  | 0 | 5D    | 4D | BC  | 1 | 69    | C2 | 8A  | 1 | A8    | CE | 38  | 1 | FA    | 93 |
| 7                     | 8A  | 1 | 04    | 2A | 9E  | 0 | B1    | 86 | CC  | 1 | 96    | 30 | BC  | 1 | F8    | 1D |

- a. Tentukan nilai S, E, dan B, lalu hitung berapa besarkah ukuran cache tersebut!

|            |  |
|------------|--|
| S          |  |
| E          |  |
| B          |  |
| Cache size |  |

- b. Box berikut merupakan format alamat fisik. Tuliskan/beri tanda dengan kode di bawah,bit mana yang digunakan untuk:

- C0 – Block offset dalam cache line
- CI – cache index
- CT – cache tag

(1 kotak merepresentasikan 1 bit)

11      10      9      8      7      6      5      4      3      2      1      0

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|--|

- c. Lengkapilah tabel berikut:

| Physical address | Byte offset | Cache index | Cache tag | Hit/Miss? | Byte returned (empty if miss) |
|------------------|-------------|-------------|-----------|-----------|-------------------------------|
| 0xBE6            |             |             |           |           |                               |
| 0xBCF            |             |             |           |           |                               |
|                  | 0x1         | 0x4         | 0xBC      |           |                               |

2. Perhatikan beberapa deklarasi sebagai berikut:

```
int
A[7];
short
B[2][5];
char C[3][4];
```

Apabila diketahui dalam memori ketiga array tersebut terletak berurutan, di mana address A adalah x, sedangkan array B terletak tepat setelah array A (address B > address A), dan array C terletak tepat setelah array B (address C > address B).

Lengkapilah tabel berikut:

| Array Reference | Address (relative to x) |
|-----------------|-------------------------|
| A[3]            |                         |
| B               |                         |
| B[2][3]         |                         |
| C               |                         |
| C[3][2]         |                         |

3. Diberikan kode berikut yang dikompilasi/dijalankan pada mesin Linux/x86:

```
void foo(char *x)
{
 int buf[1];
 strcpy((char *)buf, x);
}

void callfoo() {
 foo("abcdefghijklm");
}
```

Dengan hasil translasi assembly sebagai berikut:

```
080484f4 <foo>:
080484f4: 55 pushl %ebp
080484f5: 89 e5 movl %esp,%ebp
080484f7: 83 ec 18 subl $0x18,%esp
080484fa: 8b 45 08 movl 0x8(%ebp),%eax
080484fd: 83 c4 f8 addl $0xffffffff8,%esp
08048500: 50 pushl %eax
08048501: 8d 45 fc leal 0xfffffff8(%ebp),%eax
08048504: 50 pushl %eax
08048505: e8 ba fe ff ff call 80483c4 <strcpy>
0804850a: 89 ec movl %ebp,%esp
0804850c: 5d popl %ebp
0804850d: c3 ret
```

```

08048510 <callfoo>:
08048510: 55 pushl %ebp
08048511: 89 e5 movl %esp,%ebp
08048513: 83 ec 08 subl $0x8,%esp
08048516: 83 c4 f4 addl $0xffffffff4,%esp
08048519: 68 9c 85 04 08 pushl $0x804859c #push string address
0804851e: e8 d1 ff ff ff call 80484f4 <foo>
08048523: 89 ec movl %ebp,%esp
08048525: 5d popl %ebp
08048526: c3 ret

```

Fungsi `strcpy` (`char *dst, char *src`) mengkopi string dari alamat `src` (hingga ke karakter „\0“ penutup) ke alamat `dst`. `strcpy` tidak mencek ukuran buffer tujuan (`dst`).

Mesin Linux/x86 adalah berbasis **Little Endian** (Least significant byte has lowest address).

Nilai hexa karakter „a“ = 0x61, „b“=0x62, „c“=0x63, „d“=0x64,

„e“=0x65,  
„f“=0x66, „g“=0x67, „h“=0x68, „i“=0x69, „\0“=0x00.

Jelaskan apa yang terjadi sesaat setelah fungsi `call foo` memanggil `foo` dengan input string

“abcdefghi” dengan mengisi pertanyaan di bawah ini:

- a. Tuliskanlah isi memori pada lokasi berikut, sesaat setelah fungsi `strcpy` kembali ke fungsi

```

foo:
(6) buf[0] = 0x
(6) buf[1] =
(6) buf[2]
=

```

Sesaat sebelum instruksi `ret` pada alamat 0x0804850d dijalankan, apakah isi dari register (6) %ebp?

Sesaat setelah instruksi `ret` pada alamat 0x0804850d dijalankan, apakah isi dari register (6) %eip?

4. Diberikan kode berikut:

```
int array1[H][J];
int array2[J][H];
void copy_array(int x, int y) { array2[y][x]
 = array1[x][y];
}
```

Jika hasil assembly pada mesin x86-64 adalah sebagai berikut:

```
On entry:
%edi = x
%esi =
y #
copy_array:
 movslq %edi,%rdi
 movslq %esi,%rsi
 movq %rsi, %rdx
 salq $4, %rdx
 subq
 %rsi, %rdx addq
 %rdi, %rdx
 leaq 0(%rdi,8), %rax subq
 %rdi, %rax
 addq %rsi,
 %rax
 movl array1(%rax,4), %eax
 movl %eax,
 array2(%rdx,4) ret
```

Tentukan nilai H dan J.(20)

H = 15

J = 7

**UAS SEMESTER I – 2014/2015****1. Array**

Diberikan kode assembly berikut:

```
local_array:
 pushl %ebp
 movl %esp, %ebp
 subl $32, %esp
 movl $0x1, 0xfffffff0(%ebp)
 movl $0x3, 0xfffffff4(%ebp)
 movl $0x5, 0xfffffff8(%ebp)
 movb $0x69, 0xfffffffbeb(%ebp)
 movb $0x66, 0xfffffffec(%ebp)
 movb $0x69, 0xfffffffed(%ebp)
 movb $0x74, 0xfffffffef(%ebp)
 movb $0x62, 0xfffffffef(%ebp)
 movl -8(%ebp), %eax
 andl 8(%ebp), %eax
 movl %eax, -4(%ebp)
 leal -21(%ebp),
 %edx movl -4(%ebp),
 %eax addl %edx,
 %eax movzbl (%eax),
 %eax movsbl %al,
 %eax movl %ebp,%esp
 popl
 %ebp ret
```

Apabila diketahui bahwa huruf ‘a’ merupakan karakter ASCII ke 97 dan ‘b’ merupakan karakter ASCII ke 98, dan seterusnya. **Lengkapilah** kode bahasa C berdasarkan kode assembly di atas:

```
_____local_array(int i)
{
 _____A[3] = { _____ };

 _____B[5] = { _____ };

 int idx = _____;

 return _____;
}
```

Jika argumen i dalam fungsi local\_array diberi nilai 15, nilai apa yang akan

dikembalikan oleh fungsi local\_array tersebut? \_\_\_\_\_

Lengkapilah tabel berikut:

| Ekspresi | Alamat<br>(relatif terhadap %ebp di dalam stack frame) |
|----------|--------------------------------------------------------|
| old %ebp | %ebp + 0                                               |
| A        |                                                        |
| A[2]     |                                                        |
| B        |                                                        |
| B[4]     |                                                        |
| idx      |                                                        |

## 2. Cache

Diketahui sebuah komputer memiliki memori 32 byte, byte addressable, dengan isi memori sebagai berikut

| Address | Value | Address | Value | Address | Value | Address | Value |
|---------|-------|---------|-------|---------|-------|---------|-------|
| 0       | 0xF0  | 8       | 0xED  | 16      | 0x32  | 24      | 0x8A  |
| 1       | 0xBC  | 9       | 0x03  | 17      | 0x3E  | 25      | 0xA0  |
| 2       | 0xBC  | 10      | 0x54  | 18      | 0x9E  | 26      | 0xB6  |
| 3       | 0xBE  | 11      | 0x2F  | 19      | 0x7E  | 27      | 0xC0  |
| 4       | 0x7E  | 12      | 0x32  | 20      | 0x21  | 28      | 0x8A  |
| 5       | 0x98  | 13      | 0xA9  | 21      | 0x76  | 29      | 0x54  |
| 6       | 0x38  | 14      | 0x5D  | 22      | 0x4D  | 30      | 0xBC  |
| 7       | 0x8A  | 15      | 0x04  | 23      | 0x2A  | 31      | 0x9E  |

Untuk meningkatkan kinerja pembacaan memori, komputer tersebut memiliki **two way set associative cache** seukuran 8 byte dengan dua buah set.

$$S = \underline{\hspace{2cm}}$$

$s =$

$$E = \underline{\hspace{2cm}}$$

---

$$B = \underline{\hspace{2cm}}$$

e =

Jika dilakukan pengaksesan memori 1 byte word dengan urutan di bawah ini, tentukan apakah miss (m) atau hit (h), jika diasumsikan pada awalnya cache kosong (*cold cache*):

Gambarkan kondisi cache terakhir setelah selesainya pengaksesan memori.

| Set<br>Idx | Line 0 |   |       |   | Line 1 |   |       |   |
|------------|--------|---|-------|---|--------|---|-------|---|
|            | Tag    | V | Bytes |   | Tag    | V | Bytes |   |
|            |        |   | 0     | 1 |        |   | 0     | 1 |
| 0          |        |   |       |   |        |   |       |   |
| 1          |        |   |       |   |        |   |       |   |

### 3. Stack

Diberikan potongan kode berikut:

```
0x400446e3 <malloc+7>: call 0x400446e8 <malloc+12>
0x400446e8 <malloc+12>: popl %eax
```

Setelah instruksi `popl` selesai, berapakah isi register `%eax` (dalam hexa)?

### 4. Struktur Data

Diberikan struktur data sebagai berikut pada mesin IA32:

```
struct s1 {
 char a[3];
 union u1 b;
 int c;
};

struct s2 {
 struct s1 *d;
 char e;
 int f[4];
 struct s2 *g;
};

union u1 {
 struct s1 *h;
 struct s2 *i;
 char j;
};
```

Lengkapilah kode C yang kosong pada pasangan kode assembly – bahasa C di bawah ini

|                                                                                                                                 |                                                          |
|---------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------|
| <pre>procl: pushl %ebp movl %esp,%ebp movl 8(%ebp),%eax movl 12(%eax),%eax movl %ebp,%esp popl %ebp ret</pre>                   | <pre>int procl(struct s2 *x) { return x-&gt;_____;</pre> |
| <pre>proc2: pushl %ebp movl %esp,%ebp movl 8(%ebp),%eax movl 4(%eax),%eax movl 20(%eax),%eax movl %ebp,%esp popl %ebp ret</pre> | <pre>int proc2(struct s1 *x) { return x-&gt;_____;</pre> |

|                                                                                                                                                                      |                                                              |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------|
| <pre> proc3: pushl %ebp movl %esp,%ebp movl 8(%ebp),%eax movl (%eax),%eax movsb 4(%eax),%eax movl %ebp,%esp popl %ebp ret </pre>                                     | <pre> char proc3(union u1 *x) { return x-&gt;_____; } </pre> |
| <pre> proc4: pushl %ebp movl %esp,%ebp movl 8(%ebp),%eax movl (%eax),%eax movl 24(%eax),%eax movl (%eax),%eax movsb 1(%eax),%eax movl %ebp,%esp popl %ebp ret </pre> | <pre> char proc4(union u1 *x) { return x-&gt;_____; } </pre> |

5. Diberikan kode C rekursif berikut:

```

int silly(int n, int *p)
{
 int val, val2; if (n > 0)
 val2 = silly(n << 1, &val); else
 val = val2 = 0;
 *p = val + val2 + n;
}

```

dengan hasil assembly sebagai berikut

```

silly:
pushl %ebp
movl %esp,%ebp
subl $20,%esp
pushl %ebx
movl 8(%ebp),%ebx
testl %ebx,%ebx
jle .L3
addl $-8,%esp
leal -4(%ebp),%eax
pushl %eax
leal (%ebx,%ebx),%eax
pushl %eax
call silly
jmp .L4
.p2align 4,,7

```

```
.L3:
 xorl %eax,%eax
 movl %eax,-4(%ebp)
.L4:
 movl -4(%ebp),%edx
 addl %eax,%edx
 movl 12(%ebp),%eax
 addl %edx,%ebx
 movl %ebx,(%eax)
 movl -24(%ebp),%ebx
 movl %edx,%eax
 movl %ebp,%esp
 popl %ebp
ret
```

- a. Apakah variabel val disimpan pada stack? Jika iya, pada byte offset berapakah (relatif terhadap %ebp) variabel tersebut disimpan? Mengapa perlu disimpan pada stack?
  
  
  
  
  
  
- b. Apakah variabel val2 disimpan pada stack? Jika iya, pada byte offset berapakah (relatif terhadap %ebp) variabel tersebut disimpan? Mengapa perlu disimpan pada stack?
  
  
  
  
  
  
- c. Apakah ada nilai yang disimpan (jika ada) pada posisi -24(%ebp)? Jika ada yang disimpan, mengapa nilai tersebut perlu disimpan?
  
  
  
  
  
  
- d. Apakah ada nilai yang disimpan (jika ada) pada posisi -8(%ebp)? Jika ada yang disimpan, mengapa nilai tersebut perlu disimpan?

**UAS SEMESTER I – 2015/2016****1. Array (20)**

Diberikan kode berikut:

```
int mat1[M][N];
int mat2[N][M];
int tambah(int i, int j)
{
 return mat1[i][j] + mat2[i][j];
}
```

Dengan hasil translasi kode assembly sbb:

tambah:

```
pushl
%ebp movl
%esp,%ebp
movl 8(%ebp),%eax
movl 12(%ebp),%ecx
sall $2,%ecx
leal
0(%eax,%eax,8),%edx
subl %eax,%edx
leal
(%eax,%eax,4),%eax
movl
mat2(%ecx,%eax,4),%eax
addl
mat1(%ecx,%edx,4),%eax
movl %ebp,%esp
popl
%ebp
```

ret Tentukan nilai M

dan N. M =

N =

**2. Cache (20)**

Pada cache dengan aturan berikut:

- Memory akses berbasis byte (byte addressable)
- Memory akses 1 byte word
- Physical address 12 bit
- Cache 4-way set associative, 2 byte line size dan 32 total lines

Isi cache adalah sebagai berikut:

| 4-way Set Associative Cache |     |       |        |        |     |       |        |        |     |       |        |        |     |       |        |        |
|-----------------------------|-----|-------|--------|--------|-----|-------|--------|--------|-----|-------|--------|--------|-----|-------|--------|--------|
| Index                       | Tag | Valid | Byte 0 | Byte 1 | Tag | Valid | Byte 0 | Byte 1 | Tag | Valid | Byte 0 | Byte 1 | Tag | Valid | Byte 0 | Byte 1 |
| 0                           | 29  | 0     | 34     | 29     | 87  | 0     | 39     | AE     | 7D  | 1     | 68     | F2     | 8B  | 1     | 64     | 38     |
| 1                           | F3  | 1     | 0D     | 8F     | 3D  | 1     | 0C     | 3A     | 4A  | 1     | A4     | DB     | D9  | 1     | A5     | 3C     |
| 2                           | A7  | 1     | E2     | 04     | AB  | 1     | D2     | 04     | E3  | 0     | 3C     | A4     | 01  | 0     | EE     | 05     |
| 3                           | 3B  | 0     | AC     | 1F     | E0  | 0     | B5     | 70     | 3B  | 1     | 66     | 95     | 37  | 1     | 49     | F3     |
| 4                           | 80  | 1     | 60     | 35     | 2B  | 0     | 19     | 57     | 49  | 1     | 8D     | 0E     | 00  | 0     | 70     | AB     |
| 5                           | EA  | 1     | B4     | 17     | CC  | 1     | 67     | DB     | 8A  | 0     | DE     | AA     | 18  | 1     | 2C     | D3     |
| 6                           | 1C  | 0     | 3F     | A4     | 01  | 0     | 3A     | C1     | F0  | 0     | 20     | 13     | 7F  | 1     | DF     | 05     |
| 7                           | 0F  | 0     | 00     | FF     | AF  | 1     | B1     | 5F     | 99  | 0     | AC     | 96     | 3A  | 1     | 22     | 79     |

- a. Box berikut merupakan format address fisik. Tuliskan/beri tanda dengan kode di bawah, bit mana yang digunakan untuk:
- C0 - Block offset dalam cache line
  - CI - cache index
  - CT - cache tag

11    10    9    8    7    6    5    4    3    2    1    0

|  |  |  |  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|--|--|--|
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|--|--|--|

- b. Jika alamat yang diakses adalah 1C8, tuliskan entry cache yang diakses dan nilainya dalam hexadesimal dengan mengisi tabel di bawah.

- Address fisik:

11    10    9    8    7    6    5    4    3    2    1    0

|  |  |  |  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|--|--|--|
|  |  |  |  |  |  |  |  |  |  |  |  |  |
|--|--|--|--|--|--|--|--|--|--|--|--|--|

- Akses memori fisik:

| Parameter           | Value |
|---------------------|-------|
| Byte offset         | 0x    |
| Cache Index         | 0x    |
| Cache Tag           | 0x    |
| Cache Hit? (Y/N)    |       |
| Cache Byte returned | 0x .  |

### 3. Struktur Data (30)

Diberikan struktur data berikut:

```
struct
 Node{
 char
 c;
 double
 value;
 struct Node*
 next; int
 flag; struct
 Node* left;
 struct Node*
 right;
 };
typedef struct Node* pNode;

pNode NodeTree[N];
```

- a. Pada mesin berbasis Linux pada prosesor 32 bit Intel, gambarkanlah alokasi memori untuk struktur data Node:

|       |               |           |          |          |          |
|-------|---------------|-----------|----------|----------|----------|
| C (1) | kosong<br>(3) | value (8) | next (4) | flag (4) | left (4) |
|-------|---------------|-----------|----------|----------|----------|

- b. Jika register %eax berisi alamat awal array NodeTree, dan register %edx berisi i, lengkapilah referensi kode C yang diacu oleh bahasa assembly berikut:

|                                                                                                                                  |                |
|----------------------------------------------------------------------------------------------------------------------------------|----------------|
| sall \$2,%edx<br>leal<br>(%eax,%edx),%eax<br>movl (%eax),%eax<br>movl 24(%eax),%eax<br>movl 20(%eax),%eax<br>movl 20(%eax),%eax  | NodeTree[i] -> |
| sall \$2, %edx<br>leal<br>(%eax,%edx),%eax<br>movl (%eax),%eax<br>movl 12(%eax),%eax<br>movl 12(%eax),%eax<br>movl 16(%eax),%eax | NodeTree[i] -> |
| sall \$2,%edx<br>leal<br>(%eax,%edx),%eax<br>movl (%eax),%eax<br>movl 16(%eax),%eax                                              | NodeTree[i] -> |

#### 4. Rekursif (30)

Diberikan kode sebagai berikut:

```
int read_string()
{ char buf[8];
 scanf("%s",
 &buf); return
 buf[1];
}

int main() {
printf("0x%x\n", read_string());
return 0;
}

void secret(int
arg) { if(arg ==
0x15213)
while(1);
exit(-1);
}
```

Hasil kompilasinya pada Linux/x86 adalah sbb:

```
08048398 <read_string>:
8048398: 55 push %ebp
```

```
8048399: 89 e5 mov %esp,%ebp
804839b: 83 ec 18 sub $0x18,%esp
804839e: 8d 45 f8 lea
0xfffffff8(%ebp),%eax 80483a1: 89 44 24 04 mov
%eax,0x4(%esp,1)
80483a5: c7 04 24 74 84 04 08 movl $0x8048474,(%esp,1)
80483ac: e8 ef fe ff ff call 80482a0
<scanf> 80483b1: 8b 45 fc mov
0xfffffff8(%ebp),%eax
80483b4: c9 leave
80483b5: c3 ret

080483b6 <main>:
80483b6: 55 push %ebp
80483b7: 89 e5 mov %esp,%ebp
80483b9: 83 ec 08 sub $0x8,%esp
80483bc: 83 e4 f0 and
$0xffffffff,%esp 80483bf: b8 00 00 00 00 mov
$0x0,%eax 80483c4: 29 c4 sub %eax,%esp
80483c6: e8 cd ff ff ff call 8048398
<read_string> 80483cb: 89 44 24 04 mov
%eax,0x4(%esp,1)
80483cf: c7 04 24 77 84 04 08 movl $0x8048477,(%esp,1)
80483d6: e8 e5 fe ff ff call 80482c0
<printf> 80483db: b8 00 00 00 00 mov
$0x0,%eax
80483e0: c9 leave
80483e1: c3 ret

080483e2 <secret>:
80483e2: 55 push %ebp
80483e3: 89 e5 mov %esp,%ebp
80483e5: 83 ec 08 sub $0x8,%esp
80483e8: 81 7d 08 13 52 01 00 cmpl $0x15213,0x8(%ebp)
80483ef: 75 02 jne 80483f3 <secret+0x11>
80483f1: eb fe jmp 80483f1
<secret+0xf> 80483f3: c7 04 24 ff ff ff ff movl
$0xffffffff,%esp,1) 80483fa: e8 d1 fe ff ff call 80482d0 <exit>
80483ff: 90 nop
```

Berikut informasi tambahan yang diberikan:

- Mesin menggunakan Little Endian (least significant bytes has lowest address)
- Fungsi `scanf("%s", &buf)` membaca string dan menyimpannya sebagai null terminated string ke `buf`

- A. Jika program diberikan input string yang setara dengan kode hexa 69 6c 75 76 32 31 33, tuliskanlah isi stack yang menunjukkan alamat yang ditunjuk oleh register `%ebp` dan nilai stack yang relevan sesaat setelah pemanggilan `scanf`
- B. Tuliskan input string yang dapat mengakibatkan pemanggilan fungsi `secret`:

**UAS SEMESTER I – 2017/2018****[Struktur]**

1. Kode C berikut digunakan untuk mengelola nama penyewa kamar dalam sebuah hotel kecil. Setiap entri pada array penyewa digunakan untuk menyimpan nama penyewa. LANTAI merepresentasikan jumlah lantai pada hotel, KAMAR merepresentasikan jumlah kamar dalam satu lantai, dan LEN adalah jumlah karakter yang digunakan untuk menyimpan nama penyewa, didefinisikan 12.

```
char penyewa[LANTAI][KAMAR][LEN];

void pesan_kamar(int lantai, int kamar, char *namapelanggan) {
 strcpy(penyewa[lantai][kamar], namapelanggan);
}
```

Kode assembly untuk fungsi pesan\_kamar adalah sebagai berikut:

```
pesan_kamar:
 pushl %ebp
 movl %esp, %ebp
 movl 12(%ebp), %eax
 movl 16(%ebp), %edx
 pushl %edx
 movl 8(%ebp), %edx
 sall $4, %edx
 subl 8(%ebp), %edx
 leal (%eax,%eax,2), %eax
 leal penyewa(,%eax,4), %eax
 leal (%eax,%edx,4), %edx
 pushl %edx
 call strcpy
 movl %ebp, %esp
 popl %ebp
 ret
```

- a. Berapakah nilai konstanta KAMAR?

Jawaban:

- b. Karena terdapat bug, program mengakses penyewa [0] [1] [-2], nilai manakah yang sebenarnya diakses? (nyatakan dalam penyewa [-] [-] [-])

Jawaban:

- c. Struktur PENYEWA [LANTAI] [KAMAR] [LEN] di atas termasuk pemborosan memori, jika banyak kamar yang tidak terisi. Sehingga, implementasi tersebut diubah menjadi array 2 dimensi yang elemennya pointer, dan diisi dengan NULL jika tidak terisi. Implementasi yang baru memiliki struktur sebagai berikut:

```
char *penyewa[LANTAI][KAMAR][LEN];
void pesan_kamar(int lantai, int kamar, char *namapelanggan)
{
 penyewa[lantai][kamar] = malloc(LEN);
 strcpy(penyewa[lantai][kamar], namapelanggan);
}
```

Saat diujicoba, ternyata saat terdapat 20% kamar terisi, memori yang dapat dihemat sebanyak 168 byte. Berapakah jumlah LANTAI pada hotel ini?

Jawaban:

**[Prosedur]**

2. Diberikan prototipe fungsi C sbb:

```
void foo(char *a, int b);
```

dengan hasil translasi assembly bagian awalnya sbb:

```
foo:
 pushl %ebp
 movl %esp,%ebp
 subl $0x10,$esp
```

Gambarkanlah layout isi stack dengan menggunakan alamat yang terdapat pada register %ebp sebagai referensinya. Tuliskan posisi pada stack:

- i) Argument fungsi foo
- ii) Return address
- iii) Register yang nilainya disimpan pada stack
- iv) Alamat yang ditunjuk register %esp dan %ebp setelah operasi subl dijalankan.

3. Jawablah pertanyaan berikut pada tempat yang disediakan.

- a. Diketahui dua buah kode fungsi yaitu fungsil dan fungsii2 dalam bahasa assembly sebagai berikut:

(1)

```
funsil:
 pushl %ebp
 movl %esp, %ebp
 movl 8(%ebp), %ecx
 movl $1, %edx
 movl $1, %eax

.L3:
 imull %edx, %eax
 addl $1, %edx
 cmpl %ecx, %edx
 jle .L3
 popl %ebp
 ret
```

(2)

```
funssi2:
 pushl %ebp
 movl %esp, %ebp
 movl 8(%ebp), %ecx
 testl %ecx, %ecx
 jle .L4
 movl $1, %edx
 movl $1, %eax

.L3:
 imull %edx, %eax
 addl $1, %edx
 cmpl %edx, %ecx
 jge .L3
 jmp .L2

.L4:
 movl $1, %eax

.L2:
 popl %ebp
 ret
```

- (i) Fungsi manakah yang merupakan kode assembly dari bentuk pengulangan while? Jelaskan alasannya!

(ii) Fungsi manakah yang merupakan kode assembly dari bentuk pengulangan do-while? Jelaskan alasannya!

(iii) Kedua fungsi pada dasarnya melakukan hal yang sama, apa yang dilakukan kedua fungsi tersebut?

b. Diberikan kode assembly dari sebagai berikut:

```
pushl %ebp
movl %esp, %ebp
movl 16(%ebp), %eax
cmpl $8, %eax
ja .L2
jmp *.%L4(%eax, 4)

#Jump table section
.section .rodata
.align 4
.align 4
.L4:
.long .L2
.long .L3
.long .L5
.long .L6
.long .L2
.long .L10
.long .L8
.long .L2
.long .L8
.text
#End of jump table section

.L3:
movl 8(%ebp), %eax
imull 12(%ebp), %eax
jmp .L11
```

```
.L5:
 movl 8(%ebp), %eax
 addl 12(%ebp), %eax
 jmp .L11

.L6:
 movl 12(%ebp), %eax
 leal (%eax,%eax,4), %eax
 addl %eax, %eax
 jmp .L7

.L10:
 movl $3, %eax

.L7:
 subl 8(%ebp), %eax
 jmp .L11

.L8:
 movl 8(%ebp), %edx
 imull %edx, %edx
 movl 12(%ebp), %ecx
 leal (%edx,%ecx,2), %edx
 addl %edx, %eax
 jmp .L11

.L2:
 movl 12(%ebp), %eax
 imull %eax, %eax
 subl 8(%ebp), %eax

.L11:
 popl %ebp
 ret
```

Berdasarkan kode di atas, isilah bagian kosong pada kode bahasa C di bawah ini.

```
int getValue (int a, int b, int c) {
 int val= _____;
 switch (c) {
 case ___: { _____ }
 case ___:
 case ___: { _____ }
 default : { _____ };
 }
 return (val);
```

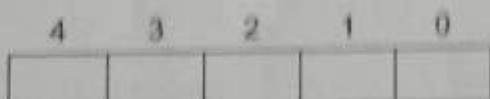
### [Cache]

4. Diketahui sebuah komputer memiliki memori 32 byte, byte addressable, dengan isi memori sebagai berikut

| Address | Value | Address | Value | Address | Value | Address | Value |
|---------|-------|---------|-------|---------|-------|---------|-------|
| 0       | 0xF1  | 8       | 0x0B  | 16      | 0x32  | 24      | 0xD   |
| 1       | 0xBC  | 9       | 0x04  | 17      | 0x3E  | 25      | 0xEF  |
| 2       | 0xAC  | 10      | 0x55  | 18      | 0x9E  | 26      | 0x02  |
| 3       | 0xBE  | 11      | 0x21  | 19      | 0x7E  | 27      | 0xAB  |
| 4       | 0x4E  | 12      | 0x43  | 20      | 0x21  | 28      | 0x8A  |
| 5       | 0x58  | 13      | 0x54  | 21      | 0x76  | 29      | 0x54  |
| 6       | 0x38  | 14      | 0x5D  | 22      | 0x4D  | 30      | 0x32  |
| 7       | 0x5A  | 15      | 0x04  | 23      | 0x2A  | 31      | 0x21  |

Untuk meningkatkan kinerja pembacaan memori, komputer tersebut memiliki **two way set associative cache** seukuran 8 byte dengan dua buah set dan block size = 2 byte. Apabila kedua line dalam satu set penuh lakukan penggantian pada line tertua (*not recently used*).

## Format alamat :



Tentukan format alamat dengan memberi label CO, CI dan CT pada setiap kotak di atas.  
(CO = block offset, CI = set index, CT = cache tag)

Jika dilakukan pengaksesan memori 1 byte word dengan urutan di bawah ini, tentukan apakah miss (m) atau hit (h), jika diasumsikan pada awalnya cache kosong (*cold cache*):

Tuliskan kondisi cache terakhir setelah selesainya pengaksesan memori tersebut diatas.

**UAS SEMESTER I – 2019/2020****Soal 1 Array**

Diberikan kode berikut:

```
int mat1[M][N];
int mat2[N][M];
int tambah(int i, int j)
{
 return mat1[i][j] + mat2[i][j];
}
```

Dengan hasil translasi kode assembly sebagai berikut:

tambah:

```
pushl %ebp
movl %esp,%ebp
movl 8(%ebp),%eax
movl 12(%ebp),%ecx
sall $2,%ecx
leal (%eax,%eax,4),%edx
sall $1,%edx
leal (%eax,%eax,2),%eax
movl mat2(%ecx,%eax,4),%eax
addl mat1(%ecx,%edx,4),%eax
movl %ebp,%esp
popl %ebp
ret
```

Tentukan nilai M dan N.

M =

N =

**Soal 2 Control**

Diberikan kode assembly berikut:

```
foo:
 pushl %ebp
 movl %esp,%ebp
 movl 8(%ebp),%ecx
 movl 16(%ebp),%edx
 movl 12(%ebp),%eax
 cmpl $0,%eax
 jl .L3
.L7:
 movl %edx,(%ecx,%eax,4)
 decl %eax
 jns .L7
.L3:
 movl %ebp,%esp
 popl %ebp
 ret
```

sesuaikanlah kode C berikut:

```
void foo(int *a, int n, int val) {
 int i;
 for (i = [inisiasi_i]; [kondisi_loop]; i = [increment_i]) {
 [isi_loop];
 }
}
```

inisiasi i = ?

kondisi loop = ?

increment i = ?

isi loop = ?

**Soal 3 Structure**

Berikut adalah suatu deklarasi struktur :

```
struct s1 {
 int a;
 char b;
 short c;
 float* d;
 double e;
 int f[3];
};
```

Kode struktur tersebut di-compile menggunakan mesin IA32-Windows.

- Gambarkan implementasi struktur tersebut pada memory, beserta penambahan paddingnya!
- Berapa total bit memory yang terpakai untuk implementasi struktur tersebut?
- Lakukan penataan ulang terhadap struktur tersebut agar dapat menghemat penggunaan memory!

**Soal 4 Cache Simulation**

Spesifikasi

$M = 64$  Byte address,  $B = 2^3 = 8$  Bytes/Block (2bit),  $S = 2^4 = 16$  (2 bit), Tag =  $2^4 = 16$  (2 bit)  
 Format:

| t  | s  | b  |
|----|----|----|
| XX | XX | XX |

Aturan pergantian adalah dengan Least Recently Used (LRU). A) Isilah tabel yang kosong di bawah ini. B) Tuliskan hanya hasil akhir pada keempat set cache (dengan 2 way associative) dibawah ini. C) Hitunglah miss rate dari ke 20 operasi tersebut.

| No | Alamat | Binary | Hit/Miss |
|----|--------|--------|----------|
| 1  | 0      | 000000 | hit      |
| 2  | 3      | 000011 |          |
| 3  | 63     | 111111 |          |
| 4  | 50     | 110010 |          |
| 5  | 37     | 100101 |          |
| 6  | 32     | 100000 |          |
| 7  | 10     | 001010 |          |
| 8  | 8      | 001000 |          |
| 9  | 21     | 010101 |          |
| 10 | 36     | 100100 |          |
| 11 | 12     | 001100 |          |
| 12 | 60     | 111100 |          |
| 13 | 11     | 001011 |          |
| 14 | 42     | 101010 |          |
| 15 | 43     | 101011 |          |
| 16 | 9      | 001001 |          |
| 17 | 25     | 011001 |          |
| 18 | 4      | 000100 |          |
| 19 | 62     | 111110 |          |
| 20 | 0      | 000000 |          |

set 0

| v | tag | block  |
|---|-----|--------|
| 1 | 0   | M[0-3] |
|   |     |        |

set 1

| v | tag | block |
|---|-----|-------|
|   |     |       |
|   |     |       |

set 2

| v | tag | block |
|---|-----|-------|
|   |     |       |
|   |     |       |

set 3

| v | tag | block |
|---|-----|-------|
|   |     |       |
|   |     |       |


 Scanned with

## UAS ORKOM 2020/2021

### Soal 1: Prosedur

```

int soal(int (*f)(int, int), int* arr, int c)
{
 int i, x;
 if (_____) return_____;
 x =_____;

 for (i =_____ ; _____; _____) x =_____;
 return x;
}

(gdb) dasas soal
Dump of assembler code for function soal:
0x080483a4 <soal+0>:push %ebp
0x080483a5 <soal+1>:mov %esp,%ebp
0x080483a7 <soal+3>:push %edi
0x080483a8 <soal+4>:push %esi
0x080483a9 <soal+5>:push %ebx
0x080483aa <soal+6>:sub $0xc,%esp
0x080483ad <soal+9>:mov 0xc(%ebp),%edi
0x080483b0 <soal+12>:mov 0x10(%ebp),%esi
0x080483b3 <soal+15>:test %esi,%esi
0x080483b5 <soal+17>:jle 0x80483db <soal+55>
0x080483b7 <soal+19>:mov (%edi),%edx
0x080483b9 <soal+21>:cmp $0x1,%esi
0x080483bc <soal+24>:jle 0x80483d9 <soal+53>
0x080483be <soal+26>:mov $0x1,%ebx
0x080483c3 <soal+31>:mov (%edi,%ebx,4),%eax
0x080483c6 <soal+34>:mov %eax,0x4(%esp)
0x080483ca <soal+38>:mov %edx,(%esp)
0x080483cd <soal+41>:call *0x8(%ebp)
0x080483d0 <soal+44>:mov %eax,%edx
0x080483d2 <soal+46>:add $0x1,%ebx
0x080483d5 <soal+49>:cmp %ebx,%esi
0x080483d7 <soal+51>:jg 0x80483c3 <soal+31>
0x080483d9 <soal+53>:mov %edx,%esi
0x080483db <soal+55>:mov %esi,%eax
0x080483dd <soal+57>:add $0xc,%esp
0x080483e0 <soal+60>:pop %ebx
0x080483e1 <soal+61>:pop %esi
0x080483e2 <soal+62>:pop %edi
0x080483e3 <soal+63>:pop %ebp
0x080483e4 <soal+64>:ret
End of assembler dump.

```

- Pada alamat 0x080483a9 terdapat instruksi `push %ebx`. Jelaskan 2 hal yang terjadi sebagai hasil dari eksekusi instruksi tersebut, dan jelaskan mengapa hal ini diperlukan.
- Misalkan setelah eksekusi instruksi pada alamat 0x080483a9 (`push %ebx`), nilai `%esp` adalah `0xfffff0000`, pada alamat berapakah parameter `f` berada?

### Soal 2: Cache Simulation

- Pada cache, apa yang dimaksud dengan:
  - Bit valid
  - Bit set
  - Bit tag
  - Bit block
- Pada cache, berapakah ukuran ukuran berikut:
  - Jumlah set, jika set bit adalah 2.
  - Berapa ukuran byte sebuah block, jika block bit adalah 4.
- Jelaskan kapan terjadi:
  - Cache Hit*
  - Cache Miss*
- Perhatikan *two-way set associative cache* berikut dengan aturan pergantian dengan *Least Recently Used (LRU)* dimana baris ke 2 pada set adalah data terbaru. Dengan komposisi bit tag (t) 2 bit, bit set (s) 1 bit, dan bit block (b) 4 bit seperti tabel berikut:

| t  | s | b    |
|----|---|------|
| xx | x | xxxx |

Cache sebelumnya telah terisi data sebagai berikut (v dan tag field adalah biner, sedangkan field block adalah decimal).

|       | v | tag | Block     |  | V     | Tag | Block |          |
|-------|---|-----|-----------|--|-------|-----|-------|----------|
| set 0 | 1 | 00  | M[0-15]   |  | set 1 | 1   | 01    | M[48-63] |
|       | 1 | 11  | M[96-111] |  |       | 1   | 00    | M[16-31] |

Selanjutnya program memanggil alamat-alamat berikut.

- Isilah kolom *hit/miss*!
- Isilah *cache* (set 0 & 1) dengan kondisi terakhir setelah alamat no 6 dijalankan!
- Hitunglah *miss rate*-nya!

| No | Alamat | Binary  | Hit/Miss |
|----|--------|---------|----------|
| 1  | 0      | 0000000 |          |
| 2  | 33     | 0100001 |          |
| 3  | 100    | 1100100 |          |
| 4  | 117    | 1110101 |          |
| 5  | 40     | 0101000 |          |
| 6  | 20     | 0010100 |          |

|       | v | tag | Block |  | v     | tag | Block |
|-------|---|-----|-------|--|-------|-----|-------|
| set 0 |   |     |       |  | set 1 |     |       |
|       |   |     |       |  |       |     |       |

### Soal 3: Array dan Struktur

#### Array

Diberikan kode berikut:

```
int mat1[M][N];
int mat2[N][M];
int tambah(int i, int j)
{
 return mat1[i][j] + mat2[i][j];
}
```

Dengan hasil translasi kode assembly sebagai berikut:

tambah:

```
pushl %ebp movl
%esp,%ebp
movl 8(%ebp),%eax movl
12(%ebp),%ecx sall $2,%ecx
leal (%eax,%eax,4),%edx sall $1, %edx
leal (%eax,%eax,2),%eax movl
mat1(%ecx,%eax,4),%eax addl
mat2(%ecx,%edx,4),%eax movl %ebp,%esp
popl %ebp ret
```

Tentukan nilai dari :

M =

N =

#### Struktur

Berikut adalah suatu deklarasi struktur :

```
struct s1 {
 int a; char b; short
 c; float* d; double
 e; int f[3];
 int g[3][4]
};
```

Kode struktur tersebut di-compile menggunakan mesin IA32-Windows.

- Nyatakan hasil implementasi struktur tersebut sebagai penempatan pada suatu rangkaian alamat memory, beserta penambahan paddingnya!  
(Gunakan notasi *base+offset* misal {s1+0 – s1+4: a, s1+4 – s1+?: b, s1+? – s1+?: padding, dst})
- Berapa total bit memory yang terpakai untuk implementasi struktur tersebut?
- Lakukan penataan ulang terhadap struktur tersebut agar dapat menghemat penggunaan memory!  
(Gunakan notasi yang sama dengan poin a)

### Soal 4: Buffer Overflow

Diketahui fungsi “hitung”, “panjang”, dan “tunjuk” yang melakukan penghitungan jumlah digit dari suatu bilangan integer:

```
int panjang(char *s) { return
 strlen(s);
}
void tunjuk(char *s, long *p) { long val = *p;
```

```

 sprintf(s, "%ld", val);
 }
int hitung(long x) { long y;
 char buf[12]; y = x;
 tunjuk(buf, &v); return
 panjang(buf);
}

```

Di bawah ini adalah potongan kode assembly dari fungsi “hitung”, yang di-compile dengan 2 cara:

**(a) Tanpa stack protector/canary:**

```

int hitung(long x) x disimpan di
%rdi
1 hitung:
2 subq $40, %rsp
3 movq %rdi, 24(%rsp)
4 leaq 24 (%rsp), %rsi
5 movq %rsp, %rdi
6 call tunjuk

```

**(b) Dengan stack protector/canary:**

```

int hitung(long x) x disimpan di
%rdi
1 hitung:
2 subq $56, %rsp
3 movq %fs:40, %rax
4 movq %rax, 40(%rsp)
5 xorl %eax, %eax
6 movq %rdi, 8(%rsp)
7 leaq 8(%rsp), %rsi
8 leaq 16(%rsp), %rdi
9 call tunjuk

```

Pertanyaan 1:

Dari informasi di atas, tentukan posisi-posisi berikut pada stack frame:

untuk cara (a),

- **buf** : ..... byte offset relatif terhadap %rsp
- **y** : ..... byte offset relatif terhadap %rsp
- 

untuk cara (b),

- **y** : ..... byte offset relatif terhadap %rsp
- **buf** : ..... byte offset relatif terhadap %rsp
- **canary**: ..... byte offset relatif terhadap %rsp

(catatan: posisi byte offset dapat bernilai positif atau negatif)

Pertanyaan 2:

Jelaskan secara sederhana bagaimana perubahan di bagian (b) dapat meningkatkan keamanan terhadap buffer overflow attack!

Pertanyaan 3:

Jelaskan 2 metode perlindungan di level sistem (system-level protection) terhadap buffer overflow attack!

## UAS ORKOM 2021/2022

1. Diberikan struktur data berikut pada mesin x86\_64:

```
typedef union {
 char c;
 char *p;
 int i;
 short s;
} union1;
typedef struct {
 char c;
 union1 u;
 int i;
 short s;
} struct1;
```

Berapakah alignment untuk struct1?

- a. 4 (0 Point)
- b. 8 (2 Point)
- c. 16 (0 Point)
- d. 24 (0 Point)

2. Diberikan struktur data berikut pada mesin x86\_64:t

```
typedef union {
 char c;
 char *p;
 int i;
 short s;
} union1;
typedef struct {
 char c;
 union1 u;
 int i;
 short s;
} struct1;
```

Berapakah offset untuk variabel s pada struct1?

- a. 12 (0 Point)
- b. 16 (0 Point)
- c. 20 (2 Point)
- d. 24 (0 Point)

Answer/Solution:

3. Diberikan struktur data berikut pada mesin x86\_64:

```
typedef union {
 char c;
 char *p;
 int i;
 short s;
} union1;
typedef struct {
 char c;
 union1 u;
 int i;
```

```
short s;
} struct1;
```

Berapakah jumlah space yang tidak digunakan pada struct1?

- a. tidak ada (0 Point)
- b. 1 (0 Point)
- c. 3 (0 Point)
- d. 7 (2 Point)

4. Diberikan kode matrix berikut (pada mesin x86):

```
struct elemen_matriks {
 tipe_1 a;
 tipe_2 b;
 int c;
 tipe_3 d;
}
struct elemen_matriks matriks[2][5];
int get_elemen(int i, int j) {
 return matriks[i][j].c;
}
```

Kode assembly untuk program di atas adalah sbb:

get\_elemen:

```
pushl %ebp
movl %esp,%ebp
movl 8(%ebp),%eax
leal (%eax,%eax,4),%eax
addl 12(%ebp),%eax
sall $4,%eax
movl matriks+4(%eax),%eax
movl %ebp,%esp
popl %ebp
ret
```

Apakah tipe sebenarnya dari tipe1 dan tipe2?

- a. tipe1= chartipe2= int (0 Point)
- b. tipe1= shorttipe2= short (2 Point)
- c. tipe1= inttipe2= int (0 Point)
- d. tipe1= shorttipe2= int (0 Point)

5. Diberikan kode matrix berikut (pada mesin x86):

```
struct elemen_matriks {
 tipe_1 a;
 tipe_2 b;
 int c;
 tipe_3 d;
}
struct elemen_matriks matriks[2][5];
int get_elemen(int i, int j) {
 return matriks[i][j].c;
}
```

Kode assembly untuk program di atas adalah sbb:

get\_elemen:

```
pushl %ebp
movl %esp,%ebp
```

```

movl 8(%ebp),%eax
leal (%eax,%eax,4),%eax
addl 12(%ebp),%eax
sall $4,%eax
movl matriks+4(%eax),%eax
movl %ebp,%esp
popl %ebp
ret

```

Apakah tipe sebenarnya dari tipe3?

- a. char (0 Point)
- b. short (0 Point)
- c. int (0 Point)
- d. double (2 Point)

6. Diberikan potongan kode berikut:

0x400440e1 malloc+7: call 0x400446e6 malloc+12

0x400440e6 malloc+12: popl %eax

Setelah eksekusi popl selesai, berapakah isi register %eax?

- a. 0x400440e1 (0 Point)
- b. 0x400440e6 (2 Point)
- c. isi register %ebp (0 Point)
- d. isi register %esp (0 Point)

7. Sebuah mesin memiliki direct mapped cache yang berukuran 64K dengan ukuran blok 16 byte. Cache menggunakan write-back dan write-allocate. Asumsi isi cache awal kosong, dan perhitungan menggunakan variabel lokal dilakukan menggunakan register. Pada kode berikut, matriks src diasumsikan dimulai pada alamat 0 dan matriks dest berada langsung setelah src

```

void copy_matrix(int dest[ROWS][COLS], int src[ROWS][COLS]) {
 int i, j;
 for (i=0; i<ROWS; i++) {
 for (j=0; j<COLS; j++) {
 dest[i][j] = src[i][j]; } } }

```

Berapakah miss rate jika ROWS = 128 dan COLS = 128?

- a. 12,5% (0 Point)
- b. 25% (0 Point)
- c. 50% (0 Point)
- d. 100% (2 Point)

8. Sebuah mesin memiliki direct mapped cache yang berukuran 64K dengan ukuran blok 16 byte. Cache menggunakan write-back dan write-allocate. Asumsi isi cache awal kosong, dan perhitungan menggunakan variabel lokal dilakukan menggunakan register. Pada kode berikut, matriks src diasumsikan dimulai pada alamat 0 dan matriks dest berada langsung setelah src

```

void copy_matrix(int dest[ROWS][COLS], int src[ROWS][COLS]) {
 int i, j;
 for (i=0; i<ROWS; i++) {
 for (j=0; j<COLS; j++) {
 dest[i][j] = src[i][j]; } } }

```

Berapakah miss rate jika ROWS = 128 dan COLS = 192?

- a. 12,5% (0 Point)
- b. 25% (2 Point)
- c. 50% (0 Point)
- d. 100% (0 Point)

9. Sebuah mesin memiliki direct mapped cache yang berukuran 64K dengan ukuran blok 16 byte. Cache menggunakan write-back dan write-allocate. Asumsi isi cache awal kosong, dan perhitungan menggunakan variabel lokal dilakukan menggunakan register. Pada kode berikut, matriks src diasumsikan dimulai pada alamat 0 dan matriks dest berada langsung setelah src

```
void copy_n_flip_matrix2(int dest[ROWS][COLS], int src[ROWS][COLS]){
 int i, j;
 for (j=0; j<COLS; j++) {
 for (i=0; i<ROWS; i++) {
 dest[i][COLS - 1 - j] = src[i][j]; } } }
```

Berapakah miss rate jika ROWS = 128 dan COLS = 128?

- a. 25% (2 Point)
- b. 50% (0 Point)
- c. 75% (0 Point)
- d. 100% (0 Point)

10. Sebuah mesin memiliki direct mapped cache yang berukuran 64K dengan ukuran blok 16 byte. Cache menggunakan write-back dan write-allocate. Asumsi isi cache awal kosong, dan perhitungan menggunakan variabel lokal dilakukan menggunakan register. Pada kode berikut, matriks src diasumsikan dimulai pada alamat 0 dan matriks dest berada langsung setelah src

```
void copy_n_flip_matrix2(int dest[ROWS][COLS], int src[ROWS][COLS]){
 int i, j;
 for (j=0; j<COLS; j++) {
 for (i=0; i<ROWS; i++) {
 dest[i][COLS - 1 - j] = src[i][j]; } } }
```

Berapakah miss rate jika ROWS = 192 dan COLS = 128?

- a. 25% (0 Point)
- b. 50% (0 Point)
- c. 75% (2 Point)
- d. 100% (0 Point)

11. Teknik mengurangi resiko serangan buffer overflow dengan cara menggunakan flag --stack-protector pada compiler GCC disebut

- a. stack randomization (0 Point)
- b. stack smashing (0 Point)
- c. canary (2 Point)
- d. read-only stack (0 Point)

12. Manakah di antara berikut ini yang tidak tepat tentang isi section .symtab (symbol table) pada ELF Object File Format

- a. berisi nama prosedur/function (0 Point)
- b. berisi nama global variabel (0 Point)
- c. berisi nama local variabel (2 Point)
- d. menyediakan informasi ukuran symbol (0 Point)

13. Pada proses linking, sebuah simbol/variabel/nama dapat GAGAL di-resolve karena

- a. Simbol tersebut merupakan variabel global, yang didefinisikan pada library yang berbeda, dengan urutan linking sebagai berikut: library yang berisi modul yang mendefinisikan variabel diproses library yang berisi modul yang digunakan program yg mereferensi variabel diproses (0 Point)
- b. Simbol tersebut merupakan variabel global, yang didefinisikan pada file/modul yang berbeda, dengan urutan linking sebagai berikut: file/modul yang mendefinisikan variabel diproses file/modul yang mereferensi variabel diproses (0 Point)

- c. Simbol tersebut merupakan variabel global, yang didefinisikan pada file/modul yang berbeda, dengan urutan linking sebagai berikut: file/modul yang mereferensi variabel diproses file/modul yang mendefinisikan variabel diproses (0 Point)
- d. Simbol tersebut merupakan variabel global, yang didefinisikan pada library yang berbeda, dengan urutan linking sebagai berikut: library yang berisi modul yang digunakan program yang mereferensi variabel diproses library yang berisi modul yang mendefinisikan variabel diproses (2 Point)

Answer/Solution:

-

14. Manakah pernyataan berikut yang TIDAK TEPAT tentang cache
- ukuran cache blok yang besar umumnya akan menyebabkan miss penalty besar (0 Point)
  - ukuran cache blok yang besar umumnya akan menyebabkan miss rate besar (2 Point)
  - semakin besar tingkat cache associativity (e.g. antara two way associative dan four way associative), maka kemungkinan miss rate akan semakin kecil (0 Point)
  - semakin besar tingkat cache associativity (e.g. antara two way associative dan four way associative), akan memperbesar harga/biaya untuk cache tersebut (0 Point)

Answer/Solution:

-

15. manakah yang TIDAK TEPAT tentang instruksi RET yang biasanya dipanggil di akhir sebuah fungsi/subrutin
- mengembalikan return value (2 Point)
  - melakukan implisit POPL pada stack (0 Point)
  - mengubah isi IR (Instruction Register) (0 Point)
  - mengubah isi register %esp (%rsp) (0 Point)

Answer/Solution:

-

16. long A[10]

Jika %rdi menyimpan nilai j (dimana 0=j10) dan %rsi menyimpan alamat A, manakah di bawah ini yang menghitung alamat A[j]?

- leaq(%rdi,%rsi,4),%rax (0 Point)
- leaq(%rdi,%rsi,8),%rax (0 Point)
- leaq(%rsi,%rdi,4),%rax (0 Point)
- leaq(%rsi,%rdi,8),%rax (2 Point)

Answer/Solution:

-

17. int \*A[5];

Pada arsitektur 64-bit, berapa nilai sizeof(A)?

- 5 bytes (0 Point)
- 8 bytes (0 Point)
- 20 bytes (0 Point)
- 40 bytes (2 Point)

Answer/Solution:

-

18. long A[4][6];

Berapa nilai perhitungan alamat untuk A[1][3]?

- A + 72 (2 Point)
- A + 56 (0 Point)
- A + 36 (0 Point)
- A + 28 (0 Point)

Answer/Solution:

-  
19. Suatu kode dieksekusi sbb:

```
mov $0x15213, %rax
pushq %rax
popq %rax
```

Berapakah nilai yang tersimpan pada %rcx jika kemudian dijalankan kode sbb:

```
mov -0x8(%rsp),%rcx
```

- a. 0x0 / NULL (0 Point)
  - b. Seg fault (0 Point)
  - c. Unknown (0 Point)
  - d. 0x15213 (2 Point)
- Answer/Solution:  
-

20. Dari komponen-komponen waktu untuk membaca sector dari sebuah disk drive di bawah ini, manakah yang nilainya akan selalu terkecil pada proses random read?

- a. seek time (0 Point)
- b. rotational latency (0 Point)
- c. transfer time (2 Point)
- d. ketiganya sama cepat (0 Point)

21. Jika sebuah fungsi memiliki 8 argumen, dimanakah argumen tersebut disimpan ketika fungsi tersebut dipanggil pada sistem x86-64?

- a. 8 argumen semuanya disimpan di stack (0 Point)
- b. 4 argumen pertama disimpan di register, 4 argumen berikutnya disimpan di stack (0 Point)
- c. 6 argumen pertama disimpan di register, 2 argumen berikutnya disimpan di stack (2 Point)
- d. 8 argumen semuanya disimpan di register (0 Point)

22. Ketika melakukan pembacaan banyak sector secara sekuensial dari disk, manakah nilai yang terbesar?

- a. seek time (0 Point)
  - b. rotational latency (0 Point)
  - c. transfer time (2 Point)
  - d. ketiganya sama besar (0 Point)
- Answer/Solution:  
-

23. Pada sistem dengan 16-bit address, berapa tag bit yang diperlukan jika E=2, B=8, dan S=4? (E adalah lines/set, B adalah bytes/block, S adalah jumlah set)

- a. 12 (0 Point)
  - b. 10 (0 Point)
  - c. 16 (0 Point)
  - d. 4 (0 Point)
  - e. 11 (2 Point)
- Answer/Solution:  
-

24. Diketahui suatu struktur sbb:

```
struct personalData {
float phone_number;
char bloodGroup[5];};
```

Berapakah nilai keluaran dari kode berikut:

```
printf("%ld\n", sizeof(struct personalData));
```

- a. 12 (2 Point)
- b. 9 (0 Point)
- c. 6 (0 Point)
- d. nilai lain (0 Point)

Answer/Solution:

-

25. Pada sistem dengan 16-bit address, berapa kapasitas total cache (dalam byte) jika E=2, B=8, dan S=4? (E adalah lines/set, B adalah bytes/block, S adalah jumlah set)

- a. 64 (2 Point)
- b. 8 (0 Point)
- c. tidak dapat dihitung (0 Point)
- d. 512 (0 Point)

Answer/Solution:

-

26. Extending the stack can be done by

- a. swapping the base pointer and the stack pointer (0 Point)
- b. subtracting a value from your stack pointer (2 Point)
- c. adding a value to your stack pointer (0 Point)
- d. executing the ret instruction (0 Point)

Answer/Solution:

-

27. How much virtual memory can be addressed by a 32-bit system?

- a. 2GB (0 Point)
- b. 4GB (0 Point)
- c. 8GB (2 Point)
- d. 240TB (0 Point)

Answer/Solution:

-

28. A disk that rotates at 7,200 RPM (8ms per full revolution), has an average seek time of 5ms, and has 1000 sectors per track. How long (approximately) does the average 1-sector access take?

- a. Not enough information to determine the answer (0 Point)
- b. 13ms (0 Point)
- c. 9ms (2 Point)
- d. 10.5ms (0 Point)

Answer/Solution:

-

```
29. struct alfa { //line 1
 char a[17]; //line 2
 int b[6]; //line 3
 long c[3]; //line 4}
```

Struct ini mengandung 3 fields pada mesin x86\_64 dengan alamat awal adalah 0x0000. Tuliskanlah alamat dari a[11], b[3], dan c[2]. Tuliskan dalam hexadecimal, contoh jawaban (tanpa spasi): 0xFFFF,0xFFE,0xFFFFD  
a. 0x000B,0x0020,0x0040 (2 Point)

Answer/Solution:

-

30. A system uses a two-way set-associative cache with 16 sets and 64-byte blocks. Which set does the byte

with the address 0xdeadbeef map to?

- a. Set 7 (0 Point)
  - b. Set 11 (2 Point)
  - c. Set 13 (0 Point)
  - d. Set 14 (0 Point)
- Answer/Solution:  
-

31. Consider the following C variable declaration:

`int *(*f[3])();`

Then f is? (Recall from the book that [] and () have higher precedence than \*)

- a. an array of pointers to pointers to functions that return int (0 Point)
  - b. a pointer to an array of functions that return pointers to int (0 Point)
  - c. a function that returns a pointer to an array of pointers to int (0 Point)
  - d. an array of pointers to functions that return pointers to int (2 Point)
  - e. a pointer to a function that returns an array of pointers to int (0 Point)
  - f. a pointer to an array of pointers to functions that return int (0 Point)
- Answer/Solution:  
-

32. `struct alfa { //line 1  
char a[17]; //line 2  
int b[6]; //line 3  
long c[3]; //line 4}`

Struct ini mengandung 3 fields pada mesin x86\_64 dengan alamat awal adalah 0x0000. Untuk mengoptimasi ukuran memory struct di atas, urutan kode yang tepat adalah: ..., contoh jawaban (tanpa spasi): 1,2,3,4.

- a. 1,4,3,2 (2 Point)

33. `struct alfa { //line 1  
char a[17]; //line 2  
int b[6]; //line 3  
long c[3]; //line 4}`

Struct ini mengandung 3 fields pada mesin x86\_64 dengan alamat awal adalah 0x0000. Tuliskan ukuran dari struct di atas ... byte.

- a. 72 (2 Point)
- Answer/Solution:

34. `int a[n][n], b[n][n];  
int sum = 0;  
for(i=0; in; i++) {  
 for(j=0; jn; j++) {  
 sum += a[i][j]*b[i][j];}}}`

Jika n sangat besar, cache block adalah B dan ukuran cache adalah 8B, maka total miss pada matrix a dan b adalah

- a.  $2n/B$  (0 Point)
- b.  $n^2/B$  (0 Point)
- c.  $2(n^2)/B$  (2 Point)
- d.  $(n^2)/2B$  (0 Point)

40. Berikut ini adalah cara untuk menghindari buffer overflow, kecuali:

- a. Randomisasi offset dari stack (0 Point)
- b. Segmen code yang tidak bisa dieksekusi (0 Point)
- c. Randomisasi segmen code (2 Point)
- d. Menggunakan -fstack-protector pada saat kompilasi GCC (0 Point)

41. Diberikan potongan kode berikut:

0x400446e3 malloc+7: call 0x400446e8 malloc+12

0x400446e8 malloc+12: popl %eax

Setelah instruksi popl selesai, berapakah isi register %eax (dalam hexa)?

- a. 0x400446e8 (2 Point)

42. Sebuah mesin memiliki cache direct mapped berukuran 64K, dengan block size 16 byte, dengan write-back dan write-allocate. Jika variabel lokal menggunakan register, dan isi cache awal kosong, hitunglah miss rate untuk kode berikut:

```
void copy_matrix(int dest[ROWS][COLS], int src[ROWS] [COLS]) {
int i, j;
for (i=0; i<ROWS; i++) {
for (j=0; j<COLS; j++) {
dest[i][j] = src[i][j];}}}
```

Hitunglah cache miss rate jika ROWS = 128 dan COLS = 256 (dalam % dan dibulatkan ke atas (integer), contoh 0%, jawaban: 0)

- a. 100 (2 Point)

43. Pada mesin x86\_64 terdapat kode matrix berikut

```
int x[5][5];
int *ptr[5];
int i, j;
for(i=0; i<5; i++){
 ptr[i] = x[i];
}
```

Berapa bytekah offset alamat x[3][2] dari alamat x?

- a. 64 (2 Point)

44. Pada mesin x86\_64 terdapat kode matrix berikut

```
int x[5][5];
int *ptr[5];
int i, j;
for(i=0; i<5; i++){
 ptr[i] = x[i];
}
```

Berapakah bytekah offset alamat ptr[3] dari alamat ptr?

- a. 24 (2 Point)

45. Konversikan bilangan biner berikut ke dalam bentuk desimal 10101111

- a. 175 (2 Point)

46. Berikut ini adalah bagian-bagian dari hirarki memori: 0: Main Memory (DRAM), 1: L2 Cache, 2: Remote Storage, 3: Register, 4: L1 Cache, 5: Local Storage, 6: L3 Cache. Urutkan hirarki memori dengan akses terlambat hingga tercepat. Contoh jawaban: 0123456

- a. 2506143 (2 Point)

47. Konversikan bilangan hexa 0xABCD ke dalam bentuk biner. Contoh jawaban: 00001111

- a. 101010111100 (2 Point)

48. Berikut ini adalah bagian-bagian yang terdapat di CPU, kecuali

- a. Register (0 Point)

- b. Cache (0 Point)
  - c. Memory (2 Point)
  - d. ALU (0 Point)
  - e. Program Counter (0 Point)
49. manakah perintah berikut yang dapat digunakan untuk menjumlahkan 2 register dan menyimpan hasilnya tanpa menimpa nilai awal kedua register tersebut
- a. mov (0 Point)
  - b. lea (2 Point)
  - c. add (0 Point)
  - d. bukan salah satu di atas (0 Point) Answer/Solution:  
-

50. register mana kah yang digunakan untuk menyimpan parameter pertama pada pemanggilan fungsi

    - a. %edi (0 Point)
    - b. %esi (0 Point)
    - c. %eax (0 Point)
    - d. bukan salah satu di atas (2 Point)

IF2220

Teori Bahasa Formal  
dan Otomata

## UTS TEORI BAHASA FORMAL DAN OTOMATA

UTS SEMESTER II – 2006/2007

Mata kuliah : OTBF (IF2253)  
 Hari/tanggal : Kamis, 29 Maret 2007  
 Waktu : 100 menit (TUTUP BUKU)

- Disain suatu DFA yang menerima bahasa dari simbol  $\{0,1\}$  dengan sifat-sifat: *mempunyai tiga simbol '0' berturutan, tidak harus diakhiri string.*

Tuliskan table transisinya.

- Diberikan  $\epsilon$ -NFA berikut:

|                 | $\epsilon$  | a       | b           | c           |
|-----------------|-------------|---------|-------------|-------------|
| $\rightarrow p$ | $\emptyset$ | $\{p\}$ | $\{q\}$     | $\{r\}$     |
| q               | $\{p\}$     | $\{q\}$ | $\{r\}$     | $\emptyset$ |
| $*r$            | $\{q\}$     | $\{r\}$ | $\emptyset$ | $\{p\}$     |

- Hitunglah  $\epsilon$ -closure dari setiap status.
- Ubahlah automaton diatas menjadi DFA.

- Diberikan table transisi dari DFA sebagai berikut:

|                   | 0     | 1     |
|-------------------|-------|-------|
| $\rightarrow q_1$ | $q_2$ | $q_1$ |
| $q_2$             | $q_3$ | $q_1$ |
| $*q_3$            | $q_3$ | $q_2$ |

- Hitunglah semua ekspresi regular  $R R_{\frac{m}{n}}^{(0)}$ , asumsikan status  $q_i$  adalah status nomor ke- $i$ .
  - Hitunglah semua ekspresi regular  $R R_{\frac{m}{n}}^{(1)}$ , sederhanakan sedapat mungkin.
- Konversikan DFA no 3) menjadi ekspresi regular menggunakan teknik eliminasi status.

IUTS SEMESTER III —  
 TAHUN 2007/2008  
 Nama : OTBF  
 NIM : 0F22531  
 Hari/tanggal : Senin, 24 Maret 2008  
 Waktu : 120 menit (TUTUP BUKU)

---

- Buat sebuah DFA yang menerima bahasa atas alphabet  $\{0,1\}$ , yang terdiri dari himpunan string-string yang berawal dengan '1' dan bila diterjemahkan sebagai bilangan bulat biner merupakan kelipatan 5. Misalkan string 101, 1010, dan 1111 termasuk dalam bahasa tersebut, sedangkan 0, 100, dan 111 tidak termasuk dalam bahasa tersebut.

- Diberikan  $\epsilon$ -NFA berikut:

|                 | $\epsilon$  | a           | b           | c           |
|-----------------|-------------|-------------|-------------|-------------|
| $\rightarrow p$ | {q,r}       | $\emptyset$ | {q}         | {r}         |
| q               | $\emptyset$ | {p}         | {r}         | {p,q}       |
| *r              | $\emptyset$ | $\emptyset$ | $\emptyset$ | $\emptyset$ |

- Hitunglah  $\epsilon$ -closure dari setiap status.
  - Ubahlah automaton diatas menjadi DFA.
- Tentukan ekspresi regular dari himpunan string-string atas alphabet  $\{0,1\}$  yang memuat '0' berjumlah genap.
  - Buktikan bahwa  $\{0^n 1^{2n} / n \geq 1\}$  bukan termasuk bahasa regular.

UNTS SEMESTER II —  
 2008/2009  
 Mata Kuliah : TBQ  
 Hari/tanggal : Rabu, 25 Maret 2009  
 Waktu : 120 menit (TUTUP BUKU)

---

1. Disain suatu DFA yang menerima bahasa dari simbol {0,1}, yang terdiri dari himpunan string-string yang berawal atau berakhir (atau keduanya) dengan substring '01'.

2. Diberikan  $\epsilon$ -NFA berikut:

|                 | $\epsilon$  | a   | b           | c           |
|-----------------|-------------|-----|-------------|-------------|
| $\rightarrow p$ | $\emptyset$ | {p} | {q}         | {r}         |
| q               | {p}         | {q} | {r}         | $\emptyset$ |
| $*r$            | {q}         | {r} | $\emptyset$ | {p}         |

- a) Hitunglah  $\epsilon$ -closure dari setiap status.  
 b) Berikan semua string yang panjangnya tiga atau kurang yang diterima oleh DFA.  
 c) Ubahlah automaton diatas menjadi DFA.
3. Ubahlah DFA berikut menjadi ekspresi regular menggunakan teknik eliminasi status.

|                  | 0 | 1 |
|------------------|---|---|
| $\rightarrow *p$ | s | p |
| q                | p | s |
| r                | r | q |
| s                | q | r |

4. Buktikan bahwa himpunan string-string dari simbol '0' dan '1' yang panjangnya bilangan kuadrat sempurna, BUKAN termasuk bahasa regular.

UJIAN  
SEMESTER II

W

A

K

T

U

:

9

0

M

E

N

I

T

S

I

F

A

T

:

T

U

T

U

P

B

U

K

U

Petunjuk umum :

Ujian ini terdiri atas 5 soal. Kerjakan semua soal pada lembar jawaban. Tiap soal mempunyai bobot nilai yang sama dan total nilai semua benar = 100.

1. Tentukan mesin DFA yang menerima/generate bahasa pada alphabet {0,1} yaitu himpunan *string-string* yang berakhir dengan 00. Jawaban minimal harus terdiri dari:  
 Ide solusi  
 Diagram transisi DFA yang diperoleh  
 Ke 5 komponen-komponen dari DFA tersebut.
2. Jelaskan perbedaan dan kesamaan, keuntungan dan kerugian antara *Deterministic Finite Automata* dan *Nondeterministic Finite Automata*.
3. *Nondeterministic Finite Automata* (NFA)  
 Buat mesin NFA, M dengan symbol masukkan  $\Sigma = \{a,b,c\}$  dan menerima string didalam  $\Sigma^*$  dimana setiap kemunculan symbol b selalu diikuti dengan sebuah symbol a di belakangnya.
  - a. Gambarkan diagram transisi dari soal diatas
  - b. Tuliskan ke-5 komponen dari DFA ini.
4. Buat mesin DFA, M' yang ekivalen dengan M pada soal (3) diatas. Jawaban minimal harus terdiri dari:

Diagram transisi dari DFA dan e-5 komponen DFA ini.

5. FST (Mesin Moore)

Buat mesin Moore M dengan masukkan  $\Sigma = \{0,1\}$  yang menerima string w didalam  $\Sigma^*$  dan akan memberikan keluaran pembagian modulo w terhadap 4.

Jawaban minimal harus terdiri dari :

Diagram transisi dari mesin Moore dan komponen-komponen dari mesin ini.

Mata kuliah : Teori Bahasa dan Otomata (IF2052)

Hari/Tanggal : Rabu, 17 Maret 2010

Waktu : 120 menit (TUTUP BUKU)

1. Buat suatu NFA yang menerima bahasa dari alphabet digit  $\{0,1,\dots,9\}$  yang membentuk string dimana symbol terakhir pernah muncul sebelumnya.

Contoh string yang valid: 012781, 44, 341818, tidak valid 244971, 17.

*Petunjuk* : gunakan status  $q_s$  (status awal)  $q_f$  (status akhir) dan status tambahan  $q_i$ ,  $i = 0\dots 9$  (yang menyatakan NFA telah membaca symbol  $i$ ).

2. Ubahlah DFA berikut menjadi ekspresi regular menggunakan teknik eliminasi status

|                   | 0     | 1     |
|-------------------|-------|-------|
| $\rightarrow q_1$ | $q_2$ | $q_3$ |
| $q_2$             | $q_1$ | $q_3$ |
| $*q_3$            | $q_2$ | $q_1$ |

3. Buktikan bahwa  $\{0^n1^m \mid n \leq m\}$  bukan termasuk bahasa regular.

4. Diberikan table transisi dari sebuah DFA sebagai berikut :

|                 | 0 | 1 |
|-----------------|---|---|
| $\rightarrow A$ | B | A |
| B               | A | C |
| C               | D | B |
| $*D$            | D | A |
| E               | D | F |
| F               | G | E |
| G               | F | G |
| H               | G | D |

- a. Buatkan table status yang tidak ekivalen/ekivalen dari DFA tersebut.  
 b. Konstruksikan DFA yang ekivalen dengan jumlah status minimum.

IUNTS SEMESTER II —  
2010/2011  
IF 2052  
Rabu, 16 Maret 2011  
120 menit

1. Diketahui sebuah  $\epsilon$ -NFA dengan tabel transisi sebagai berikut:

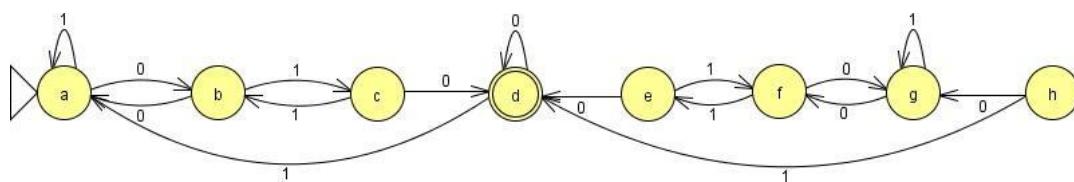
|    | $\epsilon$  | a           | b           | c           |
|----|-------------|-------------|-------------|-------------|
| p  | {q,r}       | $\emptyset$ | {q}         | {r}         |
| q  | $\emptyset$ | {p}         | {r}         | {p,q}       |
| *r | $\emptyset$ | $\emptyset$ | $\emptyset$ | $\emptyset$ |

Ubahlah automata di atas ke dalam bentuk automata sebagai berikut (tuliskan notasi formal lengkapnya):

- a. NFA: tetapkan terlebih dulu ECLOSE untuk setiap state dan tuliskan tabel transisi NFA hasil konversi
  - b. DFA: buat terlebih dulu tabel yang berisi subset construction untuk DFA lalu tetapkan state yang bisa dicapai dari state awal
  - c. Regular Expression: gunakan teknik state elimination
2. Buatlah sebuah NFA yang mampu menerima kumpulan string dari simbol {0,1,2,..,9} dengan ketentuan bahwa simbol terakhir pada string hanya pernah muncul satu kali pada string tersebut. Contoh string yang diterima adalah '2310219', '0001', dst. Yang ditolak '0980', '2300', dst.
3. Buktikan dengan menggunakan pumping lemma bahwa himpunan string dari 0 dan 1 yang mempunyai panjang kuadrat sempurna adalah bukan bahasa regular. Misalnya string dengan panjang 1 atau string dengan panjang 2 seperti 0000,0001, dst merupakan anggota dari himpunan string yang dimaksud.

UTS SEMESTER II – 2011/2012  
 Mata kuliah : Teori Bahasa dan Otomata (IF2052)  
 Hari/tanggal : Kamis, 15 Maret 2012  
 Waktu : 120 menit (TUTUP BUKU)

1. Buatlah sebuah DFA yang mampu menerima himpunan string dari alphabet {0, 1} dengan ketentuan himpunan string yang banyaknya 0 habis dibagi 5 dan banyaknya 1 habis dibagi 3. (Jawaban harus ada diagram transisi DFA)
2. Buktikan dengan menggunakan *pumping lemma* bahwa himpunan string dari 0 dan 1, yang mulai dengan sebuah simbol 1 diinterpretasi sebagai integer, integer tersebut prima adalah bukan bahasa regular.
3. Buatlah sebuah tata bahasa *context-free* yang membangun himpunan semua string dari alphabet {a, b, ., +, \*, (, )} yang terdefinisi dengan baik (*well-formed*) sebagai ekspresi regular.
4. Konstruksi DFA dengan meminimumkan status-status yang ekivalen pada diagram transisi di bawah.



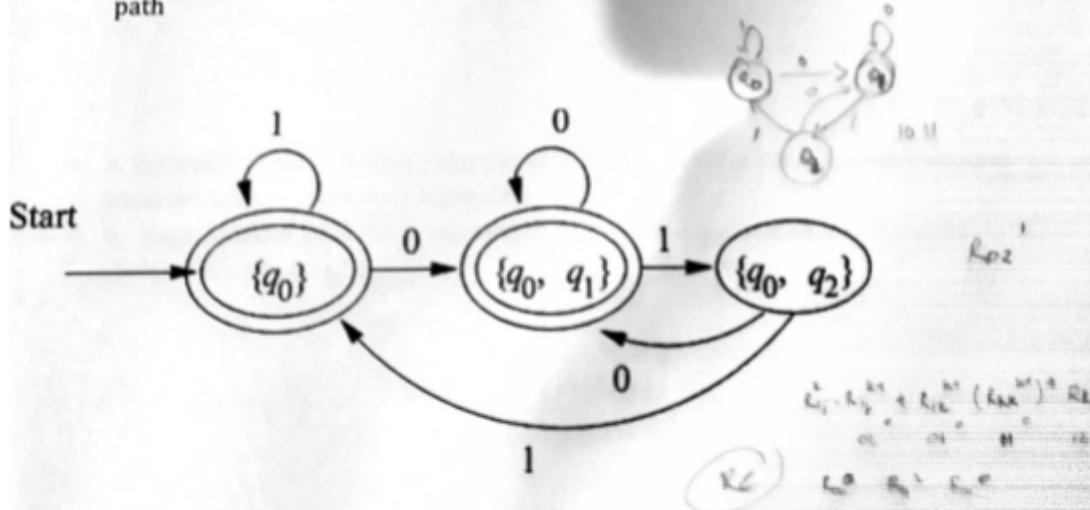
## Ujian Tengah Semester I 2014/2015

Mata kuliah : Teori Bahasa Formal dan Otomata (IF2220)

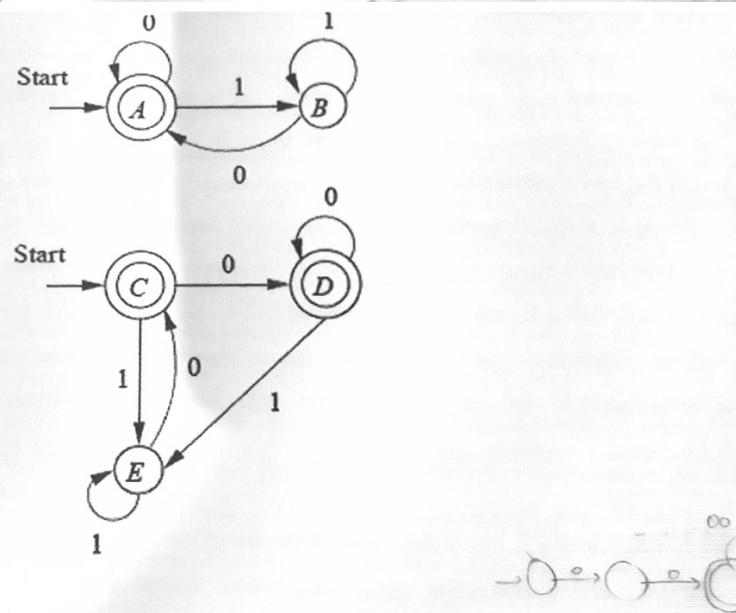
Hari/tanggal : Senin, 20 Oktober 2014

Waktu : 120 menit (TUTUP BUKU)

1. Buktikan dengan menggunakan *pumping lemma* bahwa himpunan string dari 0 dan 1, yang mempunyai panjang kudrat sempurna adalah bukan bahasa regular.
2. Tentukan ekspresi regular dari DFA ini dengan menggunakan konstruksi induksi k-path



3. Tentukan apakah 2 DFA di bawah ini memiliki bahasa yang sama atau tidak, dengan menggunakan teknik **perkalian** 2 DFA (bukan teknik yang menggunakan distinguishable table).



4. a. Rancanglah sebuah DFA yang dapat menerima string yang berakhir dengan "00" yang memiliki jumlah state lebih dari 4.  
 b. Minimumkan DFA hasil rancangan di atas menggunakan **distinguishable table** (Table Filling Algorithm).



UTS SEMESTER I – 2016/2017  
 Mata kuliah: Teori Bahasa Formal dan Otomata (IF2220)  
 Hari/tanggal: Rabu, 12 Oktober 2016  
 Waktu: 100 menit (TUTUP BUKU)

---

- Buat sebuah NFA yang menerima bahasa berikut himpunan string-string dari digit {0,1,2, ,9}, demikian sehingga digit terakhir TIDAK pernah muncul sebelumnya. Contoh: 239757798023761 (digit '1' tidak pernah muncul sebelumnya).

- Diberikan  $\epsilon$ -NFA sebagai berikut:

|                 | $\epsilon$  | a           | b           | c           |
|-----------------|-------------|-------------|-------------|-------------|
| $\rightarrow p$ | {q,r}       | $\emptyset$ | {q}         | {r}         |
| q               | $\emptyset$ | {p}         | {r}         | {p,q}       |
| *r              | $\emptyset$ | $\emptyset$ | $\emptyset$ | $\emptyset$ |

- Hitunglah  $\epsilon$ -closure dari semua statusnya.
- Ubahlah  $\epsilon$ -NFA tersebut menjadi DFA.

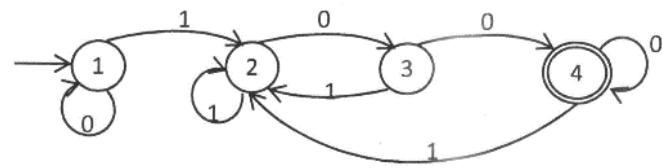
- Konversikan DFA ini menjadi ekspresi reguler menggunakan teknik eliminasi status-status.

|                  | 0 | 1 |
|------------------|---|---|
| $\rightarrow *p$ | s | p |
| q                | p | s |
| r                | r | q |
| s                | q | r |

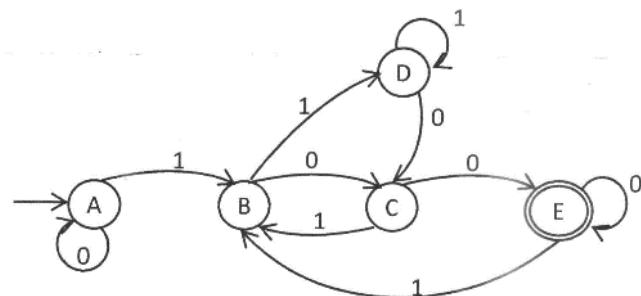
- Buatlah DFA dengan jumlah state yang minimum yang ekuivalen dengan table transisi berikut. Gunakan *table of distinguishable* untuk melakukan minimisasi tersebut.

|                 | 0 | 1 |
|-----------------|---|---|
| $\rightarrow A$ | B | E |
| B               | C | F |
| *C              | D | H |
| D               | E | H |
| E               | F | I |
| *F              | G | B |
| G               | H | B |
| H               | I | C |
| *I              | A | E |

- Nyatakan apakah kedua DFA berikut adalah ekuivalen atau tidak dengan menggunakan teknik *product automaton*. Gambarkan hasil dari *product automaton* kedua DFA di mana state yang ada pada hasil *product automaton* tersebut adalah state yang bisa dicapai dari *start state*. Setelah itu berikan kesimpulannya apakah ekuivalen atau tidak dan sertakan argumennya.



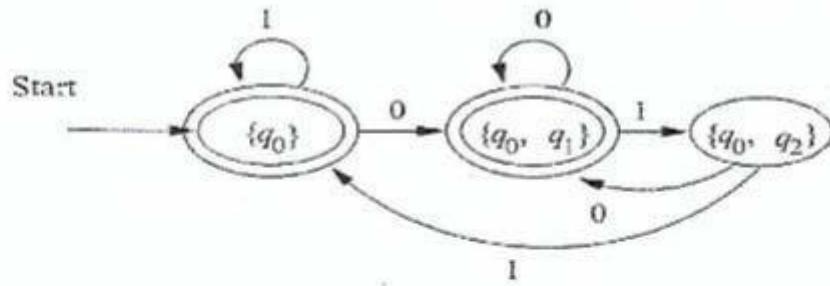
Finite Automata I



Finite Automata II

## UTS SEMESTER I – 2017/2018

- Buktikan dengan menggunakan *pumping lemma* bahwa himpunan string dari 0 dan 1, yang mempunyai panjang (string) kuadrat sempurna adalah bukan bahasa regular.
- Tentukan apakah 2 DFA M dan N ekuivalen. DFA M dinyatakan oleh finite state dan DFA N dinyatakan table transisi. Tuliskan nama Teknik yang Saudara gunakan.



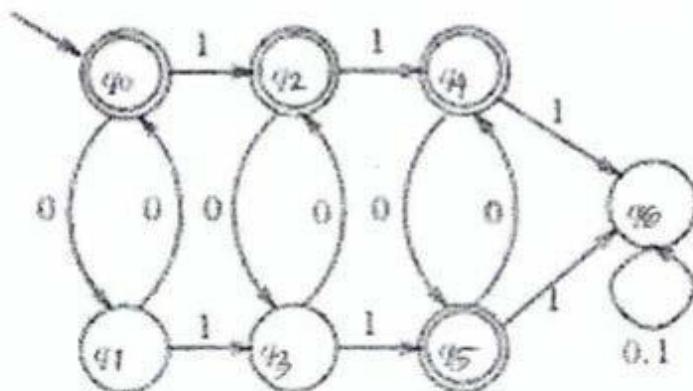
Tabel transisi DFA N.

|                   | 0 | 1 |
|-------------------|---|---|
| $\rightarrow^* A$ | B | A |
| $* B$             | C | A |
| C                 | C | C |

- Ubahlah RE berikut:  $(01^*) + (10)^*$  ke dalam Eps-NFA. Gambarkan diagram dan tabel transisinya. Tuliskan juga *Closure of States* untuk state pada Eps-NFA tsb.
- Buatlah Finite Automata dengan *minimum state* menggunakan Teknik *filling algorithm* (*mengisi distinguishable table*) untuk FA berikut.

|                   | 0 | 1 |
|-------------------|---|---|
| $\rightarrow^* A$ | B | A |
| $* B$             | E | C |
| C                 | B | A |
| D                 | C | D |
| E                 | E | G |
| F                 | E | F |
| G                 | E | F |

5. Buatlah *regular expression* (Alfabet : 0,1) yang akan menerima semua string yang dimulai dengan angka 0 dan mempunyai panjang string ganjil, atau string yang dimulai dengan angka 1 dan mempunyai panjang string genap.
6. Ubahlah fininte-automata di bawah ini menjadi *reguler expression*. Tuliskan nama teknik yang Saudara gunakan.



UTS SEMESTER I – 2019/2020  
IF2124 Teori Bahasa Formal dan Otomata  
Rabu, 9 OKtober 2020  
WAKTU : 100 MENIT

1. Tentukan ekspresi regular untuk Bahasa berikut.
  - a.) Himpunan string-string (0, 1) yang tidak memuat substring 101
  - b.) Himpunan string-string (0, 1) yang banyaknya simbol 0 habis dibagi 3.
2. Buktikan Bahasa L ini bukan Bahasa regular dengan menggunakan Pumping Lemna  
 $L = \text{himpunan string-string dari simbol 1 yang panjangnya adalah bilangan prima.}$
3. Diketahui FA di bawah ini dalam bentuk tabel transisi:

|            | <b>0</b> | <b>1</b> |
|------------|----------|----------|
| <b>→ A</b> | H        | B        |
| <b>B</b>   | H        | A        |
| <b>C</b>   | E        | F        |
| <b>D</b>   | E        | F        |
| <b>E</b>   | F        | G        |
| <b>*F</b>  | F        | F        |
| <b>*G</b>  | G        | F        |
| <b>H</b>   | C        | C        |

Lakukan minimasisasi dari FA di atas dengan menggunakan *table-filling algorithm*, dan gambarkan hasil minimisasi FA nya dalam bentuk diagram transisi

4. DIketahui FA di bawah ini dalam bentuk tabel transisi:

|              | <b>0</b> | <b>1</b> |
|--------------|----------|----------|
| <b>→ *Q0</b> | Q1       | Q2       |
| <b>Q1</b>    | Q0       | Q3       |
| <b>*Q2</b>   | Q3       | Q4       |
| <b>Q3</b>    | Q2       | Q5       |
| <b>*Q4</b>   | Q5       | Q6       |
| <b>*Q5</b>   | Q4       | Q6       |
| <b>Q6</b>    | Q6       | Q6       |

Ubahlah FA di atas ke dalam notasi Regular Expression menggunakan metode *state-elimination technique*.

5. Tuliskan CL (*Closure of state*) untuk setiap *state* pada Eps-NFA di bawah ini. Ubahlah Eps-NFA tersebut ke dalam NFA dan kemudian DFA. Tuliskan NFA dan DFA dalam bentuk tabel transisi.

|            | <b>0</b> | <b>1</b> | <b>Eps</b> |
|------------|----------|----------|------------|
| <b>→ A</b> | -        | B        | B          |
| <b>B</b>   | C        | B        | C          |
| <b>*C</b>  | -        | -        | -          |

6. Tunjukkan menggunakan teknik *Product Automaton*, apakah  $L \subseteq M$ . Tuliskan hasil Product Automaton dalam bentuk *transition diagram*.

Tabel transisi L:

|             | <b>0</b> | <b>1</b> |
|-------------|----------|----------|
| <b>→ *G</b> | H        | G        |
| <b>*H</b>   | H        | I        |
| <b>I</b>    | I        | I        |

Tabel transisi M:

|             | <b>0</b> | <b>1</b> |
|-------------|----------|----------|
| <b>→ *A</b> | B        | D        |
| <b>*B</b>   | B        | C        |
| <b>*C</b>   | F        | C        |
| <b>*D</b>   | E        | D        |
| <b>*E</b>   | E        | F        |
| <b>F</b>    | F        | F        |

--- Selamat ujian ---

**UTS SEMESTER 1 - 2020/2021**  
IF2124 Teori Bahasa Formal dan Otomata  
Metode : Google Form Multiple Choice>Select

**DFA dan NFA**

1. Jesslyn diminta untuk membuat robot berbasis non- Deterministic Finite Automata (NFA) yang menerima string yang diberikan untuk semua laguge yang memiliki string '1001' di dalamnya. Berapa jumlah transisi minimum yang akan digunakan Jesslyn sehingga aplikasi pemrosesan string berfungsi?
  - a. 4
  - b. 5
  - c. 6
  - d. 7**
  - e. 8
  - f. 9
2. Jumlah total states untuk membangun bahasa tertentu menggunakan DFA:  $L = (w \mid w \text{ memiliki tepat } 2 \text{ a dan setidaknya } 2 \text{ b})$ 
  - a. 10**
  - b. 11
  - c. 12
  - d. 13
  - e. 14
  - f. 15
3. NFA memiliki kata 'non-deterministic', apa makna non-deterministic di sini?
  - a. Hasilnya adalah non-deterministik
  - b. Edge input dari current state ke next state adalah non-deterministik**
  - c. Next state adalah non-deterministik
  - d. Jawaban A dan B benar
  - e. Jawaban B dan C benar
  - f. Jawaban A, B, dan C benar
4. Anto diminta untuk mendesain Finite State Machine atau Finite Automata (FA) without output yang menerima language semua kalimat berbahasa Indonesia yang mengandung kata "LULUS". Dengan menyertakan initial state dan final state, berapa jumlah state minimum yang digunakan untuk membangun FA tersebut?
  - a. 3
  - b. 4
  - c. 5
  - d. 6**
  - e. 26
  - f. 27
5. Manakah language yang diterima oleh NFA berikut ini
  - a.  $\{11, 110\}^*\{0\}^*$
  - b.  $\{110, 01\}^*\{1\}^*$**
  - c.  $\{11, 110\}^*\{0\}$**
  - d.  $\{00, 110\}^*\{0\}^*$**
  - e.  $\{00, 110\}^*\{0\}$**
  - f.  $\{11, 101\}^*\{0\}^*$

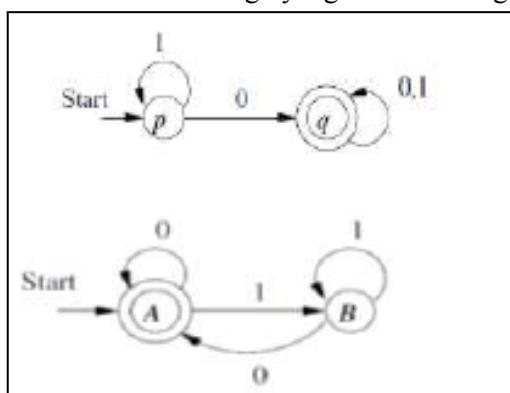
**Reguler Ekspresi**

6. Manakah yang bukan merupakan sub-language dari Finite automata di bawah ini dalam notasi reguler expression ?
  - a.  $0^*(11)^*0^*$
  - b.  $0^*1(10^*1)^*1$
  - c.  $0^*1(10^*1)^*10^*$
  - d.  $0^*1(10^*1)0(100)^*$**
  - e.  $(0^*1(10^*1)^*10^*+0^*)^*$

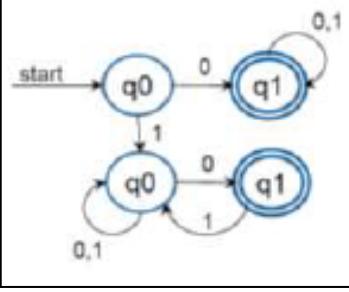
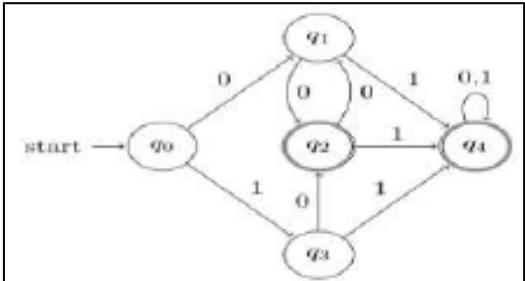
- f. Semua jawaban benar  
g. Semua jawaban salah
7. Diketahui Finite automata di bawah ini. Catatan : Lambda adalah lambang untuk string kosong. Ubahlah ke dalam regular expression.
- $a+b+c$
  - $(a+b+c)^*$
  - $a^*+c+bc$
  - $a^*+c^*+bc$
  - $a^*+c^*+bc^*$
  - $(a+c+bc)^*$
  - $a^*c^*$
  - $a^*+c^*$
  - $(a+c)^*$
  - semua jawaban salah
  - semua jawaban benar
- 
8. Manakah di bawah ini yang bukan termasuk reguler expression ?
- $a^*+c^*$
  - $ac^*bd$
  - $abc + cbd$
  - $\{a b\}^*c$
  - Semua jawaban benar Semua jawaban salah
9. Manakah yang merupakan pernyataan yang salah tentang reguler expression?
- Semua bahasa yang diterima oleh reguler expression bisa dimodelkan dengan DFA juga
  - Semua bahasa yang bisa dimodelkan oleh DFA juga bisa dituliskan ekspresi regulernya.
  - Semua bahasa yang dimodelkan oleh NFA bisa dimodelkan dengan ekspresi reguler
  - Semua bahasa yang dimodelkan oleh NFA with epsilon bisa dimodelkan dengan ekspresi reguler
  - Semua bahasa bisa dimodelkan dengan ekspresi reguler**
  - Semua jawaban benar
  - Semua jawaban salah
10. Sifat manakah yang dipunyai oleh reguler expression?
- annihilator
  - idempotent
  - semua jawaban benar**
  - semua jawaban

### Properti Regular Language

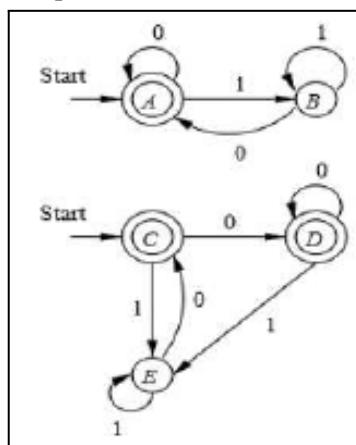
11. Diketahui finite otomata (a),  $L_1$  = menerima semua string2 yang memiliki satu symbol 0 dan finite otomata (b) , $L_2$  = menerima semua string2 yang berakhir dengan symbol 0. Operasi irisan dari kedua otomata  $L_1 \cap L_2$  adalah bahasa



- Menerima semua string2 yang memiliki satu symbol 0.
- Menerima semua string2 yang berakhir dengan symbol 0.**
- Menerima semua string2 yang berawalan dan berakhir dengan symbol 0.

- d. Menerima semua string2 yang berakhir dengan symbol 1.  
e. Menerima semua string2 yang tidak memuat symbol 1.
12. Operasi Reversal (balikan) dari string  $X = X^R$ . Jika  $L = \text{Bahasa regular ditulis dalam notasi ekspresi regular} = (0+1)^*001$  maka hasil operasi reversal  $L$  adalah  
a.  $(0+1)^*100$   
b.  $001(0+1)^*$   
c.  $001(1+0)^*$   
d.  $(1+0)^*001$   
e.  $100(0+1)^*$
13. Operasi komplernen pada Bahasa regular yang dinyatakan oleh Finite Automata di bawah menghasilkan  
  
a. Menerima semua string2 yang diawali satu symbol 0.  
b. Menerima semua string2 yang berakhir satu symbol 0.  
c. Menerima semua string2 yang berawalan dan berakhir dengan symbol 0.  
d. Menerima semua string2 yang diawali dengan symbol 1.  
e. Menerima semua string2 yang memuat  $(0+1)$
14. Diketahui Deterministic Finite Automata (DFA) menerima semua string-string 0 dan 1 yang berakhir 010 =  $L$ .  $L$  dinyatakan dalam bentuk ekspresi regular. Jika  $h$  adalah operasi homomorfisme didefinisikan  $h(0) = a$   $h(1) = ab$  maka  $h(L)$  adalah  
a.  $L((a+ab)^* aaba)$   
b.  $L((ab)^* aaba)$   
c.  $L((a+ab)^*)$   
d.  $L((aba)^*)$   
e.  $L((aab)^* aaba)$
15. Jika  $h$  adalah operasi homomorfisma dari alfabet  $\{0,1,2\}$  ke alfabet  $\{a, b\}$  sebagai berikut  $h(0) = a$   $h(1) = ab$ ,  $h(2) = ba$ .  $L$  adalah Bahasa  $\{ababa\}$ . Operasi  $h^{-1}$  adalah operasi invers homomorfisma maka  $h^{-1}(L)$  adalah  
a.  $L\{110,112,022\}$   
b.  $L\{110,102,112\}$   
c.  $L\{110,102,022\}$   
d.  $L\{012,110,112\}$   
e.  $L\{022,110,112\}$
- Minimalisasi DFA dan Ekuivalensi**
16. Pilihlah jawaban yang paling tepat tentang Pumping Lemma(PL)  
a. PL digunakan untuk membuktikan apakah suatu bahasa adalah reguler  
b. Jika diambil  $w=xyz$  elemen dari  $L$  untuk pemisalan maka  $y$  boleh sama dengan epsilon  
c. Menurut PL jika  $L$  adalah reguler dan  $w$  elemen  $L$  maka  $xz$  bukan elemen  $L$   
d. Pernyataan A,B,C benar semua  
e. Pernyataan A,B,C salah semua
17. Diberikan gambar DFA sebagai berikut, Jika dipartisi menjadi kelas-kelas ekivalen, maka yang **PALING** benar dari jawaban berikut adalah:  
  
a.  $\{q0, q1, q4\}, \{q2, q4\}$   
b.  $\{q0\}, \{q1, q3\}, \{q2, q4\}$   
c.  $\{q0, q1, q3\}, \{q2\}, \{q4\}$   
d.  $\{q0\}, \{q1, q3\}, \{q2\}, \{q4\}$   
e. Tidak ada jawaban yang benar

18. Diberikan dua buah DFA yang ekivalen menurut algoritma TF seperti gambar berikut. Kedua DFA tersebut bila diuji dengan products construction ternyata juga ekivalen. alasan yang PALING tepat DFA tersebut ekivalen adalah DFA hasil produksi :



- Tidak mempunyai pasangan final state [p,q] dimana salah satunya adalah final state.
- Semua transisi dengan input '0' menuju ke final state, sedangkan input '1' ke non-final state
- Ada pasangan state [p,q] dimana salah satunya final state tetapi non-accessible**
- Mempunyai final state [p,q], dimana keduanya merupakan final state.
- Semua jawaban benar

19. Diberikan DFA dalam bentuk tabel untuk menjawab dua soal berikutnya.

|       | 0 | 1 |  |
|-------|---|---|--|
| --> A | B | F |  |
| B     | G | C |  |
| *C    | A | C |  |
| D     | C | G |  |
| E     | H | F |  |
| F     | C | G |  |
| *G    | G | E |  |
| H     | G | C |  |

**Mengisi Tabel Filling: (pilih equivalent atau distinguishable)**

*Mark only one oval per row.*

|       | EQUI                                | DIST                                |
|-------|-------------------------------------|-------------------------------------|
| 1. AE | <input checked="" type="checkbox"/> | <input type="checkbox"/>            |
| 2. AF | <input type="checkbox"/>            | <input checked="" type="checkbox"/> |
| 3. DE | <input type="checkbox"/>            | <input checked="" type="checkbox"/> |
| 4. DF | <input checked="" type="checkbox"/> | <input type="checkbox"/>            |
| 5. HE | <input type="checkbox"/>            | <input checked="" type="checkbox"/> |

20. DFA minimum yang terbentuk pada soal tersebut (sebelumnya) mempunyai state sebanyak :

- 2
- 3
- 4

d. 5

e. 6

## UAS TEORI BAHASA FORMAL DAN OTOMATA

UAS SEMESTER II – 2003/2004  
IF2253 OTOMATA&TEORI BAHASA FORMAL  
WAKTU : 90 MENIT

---

Petunjuk umum: Kerjakan semua soal ini pada lembar jawaban. Ujian ini terdiri dari 4 soal, tiap soal mempunyai nilai yang sama.

1. Desain suatu tata bahasa *Context-free* yang menerima/ membangun bahasa  $\{0^n1^n \mid n \geq 1\}$  yaitu himpunan semua string 0 diikuti 1 dengan jumlah 0 = jumlah 1.
2. Tentukan suatu tata bahasa *Context-free* dengan menghilangkan simbol-simbol tak berguna sehingga tata bahasa ini ekivalen dengan:

$$\begin{array}{ll} S & AB|CA \\ A & a \\ \end{array} \quad \begin{array}{ll} B & BC|AB \\ C & aB|b \end{array}$$

3. Tentukan satu PDA yang ekivalen dengan tata bahasa

$$S \quad aAA \quad \quad \quad A \quad aS|bS|a$$

4. Desain suatu mesin Turing untuk menghitung fungsi integer – (proper minus) yaitu  $m - n = m - n, \quad \text{jika } m \geq n$   
 $m - n = 0, \quad \quad \quad \text{jika } m < n$

Masukan mesin Turing adalah  $0^m10^n$

1. Diketahui tata-bahasa (*grammar*) yang membangkitkan ekspresi *prefix* dengan operand *x* dan *y* dan operator biner +, -, dan \*

$$E \rightarrow EE \mid *EE \mid -EE \mid x \mid y$$

- a) Tentukan turunan *leftmost* dan *rightmost* dari string  $+^*xyxy$
- b) Buktikan bahwa tata-bahasa tersebut tidak ambiguous

2. Ubahlah tata-bahasa berikut ke PDA yang menerima bahasa yang sama dengan stack kosong.

$$S \rightarrow aAA$$

$$A \rightarrow aS \mid bS \mid a$$

3. Diketahui dua bahasa CFL sebagai berikut:

$$L_1 = \{a^i b^k c^{2k} \mid i, k \geq 0\}$$

$$L_2 = \{a^k b^j c^{2k} \mid j, k \geq 0\}$$

- a) Tuliskan tata-bahasa dari  $L_1$  dan  $L_2$ .
- b) Tentukan himpunan  $L_1 \cap L_2$ .
- c) Periksa apakah  $L_1 \cap L_2$  CFL (Petunjuk: gunakan pumping lemma).

ILMU SISTEMESTER III —  
 PAGE 2006/2007  
 Nama : OTIB  
 NIM : 0122531  
 Hari/tanggal : Senin, 28 Mei 2007  
 Waktu : 120 menit  
 Sifat : TUTUP BUKU (boleh satu lembar catatan)

---

- Buktikan dengan menggunakan *pumping lemma*, bahasa yang terdiri dari kumpulan string  $\{0^n1^m \mid n \leq m\}$  bukan merupakan bahasa regular.

- Diberikan table transisi dari sebuah DFA sebagai berikut:

|                 | 0 | 1 |
|-----------------|---|---|
| $\rightarrow A$ | B | A |
| B               | A | C |
| C               | D | B |
| *D              | D | A |
| E               | D | F |
| F               | G | E |
| G               | F | G |
| H               | G | D |

- a) Buatlah tabel status yang tidak ekivalen/ ekivalen dari DFA tersebut.  
b) Konstruksikan DFA lain yang ekivalen dengan jumlah status minimum.
- Disain tata-bahasa bebas konteks (CFG) dari bahasa yang terdiri dari kumpulan string  $\{0^n1^n \mid n \geq 0\}$ .
- Diberikan PDA  $P = (\{q,p\}, \{0,1\}, \{Z_0, X\}, \delta, q, Z_0, \{p\})$  dengan fungsi transisi didefinisikan sebagai berikut:

- $\delta(q, 0, Z_0) = \{(q, XZ_0)\}$
- $\delta(q, 0, X) = \{(q, XX)\}$
- $\delta(q, 1, X) = \{(q, X)\}$
- $\delta(q, \epsilon, X) = \{(p, \epsilon)\}$
- $\delta(p, \epsilon, X) = \{(p, \epsilon)\}$
- $\delta(p, 1, X) = \{(p, XX)\}$
- $\delta(p, 1, Z_0) = \{(p, \epsilon)\}$

Mulai dari ID awal ( $q, 0101, Z_0$ ), tunjukkan semua ID yang dapat dicapai dari ID tersebut.

- Ubahlah PDA *null stack* yang memproses urutan *if (i)* dan *else (e)* pada bahasa pemrograman,  $P_N = (\{q\}, \{i, e\}, \{Z\}, \delta_N, q, Z)$  menjadi tata-bahasa (*grammar*) bebas konteks (CFG), dimana  $\delta_N$  didefinisikan sebagai berikut:

- $\delta_N(q, i, Z) = \{(q, ZZ)\}$
- $\delta_N(q, \epsilon, Z) = \{(q, \epsilon)\}$

ILMU SISTEMESTER III —  
 2007/2008  
 Diklatasi : OTBIF  
 Nomor : 0F22531  
 Hari/tanggal : Senin, 19 Mei 2008  
 Waktu : 100 menit  
 Sifat : TUTUP BUKU (boleh satu lembar catatan)

---

1. Diberikan table transisi dari sebuah DFA sebagai berikut:

|    | 0 | 1 |
|----|---|---|
| →A | B | E |
| B  | C | F |
| *C | D | H |
| D  | E | H |
| E  | F | I |
| *F | G | B |
| G  | H | B |
| H  | I | C |
| *I | A | E |

- a) Buatkan tabel status yang tidak ekivalen/ ekivalen dari DFA tersebut.  
 b) Konstruksi DFA yang ekivalen dengan jumlah status minimum.
2. Disain  
 3. tata-bahasa bebas konteks (CFG) dari bahasa yang terdiri dari kumpulan string  $\{0^n1^m \mid n > m \geq 0\}$ . Contoh string yang valid adalah : 0, 00, 01, 000, 0001, 00011, dst.
4. Diberikan PDA  $P = (\{q,p\}, \{0,1\}, \{Z_0, X\}, \delta, q, Z_0, \{p\})$  dengan fungsi transisi didefinisikan sebagai berikut:

1.  $\delta(q, 0, Z_0) = \{(q, XZ_0)\}$
2.  $\delta(q, 0, X) = \{(q, XX)\}$
3.  $\delta(q, 1, X) = \{(q, X)\}$
4.  $\delta(q, \epsilon, X) = \{(p, \epsilon)\}$
5.  $\delta(p, \epsilon, X) = \{(p, \epsilon)\}$
6.  $\delta(p, 1, X) = \{(p, XX)\}$
7.  $\delta(p, 1, Z_0) = \{(p, \epsilon)\}$

Mulai dari ID awal ( $q, 0011, Z_0$ ), tunjukkan semua ID yang dapat dicapai dari ID tersebut.

5. Ubahlah tata-bahasa berikut ke PDA yang menerima bahasa dengan stak kosong.

$$\begin{array}{ll} S & aAA \\ A & aS \mid bS \mid a \end{array}$$

Mata kuliah : TBO (IF2052)  
Hari/tanggal : Rabu, 27 Mei  
2009 Waktu : 100 menit  
Sifat : TUTUP BUKU (boleh satu lembar catatan)

---

1. Disain tata-bahasa bebas konteks yang menerima bahasa:

$$L = \{(ab)^i aa \mid i = 0, 1, \dots\}$$

2. Diberikan tata-bahasa sebagai berikut:

$$S \quad aS \mid aSbS \mid \epsilon$$

Tata-bahasa tersebut adalah *ambiguous*. Tunjukkan dalam kasus tertentu string ‘aab’ mempunyai dua *turunan paling-kiri (leftmost)* dan gambarkan pohon parsingnya.

3. Disain PDA (dengan *stack* kosong ATAU *final state*) yang menerima bahasa dari string ‘0’ dan ‘1’ dengan jumlah yang sama
4. Ubahlah ekspresi tata-bahasa berikut menjadi PDA:

$$\begin{array}{l} I \quad a \mid b \mid Ia \mid Ib \mid I0 \mid I1 \\ E \quad I \mid E^*E \mid E+E \mid (E) \end{array}$$

Mata kuliah : OTBF (IF2052)  
Hari/Tanggal : Selasa, 18 Mei 2010  
Waktu : 100 menit  
Sifat : TUTUP BUKU (boleh satu lembar catatan)

1. Disain tata-bahasa bebas konteks yang menerima bahasa :

$$L = \{a^i b \mid i = 2, 3, \dots\}$$

2. Ubahlah tata-bahasa berikut ke PDA dengan stack kosong yang menerima bahasa dari tata-bahasa berikut :

$$S \quad 0S1 \mid A$$

$$A \quad 1A0 \mid S \mid \epsilon$$

3. Diberikan PDA  $P = (\{p,q\}, \{0,1\}, \{Z_0, X\}, \delta, q, Z_0, \{p\})$  dengan fungsi transisi didefinisikan sebagai berikut :

$$1. \delta(q, 0, Z_0) = \{(q, X Z_0)\} \quad 5. \delta(p, \epsilon, X) = \{(p, \epsilon)\}$$

$$2. \delta(q, 0, X) = \{(q, X X)\} \quad 6. \delta(p, 1, X) = \{(p, X X)\}$$

$$3. \delta(q, 1, X) = \{(q, X)\} \quad 7. \delta(p, 1, Z_0) = \{(p, \epsilon)\}$$

$$4. \delta(q, \epsilon, X) = \{(p, \epsilon)\}$$

Konversikan PDA ini ke dalam suatu tata-bahasa bebas konteks.

4. Diberikan tata-bahasa

berikut :  $S \quad AAA \mid B$

$$A \quad aA \mid B$$

$$B \quad \epsilon$$

a. Eliminasi produksi- $\epsilon$

b. Eliminasi produksi unit dari hasil (a)

c. Eliminasi symbol-simbol yang tidak berguna dari hasil

(b) Ubahlah tata-bahasa tersebut ke dalam bentuk CNF dari hasil (c)

1. Buatlah sebuah CFG yang mampu menerima masukan kalimat bahasa Indonesia sederhana yang berpolai:

- Subjek presikat-transitif objek
- Subjek predikat-intransitif

Simbol terminal yang diterima adalah {Adi, Bayu, membuat, tertawa, program}. Tentukan simbol non terminal, simbol non terminal awal, serta aturan produksi yang ada sehingga kalimat-kalimat berikut dapat diterima: "Adi membuat program", "Bayu membuat program", "Adi tertawa", "Bayu tertawa". Dan kalimat selainnya ditolak.

2. Buatlah sebuah PDA yang mampu menerima sekumpulan string dengan jumlah angka '1' adalah 2 kali jumlah angka '0'.

3. Ubahlah tata bahasa berikut ke suatu PDA yang menerima bahasa ini oleh stack kosong

$$S \rightarrow aAA$$

$$A \rightarrow aS \mid bS \mid a$$

1. Buatlah sebuah PDA diterima oleh status akhir yang mampu menerima himpunan string dari alfabet  $\{0, 1\}$ : dengan ketentuan himpunan string yang banyaknya 0 sama dengan yang banyaknya 1 (jawaban harus ada dengan transisi PDA).
2. Konversikan PDA diterima oleh status akhir pada soal (1) ke PDA diterima stak kosong (jawaban harus ada 6 komponen PDA stak kosong).
3. Ubahlah tata bahasa berikut ke suatu PDA yang menerima bahasa ini oleh stak kosong.  
S    1AB  
A    0A|0  
B    0B|1B|1
4. Diberikan tata bahasa berikut:  
S    0A0|1B1|AB  
A    C  
B    S|A  
C    S| $\epsilon$ 
  - a. Eliminasi produksi- $\epsilon$
  - b. Eliminasi produksi unit dari hasil (a).
  - c. Eliminasi symbol-simbol yang tidak bermanfaat dari hasil (b).
  - d. Ubahlah tata bahasa tersebut ke dalam bentuk CNF dari hasil (c).

Mata kuliah : Teori Bahasa dan Otomata (IF2052)

Hari/tanggal : Selasa, 14 Mei 2013

Waktu : 120 menit (TUTUP BUKU)

1. Buatlah sebuah PDA *final state* yang mampu menerima himpunan string dari alphabet {0,1} dengan syarat banyaknya 0 dua kali banyaknya 1 (jawaban harus berupa diagram transisi PDA).
2. Buatlah suatu tata bahasa bebas konteks (CFG) yang membentuk bahasa L berikut.

$$L = \{a^n b^{2n} c \mid n \geq 1\}$$

3. Ubahlah tata bahasa berikut ke suatu PDA *empty stack*.

|   |             |
|---|-------------|
| S | AS   A      |
| A | 0A   1B   1 |
| B | 0B   0      |

4. Diberikan tata bahasa G berikut dalam bentuk CNF. Lakukan tes keanggotaan dalam bahasa L(G) untuk string “ababa” dengan algoritma CYK.

|   |         |
|---|---------|
| S | AB   BC |
| A | BA   a  |
| B | CC   b  |
| C | AB   a  |

Ujian Akhir Semester I 2014/2015  
Mata kuliah : Teori Bahasa Formal dan Otomata (IF2220)  
Hari/tanggal : Selasa, 16 Desember 2014  
Waktu : 120 menit (TUTUP BUKU)

12.30 -

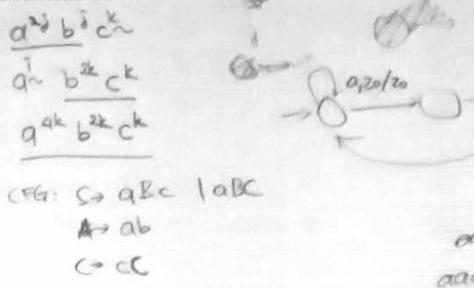
1. Buatlah sebuah PDA (Push Down Automata) yang mampu menerima bahasa sbb:  
 $\{a^i b^j c^k \mid i=2j \text{ or } j=2k\}$

$$i = 2j = 4k \\ i, j, k \neq 0$$

2. Konversikan PDA berikut ke dalam CFG (Context Free Grammar):

$P = (\{p, q\}, \{0, 1\}, \{Z_0, X\}, \delta, q, Z_0, \{p\})$  dgn fungsi transisi sbb:

1.  $\delta(q, 0, Z_0) = \{(q, XZ_0)\} \rightarrow \text{POP PUSH}$
2.  $\delta(q, 0, X) = \{(q, XX)\} \rightarrow \text{POP PUSH}$
3.  $\delta(q, 1, X) = \{(q, X)\} \rightarrow \text{PUSH}$
4.  $\delta(q, \epsilon, X) = \{(p, \epsilon)\} \rightarrow \text{POP}$
5.  $\delta(p, \epsilon, X) = \{(p, \epsilon)\} \rightarrow \text{POP}$
6.  $\delta(p, 1, X) = \{(p, XX)\} \rightarrow \text{PUSH}$
7.  $\delta(p, 1, Z_0) = \{(p, \epsilon)\} \rightarrow \text{POP}$



3. Diketahui CFG  $G = (\{A, S\}, \{a, b\}, P, S)$  dgn aturan produksi P sbb:

$$\begin{aligned} S &\rightarrow aAA \\ A &\rightarrow aS \mid bS \mid a \mid C \\ C &\rightarrow b \end{aligned}$$

- a. Konversikan CFG di atas ke dalam bentuk PDA  
 b. Konversikan CFG di atas ke dalam bentuk CNF (Chomsky Normal Form)

4. Diberikan tata bahasa  $G$  berikut. Gunakan algoritma CYK untuk memeriksa apakah string ababaa anggota Bahasa  $L(G)$ .

$$\begin{array}{ll} S \rightarrow AB|BC & AB \rightarrow S, C \\ A \rightarrow BA|a & BA \rightarrow A \\ B \rightarrow CC|b & BC \rightarrow S \\ C \rightarrow AB|a & CC \rightarrow B \end{array}$$

5. Desain suatu mesin Turing yang menerima Bahasa  
 $\{ww^R \mid w \text{ adalah suatu string anggota } (0+1)^*\}$

Mata kuliah: Teori Bahasa Formal dan Otomata

Hari/tanggal: Kamis, 10 Desember 2015

Waktu: 120 menit (TUTUP BUKU)

1. Diberikan tata bahasa G berikut. Gunakan algoritma CYK untuk memeriksa apakah string  $dddbbbdbbb$  anggota bahasa  $L(G)$ .

$$\begin{aligned} SS &\rightarrow AAAA \mid AAAA \\ AA &\rightarrow AAAA \mid dd \\ AA &\rightarrow AAAA \mid bb \\ AA &\rightarrow AAAA \mid dd \end{aligned}$$

2. Diberikan tata bahasa G berikut. Tentukan bahasa yang dibangun oleh G ini,  $L(G)$ .

$$\begin{aligned} SS &\rightarrow ddbbSSssAA \mid \epsilon\epsilon \\ AA &\rightarrow bbAA \mid bb \end{aligned}$$

3. (a) Buatlah PDA dengan *empty stack* yang menerima *binary string* dengan jumlah 0 dan 1 yang sama.  
(b) Ubahlah ke CFG dengan aturan konversi seperti telah dijelaskan di kelas.
4. (a) Buatlah CFG untuk persoalan berikut.  
$$L(GG) = \{dd^{nn}bb^{mm} \mid 0 \leq dd \leq dd \leq 3dd\}$$
  
(b) Ubahlah ke PDA *final state* (setelah sebelumnya diubah ke PDA *empty stack*).  
(c) Ubahlah CFG di atas ke bentuk CNF.

**UJIAN AKHIR SEMESTER I 2016/2017**

Mata kuliah : TBFO (IF2220)  
Hari/tanggal : Selasa, 06 Desember 2016  
Waktu : 120 menit  
Sifat : tutup buku

- 
1. Buat suatu tata-bahasa bebas konteks yang menerima bahasa dari simbol '0' dan '1' dengan jumlah '0' dan '1' SAMA, dimana posisi dari '0' dan '1' sebarang. Contoh: 00101110; 0011; 010101101010; 111000 dsb.
  2. Diberikan tata bahasa bebas konteks berikut:

$$\begin{aligned}S &\rightarrow A1B \\A &\rightarrow 0A \mid \epsilon \\B &\rightarrow 0B \mid 1B \mid \epsilon\end{aligned}$$

Tuliskan langkah-langkah penurunan *leftmost* dan *rightmost* untuk string '00011'. Kemudian masing-masing gambar kan pohon parsingnya.

3. Diberikan tata-bahasa G berikut. Gunakan algoritma CYK untuk memeriksa apakah string baaab anggota Bahasa  $L(G)$ .

$$\begin{aligned}S &\rightarrow AB \mid BC \\A &\rightarrow BA \mid a \\B &\rightarrow CC \mid b \\C &\rightarrow AB \mid a\end{aligned}$$

4. Diberikan tata-bahasa berikut:

$$\begin{aligned}S &\rightarrow ASB \mid \epsilon \\A &\rightarrow aAS \mid a \\B &\rightarrow SbS \mid A \mid bb\end{aligned}$$

- a). Eliminasi produksi- $\epsilon$
- b). Eliminasi produksi unit dari hasil (a)
- c). Eliminasi simbol-simbol yang tidak berguna dari hasil (b)
- d). Ubahlah tata-bahasa tersebut ke dalam bentuk CNF dari hasil (c)

5. Buatlah PDA yang dapat menerima bahasa berikut:  
 $L = \{a^{n+1}b^{2n+m}c^m : n \geq 0, m \geq 1\}$ . Catatan: PDA yang dibuat, harus dirancang langsung ke bentuk PDA, tidak boleh membuat PDA dengan merancang CFG terlebih dahulu.
6. Ubahlah PDA berikut menjadi CFG.  
PDA  $P = (\{q_0, q_1\}, \{0, 1\}, \{0, 1, Z_0\}, \delta, q_0, Z_0, \{q_1\})$ , dengan fungsi transisi sebagai berikut:

$$\begin{aligned}\delta(q_0, 0, Z_0) &= \{(q_0, 0Z_0)\} \\ \delta(q_0, 1, Z_0) &= \{(q_0, 1Z_0)\} \\ \delta(q_0, 0, 0) &= \{(q_0, 00)\} \\ \delta(q_0, 0, 1) &= \{(q_0, \epsilon)\} \\ \delta(q_0, 1, 0) &= \{(q_0, \epsilon)\} \\ \delta(q_0, 1, 1) &= \{(q_0, 11)\} \\ \delta(q_0, \epsilon, Z_0) &= \{(q_1, Z_0)\}\end{aligned}$$

Catatan: Sebelum mengubah ke CFG, PDA final state perlu diubah dulu ke PDA empty stack.

## UAS SEMESTER I – 2017/2018

1. Diberikan tata bahasa G berikut. Gunakan algoritma CYK untuk memeriksa apakah ekspresi  $a + a^*$  anggota Bahasa  $L(G)$ .

$$E \rightarrow E + T/T$$

$$T \rightarrow F|T * F$$

$$F \rightarrow E|a$$

2. Diketahui suatu grammar  $G = (\{STMT, EXPR, OTHER\}, \{if, then, else\}, Prod, STMT)$  dalam suatu bahasa pemrograman dengan produksi Prod sebagai berikut :

$$STMT \rightarrow if\ EXPR\ then\ STMT$$

$$STMT \rightarrow if\ EXPR\ then\ STMT\ else\ STMT$$

$$STMT \rightarrow OTHER$$

- Periksalah apakah grammar di atas termasuk grammar ambigu atau bukan ? Buktikanlah!
- Jika jawaban anda di nomor a) tidak ambigu, maka kosongkanlah bagian ini. Jika jawaban anda di nomor a) tersebut ambigu maka buatlah grammar yang ekuivalen dengan grammar I atas (mendeskripsikan bahasa yang sama) tetapi tidak ambigu.

3. Diketahui suatu grammar  $G = (\{STMT, OPTION_LIST, OPTION, MODE, SCALE, PRECISION, BASE\}, \{declare\_id, real, complex, fixed, floating, single, double, binary, decimal\}, Prod, STMT)$ , dengan produksi Prod sebagai berikut :

$$STMT \rightarrow declare\_id\ OPTION\_LIST$$

$$OPTION\_LIST \rightarrow OPTION\_LIST\ OPTION\ |\ \epsilon$$

$$MODE \rightarrow real\ |\ complex$$

$$SCALE \rightarrow fixed\ |\ floating$$

$$PRECISION \rightarrow single\ |\ double$$

$$BASE \rightarrow binary\ |\ decimal$$

Ubahlah grammar di atas ke dalam CNF

4. Rancanglah sebuah CFG yang mampu menerima bahasa berikut.

a.  $L = \{w \mid \text{dimana } |w| \text{ adalah ganjil dan symbol di tengahnya adalah } \emptyset\}$

b.  $L = \{a^{2n}b^mc^nd^n \mid n \geq 0, m \geq 0\}$

c.  $L = \{\text{string yang terdiri atas symbol 0 dan 1, dan tidak berupa "ww" dimana } w \text{ adalah string}\}$

5. Rancanglah sebuah PDA dengan *final state* yang mampu menerima bahasa berikut. Catatan: (1) Jangan gunakan teknik yang membuat CFG terlebih dahulu dan kemudian mengubahnya ke dalam PDA; (2) aksi “push” untuk stack hanya bias menambah satu input

a.  $L = \{a^{n+1}b^mc^{n-m} \mid n \geq 0, m \geq 0, n > m\}$

b.  $L = \{w \mid w \text{ memiliki banyaknya symbol 0 = 2x banyaknya symbol 1}\}$

c.  $L = \{a^i b^j c^k \mid i \neq j \text{ or } j \neq k\}$

**UAS SEMESTER I – 2019/2020**

Kamis, 12 Desember 2019

120 Menit

1. Diberikan tata bahasa  $G = (\{E, T, F\}, \{a, +, *, (, )\}, \text{Produksi}, E)$ . Aturan produksi dibawah ini. Konversikan tata bahasa ini ke PDA stack kosong.

$$E \rightarrow E + T \mid T$$

$$T \rightarrow F \mid T * F$$

$$F \rightarrow (E) \mid a$$

2. Diberikan tata bahasa  $G$  berikut. Gunakan algoritma CYK untuk memeriksa apakah string  $(( ))()$  anggota Bahasa  $L(G)$ .

$$B \rightarrow BR \mid ($$

$$R \rightarrow RR \mid ()$$

3. Buatlah CFG untuk persoalan berikut.

a.  $L(G) = \{a^{n+2}b^m c^{n-m} : n > 0, m \geq 0\}$

b.  $L(G) = \{a^n b^m : 0 \leq n \leq m \leq 3n\}$

c.  $L(G) = \{w\#x \mid w^R$  (reverse dari  $w$ ) adalah substring dari  $x$ , dimana  $w, x \in \{a, b\}^*$ , contoh string diterima adalah:  $ba\#aaabbb$  dimana  $w$  adalah "ab" dan  $x$  adalah "aaabbb"

4. Ubahlah CFG  $G$  berikut ke dalam CNF dengan langkah-langkah eliminasi produksi- $\epsilon$ , eliminasi produksi unit dan eliminasi simbol-simbol yang tidak bermanfaat.

$G = (\{A, S\}, \{a, b\}, P, S)$  dgn aturan produksi  $P$  sbb:

$$S \rightarrow AAa$$

$$A \rightarrow aS \mid bS \mid a \mid C$$

$$C \rightarrow b \mid \epsilon$$

$$D \rightarrow AC \mid SA$$

5. Buatlah PDA yang akan menerima bahasa :

$$L(G) = \{a^{m+2}b^{n+4}c^nd^m : n \geq 0, m \geq 0\}$$

Jawaban anda harus berisi deskripsi state-statenya, dan algoritma/strategi untuk pembuatan PDA nya. Jika deskripsi state dan algoritma/strateginya tidak jelas, maka walaupun PDA anda benar maka nilainya akan 0 besar.

6. Konversikanlah PDA di bawah ini menjadi sebuah CFG.

1.  $Q = \{q_1, q_2\}$
2.  $\Sigma = \{0,1,2\}$
3.  $\Gamma = \{Z, B, G\}$
4.  $S = q_1$
5.  $F = \emptyset$
6.  $Z = Z$

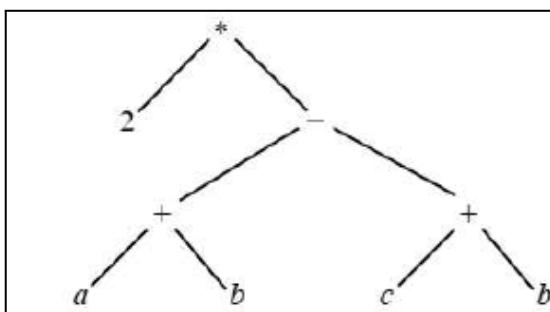
Memiliki fungsi transisi:

1.  $\Delta(q_1, 0, Z) = \{(q_1, BZ)\}$
2.  $\Delta(q_1, 0, B) = \{(q_1, BB)\}$
3.  $\Delta(q_1, 0, G) = \{(q_1, BG)\}$
4.  $\Delta(q_1, 2, Z) = \{(q_2, Z)\}$
5.  $\Delta(q_1, 2, B) = \{(q_2, B)\}$
6.  $\Delta(q_1, 2, G) = \{(q_2, G)\}$
7.  $\Delta(q_2, 0, B) = \{(q_2, \epsilon)\}$
8.  $\Delta(q_2, \epsilon, Z) = \{(q_2, \epsilon)\}$
9.  $\Delta(q_1, 1, Z) = \{(q_1, GZ)\}$
10.  $\Delta(q_1, 1, B) = \{(q_1, GB)\}$
11.  $\Delta(q_1, 1, G) = \{(q_1, GG)\}$
12.  $\Delta(q_2, 1, G) = \{(q_2, \epsilon)\}$

## UAS TBFO 2020/2021

### CFG dan CFL

1. Ekspresi aritmatika yang berkoresponden langsung dengan parse tree dibawah adalah



- a.  $2(a-c)$
- b.  $2a-2c$
- c.  $2(a+b-c+b)$
- d.  $2((a+b)-(c+b))$**
- e.  $2(a+b)-2(c+b)$

2. Tata Bahasa Context-free pada himpunan Terminal = {a, b} mempunyai aturan produksi berikut ini. Bahasa L yang dihasilkan oleh tata bahasa adalah

- a.  **$L = \{(ab)^i aa \mid i=0, 1, 2, \dots\}$**
- b.  $L = \{a^j b^i a^j \mid i=0, 1, 2, \dots \text{ dan } j=1, 2, \dots\}$
- c.  $L = \{a^i b^j a^j \mid i=0, 1, 2, \dots \text{ dan } j=1, 2, \dots\}$
- d.  $L = \{a^i baa \mid i=0, 1, 2, \dots\}$
- e.  $L = \{a^i b \mid i=0, 1, 2, \dots\}$

$$\begin{array}{l} S \rightarrow abS | aA \\ A \rightarrow a \end{array}$$

3. Tata Bahasa Context-free pada himpunan Terminal = {0, 1} mempunyai aturan produksi berikut ini. Penurunan paling kiri atau paling kanan pada tata bahasa di atas menghasilkan kalimat/ string terminal

- a. Semua string2 terminal yang diperoleh berakhir satu symbol 0.

$$\begin{array}{l} S \rightarrow AB \\ A \rightarrow 0A1|01 \\ B \rightarrow 11A \end{array}$$

- b. Semua string2 terminal yang diperoleh berakhir satu symbol 01.
- c. Semua string2 terminal yang diperoleh berawalan dengan symbol 01.
- d. Semua string2 terminal yang diperoleh diawali dengan symbol 1.
- e. Semua string2 terminal yang diperoleh memuat banyak 0 dan 1 sama.
- f. Tidak ada jawaban yang benar**

4. Tata Bahasa Context-free pada himpunan Terminal = {a,b} mempunyai aturan produksi berikut :  $S \rightarrow SS \mid a \mid b$ . Pernyataan yang benar adalah

- a. Tata bahasa ini tidak ambigu.
- b. Semua string2 terminal yang diperoleh mempunyai hanya satu penurunan.
- c. Tata Bahasa ini ambigu karena semua string2 terminal yang diperoleh mempunyai beberapa pohon parsing.
- d. Semua string2 terminal yang diperoleh diawali dengan symbol a.
- e. Tata Bahasa ini ambigu karena ada string2 terminal yang diperoleh mempunyai beberapa pohon parsing.**

5. Tata Bahasa Context-free pada himpunan Terminal = {0, 1} mempunyai aturan produksi berikut ini. String terminal yang diperoleh dari tata bahasa dengan panjang string 5 adalah

- a. 00011
- b. 00110
- c. 00101
- d. 00111
- e. 01101

$$\begin{array}{l} S \rightarrow AOB \\ A \rightarrow BB \mid 0 \\ B \rightarrow AA \mid 1 \end{array}$$

### PDA / DPDA

6. Konversi PDA  $P = (\{p,q\}, \{0,1\}, \{X, Z_0\}, 6, q, Z_0)$  ke CFG (Context Free Grammar) dilakukan dengan memadankan setiap fungsi transisi dengan aturan produksi tertentu. Diantara dua fungsi transisinya adalah :  $6(q, 1, X) = \{(q, XX)\}$  dan  $6(p, 1, X) = \{(p, c)\}$ . Pilihlah aturan produksi berikut yang valid yang merupakan terjemahan dari kedua fungsi tersebut

- a.  $[qxq] \rightarrow 1[qxp][pxq] \text{ dan } [pxp] \rightarrow 1$**
- b.  $[qxq] \rightarrow [qxp][pxq] \text{ dan } [pxp] \rightarrow 1$
- c.  $[qxq] \rightarrow 1[qxp][pxq] \text{ dan } [pxp] \rightarrow \epsilon$
- d.  $[qxq] \rightarrow 1[qxp][pxq] \text{ dan } [qxq] \rightarrow 1$

- e.  $[qxq] \rightarrow 1[qxp][qxq]$  dan  $[qxq] \rightarrow 1$
7. Diberikan PDA  $P = (\{q_0, q_1, q_2, q_3, f\}, \{a, b\}, \{Z_0, A, B\}, \delta, q_0, Z_0, \{f\})$  dengan fungsi transisi 6 seperti dibawah. Salah satu dari rangkaian transisi yang valid jika diberikan input  $w=abb$  adalah: \*
- $(q_1, bb, AZ_0)$
  - $(q_0, \epsilon, AZ_0)$
  - $(q_1, \epsilon, AA)$
  - $(q_0, \epsilon, Z_0)$**
  - $(q_3, \epsilon, epsilon)$
  - Tidak ada jawaban benar
- |                                              |                                            |                                              |
|----------------------------------------------|--------------------------------------------|----------------------------------------------|
| $\delta(q_0, a, Z_0) = (q_1, AAZ_0)$         | $\delta(q_0, b, Z_0) = (q_2, BZ_0)$        | $\delta(q_0, \epsilon, Z_0) = (f, \epsilon)$ |
| $\delta(q_1, a, A) = (q_1, AAA)$             | $\delta(q_1, b, A) = (q_1, \epsilon)$      | $\delta(q_1, \epsilon, Z_0) = (q_0, Z_0)$    |
| $\delta(q_2, a, B) = (q_3, \epsilon)$        | $\delta(q_2, b, B) = (q_2, BB)$            | $\delta(q_2, \epsilon, Z_0) = (q_0, Z_0)$    |
| $\delta(q_3, \epsilon, B) = (q_2, \epsilon)$ | $\delta(q_3, \epsilon, Z_0) = (q_1, AZ_0)$ |                                              |
8. Pilihlah satu pernyataan yang paling tepat diantara pernyataan-pernyataan berikut
- Jika  $L = L(DPDA)$  maka  $L$  adalah tidak ambigu dan sebaliknya
  - Semua string  $w$  yang diterima oleh  $L(DPDA)$  adalah diterima dengan final state
  - Jika  $R$  adalah reguler ekspresi maka  $R$  adalah  $L(DPDA)$  dan sebaliknya
  - Semua jawaban benar
  - Semua jawaban salah**
9. Diantara himpunan string-string berikut yang tidak mempunyai sifat prefix property adalah :
- $\{O^n \mid n > 0\}$
  - $\{(01)^n \mid n > 0\}$
  - $\{(O^n + O^m) \mid m, n > 0\}$
  - $\{(110)^n \mid n > 0\}$
  - Semua jawaban salah
  - Semua jawaban benar.**

## CNF dan CYK

10. Pada tata-bahasa (grammar) berikut, yang merupakan simbol nullable adalah:
- C
  - ABC
  - SABC
  - SABCD
  - SABD
- $$\begin{aligned} S &\rightarrow aAa \mid bBb \mid AB \\ A &\rightarrow C \mid a \\ B &\rightarrow C \mid b \\ C &\rightarrow CDE \mid \epsilon \\ D &\rightarrow A \mid B \mid ab \end{aligned}$$
11. Pada tata-bahasa (grammar) di soal sebelumnya setelah dilakukan eliminasi epsilon maka simbol S akan menjadi
- $S \rightarrow aa \mid bb \mid AB$
  - $S \rightarrow aAa \mid aa \mid bBb \mid bb \mid AB$
  - $S \rightarrow aAa \mid aa \mid bBb \mid bb \mid AB \mid A \mid B$**
  - $S \rightarrow aAa \mid aa \mid bBb \mid bb \mid AB \mid A \mid B \mid ab$
  - Tidak ada jawaban yang benar
12. Untuk menghasilkan bentuk CNF maka aturan produksi  $S \rightarrow ABC \mid Bbb \mid aAa$ , diperlukan variabel tambahan sebanyak
- 2
  - 3
  - 4
  - 5**
  - 6
13. Algoritma CYK digunakan untuk menguji apakah sebuah string merupakan elemen dari bahasa yang diberikan. Dari tabel-filling CYK bisa diketahui bahwa string  $w=x_1 x_2 x_3 x_4 x_5$  merupakan elemen dari bahasa atau bukan, jika :
- Pada tabel tersebut tidak ada elemen yang kosong
  - Semua elemen pada 3 baris pertama dari bawah sudah dihitung semua
  - Semua elemen pada kolom pertama sudah dihitung semua**
  - Elemen pada posisi  $X_{55}$  sudah dihitung
  - Tidak ada jawaban yang benar.

14. Gunakan algoritma CYK untuk mengisi elemen  $X_{ij}$  dari tabel-filling jika diberikan string bbaab dengan grammar sebagai berikut (silahkan dengan cara memilih jawaban yang benar) \*

|                            |
|----------------------------|
| $S \rightarrow AB \mid BC$ |
| $A \rightarrow BA \mid a$  |
| $B \rightarrow CC \mid b$  |
| $C \rightarrow AB \mid a$  |

Mark only one oval per row.

|     | -                                | {A}                   | {B}                              | {S,A}                            | {S,C}                            | {A,C}                 | {S,A,C}               |
|-----|----------------------------------|-----------------------|----------------------------------|----------------------------------|----------------------------------|-----------------------|-----------------------|
| X12 | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/>            | <input type="radio"/>            | <input type="radio"/>            | <input type="radio"/> | <input type="radio"/> |
| X23 | <input type="radio"/>            | <input type="radio"/> | <input type="radio"/>            | <input checked="" type="radio"/> | <input type="radio"/>            | <input type="radio"/> | <input type="radio"/> |
| X34 | <input type="radio"/>            | <input type="radio"/> | <input checked="" type="radio"/> | <input type="radio"/>            | <input type="radio"/>            | <input type="radio"/> | <input type="radio"/> |
| X45 | <input type="radio"/>            | <input type="radio"/> | <input type="radio"/>            | <input type="radio"/>            | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/> |
| X15 | <input checked="" type="radio"/> | <input type="radio"/> | <input type="radio"/>            | <input type="radio"/>            | <input type="radio"/>            | <input type="radio"/> | <input type="radio"/> |

### Turing Machine

15. Sebuah MT M mempunyai fungsi transisi yang ditunjukkan pada tabel dibawah. Diantara ID yang valid dari rangkaian pengenalan string 000111 adalah:

- a. 0q100111
- b. XXXq3YYY
- c. XX0YYq01
- d. XXq20YY1**
- e. XXXYYYq4

| state | Simbol        |               |               |               |               |
|-------|---------------|---------------|---------------|---------------|---------------|
|       | 0             | 1             | X             | Y             | B             |
| $q_0$ | $(q_1, X, R)$ |               |               | $(q_1, Y, R)$ |               |
| $q_1$ | $(q_1, 0, R)$ | $(q_2, Y, L)$ |               | $(q_1, Y, R)$ |               |
| $q_2$ | $(q_2, 0, L)$ |               | $(q_0, X, R)$ | $(q_2, Y, L)$ |               |
| $q_3$ |               |               |               | $(q_1, Y, R)$ | $(q_1, B, R)$ |
| $q_4$ |               |               |               |               |               |

16. Sebuah nondeterministik MT  $M=((q_0, q_1, q_2), \{0, 1\}, \{0, 1, B\}, \delta, q_0, B, \{q_2\})$  mempunyai fungsi transisi sebagai berikut. Jika diberikan string 011 maka ID akhir ketika string tersebut diterima adalah

- a. 101q2
- b. 111q2**
- c. 011 Bq2
- d. 110q2
- e. 110Bq2

| $\delta$ | 0                              | 1                              | B                 |
|----------|--------------------------------|--------------------------------|-------------------|
| $q_0$    | $\{(q_0, 1, R)\}$              | $\{(q_1, 0, R)\}$              | $\emptyset$       |
| $q_1$    | $\{(q_1, 0, R), (q_0, 0, L)\}$ | $\{(q_1, 1, R), (q_0, 1, L)\}$ | $\{(q_2, B, R)\}$ |
| $q_2$    | $\emptyset$                    | $\emptyset$                    | $\emptyset$       |

- f. Tidak ada jawaban yang benar

17. Diantara bahasa-bahasa berikut yang HANYA bisa dikenali oleh MT adalah ( $w^R$ : w reversal/dibalik)

- a.  $L = \{a^n b^n \mid n > 0\}$
- b.  $L = \{ww^R \mid w = (a+b)^n, n > 0\}$**
- c.  $L = \{wcw^R \mid w = (a+b)^n, n > 0\}$
- d.  $L = \{a^m b^n c^n \mid m, n > 0\}$
- e.  **$L = \{a^n b^n c^n \mid n > 0\}$**

18. Sebuah deterministik Mesin Turing M dengan satu pita dapat mengenali bahasa  $L = \{ww^R \mid w \text{ string dengan panjang } n > 0\}$  dengan kompleksitas waktu (orde) :

- a.  $O(\sqrt{n})$
- b.  $O(\log(n))$
- c.  $O(n)$
- d.  $O(n^2)$**
- e.  $O(n^3)$

19. Statement: Jika sebuah language L adalah rekursif, maka itu artinya is bersifat tertutup untuk operasi:
- a. Union
  - b. Intersection
  - c. Complement
  - d. Union dan Intersection
  - e. **Union, Intersection, dan Complement**
  - f. Union dan Negation
20. Andaikan kita memiliki komputer sederhana (Von Neuman) yang memiliki kontrol unit 32 bit address + aritmatika unit 64 bit. Jumlah status yang dapat dimiliki komputer sederhana ini maksimal sebanyak? keterangan: "tanda  $\wedge$  dibaca pangkat"
- a. 32
  - b. 64
  - c. **96**
  - d.  **$2^{32}$**
  - e.  **$2^{64}$**
  - f.  **$2^{96}$**
  - g.  **$2^{(32*64)}$**

## UAS TBFO Semester I 2021/2022

### Desain CFG

Untuk setiap soal esai di bawah ini, tuliskan CFG nya pada tempat yang disediakan. Setiap baris pada esai hanya berisi list aturan produksi untuk satu LHS yang sama. Contoh:

S -> SA | AB | B

A -> 0A1 | eps B -> 1B0 | 10

1. Buatlah CFG dengan hanya 1 simbol non terminal (S) dan 2 simbol terminal (a, b) serta simbol "eps" untuk bahasa di bawah ini. Contoh string diterima adalah eps, ab, abb, aabb, aabbb, aabbbb.

$$L(G) = \{ a^n b^m \mid 0 \leq n \leq m \leq 2n \}$$

2. Buatlah CFG dengan hanya 3 simbol non terminal (S, A, T) dan 3 simbol terminal (a, b, #) serta simbol khusus (eps) untuk bahasa di bawah ini. Contoh string diterima adalah ab#ba, ab#bba, ab#aaaabaaaaaa

$$\{w\#x \mid w^R \text{ is a substring of } x, \text{ where } w, x \in \{a, b\}^*\}$$

3. Buatlah CFG dengan hanya 5 simbol non terminal (S, A, B, C, D) dan 3 simbol terminal (0, 1, 2) serta simbol "eps" untuk bahasa di bawah ini. Contoh string diterima adalah eps, 01, 12, 01222, 00001122

$$L(G) = \{0^i 1^j 2^k : i=j \text{ or } j=k\}$$

### CFG

1. Diberikan CFG sebagai berikut. Pernyataan yang tepat untuk string-string yang dihasilkan adalah:

|               |             |
|---------------|-------------|
| S -> XD   AY; | A -> aA   a |
| X -> aXc   B; | B -> bB   b |
| Y -> bYd   C; | C -> cC   c |
|               | D -> dD   d |

- a). aaabbcccccddd
- b). aaaabbccddd
- c). aabbcccccddd
- d). aabbcccccddd
- e). Tidak ada jawaban yang benar.

2. Berdasarkan grammar berikut, yang merupakan bentuk left-sentential-form adalah :

S -> A1B

$$A \rightarrow 0A \mid 0$$

B -> 0B | 1B | 0 | 1

- a). 0001B
  - b). 0010B
  - c). 0011B
  - d). 0101B
  - e). Tidak ada jawaban yang benar

3. Diantara pernyataan berikut yang paling benar tentang inherently unambiguous grammar. Jika  $L$  adalah CFL maka:

- a). Untuk setiap  $w$  anggota  $L$ ,  $w$  mempunyai dua pohon parsing berbeda
  - b). Grammar dari  $L$  bisa menghasilkan banyak pohon parsing untuk setiap  $w$  di  $L$
  - c). Beberapa grammar  $L$  bersifat ambiguous dan beberapa yang lain unambiguous
  - d). Bahasa  $L$  tidak mempunyai grammar yang unambiguos
  - e). Tidak ada jawaban yang benar

4. Pilihlah simbol benar isi setiap sel di tabel penelusuran CYK untuk kalimat "Book that flight" pada soal di bawah ini. Semua jawaban benar untuk setiap sel harus dipilih. Misalkan jika sebuah sel seharusnya berisi dua simbol, sedangkan jawaban mahasiswa itu hanya satu simbol, maka nilai dari jawaban tsb adalah nol.

Untuk Language dengan CFG di bawah ini:

$S \rightarrow NP\ VP \mid VP$

NP → Det Noun | Noun

VP → Verb NP

Noun → book | flight

Verb → book

Det → that

Lengkapilah setiap sel ber-index di bawah ini dengan menggunakan CYK algorithm

|       |       |        |
|-------|-------|--------|
| [1,1] |       |        |
| [2,1] | [2,2] |        |
| [3,1] | [3,2] | [3,3]  |
| Book  | that  | flight |

|       |          |          |          |          |          |          |          |
|-------|----------|----------|----------|----------|----------|----------|----------|
| [3,2] | Ya/Tidak |
| [3,3] | Ya/Tidak |
| [2,1] | Ya/Tidak |
| [2,2] | Ya/Tidak |
| [1,1] | Ya/Tidak |

5. Untuk language  $L = \{ w \mid w \text{ memiliki prefix dimana jumlah } 0 \geq \text{jumlah } 1 \}$ , pilihlah benar salah untuk setiap kandidat aturan produksi berikut:

|                            | Benar | Salah |
|----------------------------|-------|-------|
| $S \rightarrow 0 S 1$      |       |       |
| $S \rightarrow 0 S$        |       |       |
| $S \rightarrow \text{eps}$ |       |       |

6. diketahui CFG sbb  $S \rightarrow S0S1S \mid S1S0S \mid \text{eps}$  ; pilihlah benar atau salah untuk setiap pernyataan di bawah ini terkait PDA hasil konversi CFG tsb

|                                                                   | Benar | Salah |
|-------------------------------------------------------------------|-------|-------|
| Stack symbol terdiri atas: S, 0, 1, eps                           |       |       |
| $\delta(q, S, S) = \{(q, \text{eps})\}$                           |       |       |
| salah satu output dari $\delta(q, \text{eps}, S)$ adalah (q, eps) |       |       |

7. Diberikan CFG sebagai berikut. Pernyataan yang tepat untuk string-string yang dihasilkan adalah:

$$\begin{aligned} S &\rightarrow aB \mid bA \\ A &\rightarrow a \mid aS \mid bAA \\ B &\rightarrow b \mid bS \mid aBB \end{aligned}$$

- a) aaabbabbba
- b) ababbbbaaab
- c) aabaabaaab

d) abbaabaaab

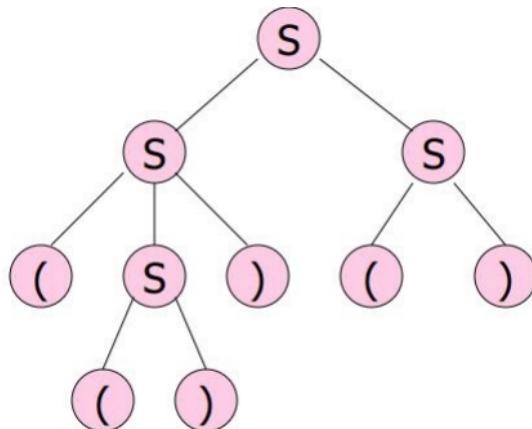
8. Diberikan CFG sebagai berikut. Pernyataan yang tepat untuk string-string yang dihasilkan adalah:

$$S \rightarrow aAS \mid a$$

$$A \rightarrow SbA \mid ba$$

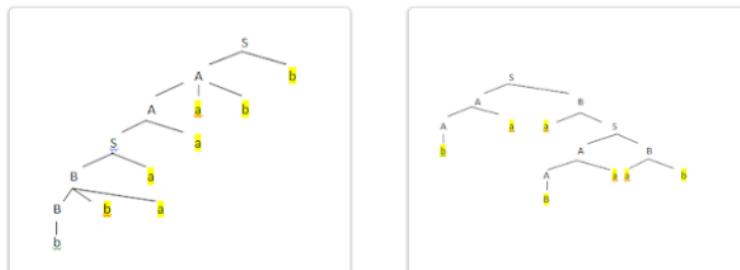
- a) abaaba
- b) aabbba
- c) abbaab
- d) ababab

9. Bentuk dari parse tree



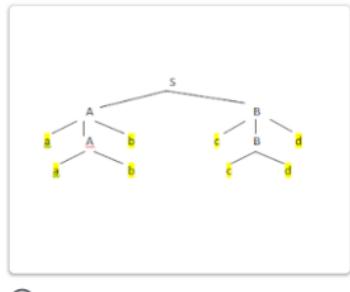
- a)  $S \rightarrow SS \mid (S) \mid ()$
- b)  $S \rightarrow SS \mid () \mid (S)$
- c)  $S \rightarrow (S) \mid S \mid ()$
- d)  $S \rightarrow (S) \mid SS \mid ()$

10.  $S \rightarrow AB$   $A \rightarrow Aa \mid a \mid b$   $B \rightarrow a \mid Sb$  Pohon penurunan untai 'baabaab'

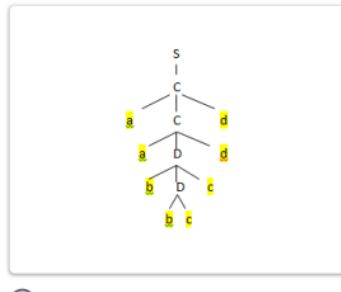


A

B



C



D

1. Di bawah ini adalah PDA dengan 4 state ( $q_0, q_1, q_2, q_3$ ) yang perlu dilengkapi.  $q_0$  adalah start state, dan  $q_3$  adalah final state. A, B,  $Z_0$  adalah stack symbol dimana  $Z_0$  adalah start top of stack symbol. Sebagai tambahan catatan: cursor berpindah dari state  $q_0$  ke  $q_1$  ketika input simbol adalah "b",  $q_0$  adalah state untuk menerima "a",  $q_1$  adalah state untuk menerima "b". Untuk PDA yang lengkap, pilihlah benar atau salah untuk setiap transition function pada pernyataan di bawah

$$L = \{a^n b^m : m \geq n + 2\}$$



|                                           | Benar | Salah |
|-------------------------------------------|-------|-------|
| $\delta(q_0, b, A) = \{(q_1, A)\}$        |       |       |
| $\delta(q_1, b, A) = \{(q_1, \epsilon)\}$ |       |       |
| $\delta(q_3, b, Z_0) = \{(q_3, Z_0)\}$    |       |       |

2. Pilihlah benar atau salah untuk transition function pada PDA dengan 3 points language yang menerima string dengan jumlah simbol 0 nya dua kali lipat dari jumlah simbol 1. Contoh string yang diterima adalah  $\epsilon$ , 010, 100, 001, 010010. PDA tersebut memiliki 3 state ( $q_0, q_1, q_2$ ) dimana  $q_0$  adalah start state dan sekaligus final state. Dari state  $q_1$  akan berpindah ke state  $q_2$  jika input symbol adalah 1 dan top of stack adalah A. Tujuan dari adanya state  $q_2$  adalah untuk menghindari pembacaan dua nilai top of stack pada satu saat tertentu. PDA memiliki 3 jenis top of stack yaitu  $Z_0$  (start top of stack), A, dan B. Top of stack A akan di-push ketika input symbol yang diterima adalah 0; top of stack B akan di-push sebanyak 2 kali ketika input symbol yang diterima adalah 1. Salah satu transition function yang benar adalah  $\delta(q_0, 0, A) = \{(q_0, AA)\}$

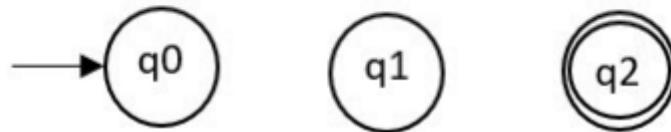
$$L = \{ w \mid w \text{ memiliki jumlah } 0 \text{ sebanyak } 2 \text{ kali jumlah } 1 \}$$



|                                            | Benar | Salah |
|--------------------------------------------|-------|-------|
| $\delta(q_0, 1, Z_0) = \{(q_1, BBZ_0)\}$   |       |       |
| $\delta(q_1, 1, B) = \{(q_2, BB)\}$        |       |       |
| $\delta(q_2, \epsilon, B) = \{(q_1, BB)\}$ |       |       |

3. Pilihlah benar atau salah pada setiap transition function untuk soal PDA di bawah ini. PDA memiliki tiga state ( $q_0, q_1, q_2$ ) dan satu stack symbol (A). Salah satu transition functionnya adalah  $\delta(q_0, a, A) = \{(q_0, AA)\}$

$$L = \{a^n b^m c^{n-m} \mid n > 0, m \geq 0, n > m\}$$



|                                           | Benar | Salah |
|-------------------------------------------|-------|-------|
| $\delta(q_0, b, A) = \{(q_1, A)\}$        |       |       |
| $\delta(q_1, c, A) = \{(q_1, \epsilon)\}$ |       |       |

4. Diberikan PDA  $P = \{p, q, r\}, \{0, 1\}, \{X, Z\}, \delta, q_0, Z\}$  dan salah satu fungsi transisinya adalah  $\delta(p, 0, X) = \{\{r, XX\}\}$ . Jika PDA tersebut diubah menjadi CFG, maka aturan produksi yang valid dari fungsi transisi tersebut adalah:
  - a).  $[pXq] \rightarrow 0 [qXr][rXq]$
  - b).  $[pXq] \rightarrow 0 [rXp][pXr]$
  - c).  $[pXr] \rightarrow 0 [pXq][qXr]$
  - d).  $[pXp] \rightarrow 0 [rXr][qXq]$
  - e).  $[pXq] \rightarrow 0 [rXp][pXq]$
5. Setiap bahasa yang termasuk Deterministik Pushdown Automata (DPDA) bersifat unambiguous, tetapi tidak semua bahasa yang unambiguous adalah DPDA. String berikut adalah anggota bahasa yang tidak termasuk DPDA tetapi unambiguous.
  - a). 1010101010
  - b). 111000111000
  - c). 010101101010
  - d). 000000111111
  - e). Tidak ada jawaban yang benar
6. Suatu instantaneous description (ID) dideskripsikan triple  $(q, w, \alpha)$ , dimana
  - q adalah current state
  - w adalah remaining input
  - alpha adalah stack contents, top at the left
  - None of them

### Properties CFL

1. Pilihan yang paling benar diantara jawaban berikut, tentang sifat-sifat ketertutupan CFL. CFL mempunyai sifat tidak tertutup terhadap:
  - a). Union
  - b). Concatenation
  - c). Complement
  - d). Jawaban a dan b benar
  - e). Jawaban b dan c benar
2. Berikut ini merupakan langkah-langkah mengubah CFG ke CNF, KECUALI
  - a) Hapus e-production

- b) Hapus unit production
  - c) Hapus useful variable
  - d) Hapus useless variable
3. Untuk membentuk clean grammar, salah satu langkah yang harus dilewati adalah menghilangkan unit production. Pernyataan yang benar tentang eliminasi unit production adalah.
- a). Mengganti variabel ruas kanan unit production dengan simbol terminal
  - b). Menghapus langsung unit production yang ada pada grammar
  - c). Mengganti variabel ruas kanan unit production dengan variabel yang lain
  - d). Semua jawaban a, b, dan c benar
  - e). Tidak ada jawaban yang benar
4. Pilihan yang paling benar tentang sifat undecidable untuk bahasa bebas konteks adalah.
- a). Berapa jumlah variable dari suatu CFL
  - b). Apakah sebuah CFL ambiguous
  - c). Apakah dua buah CFL sama
  - d). Jawaban a dan b benar
  - e). Jawaban b dan c benar