PRAKTIKUM 1.1 dan 1.2

Mata Kuliah Algoritma dan Struktur Data



Nama Dosen Pengampu

Ibu Umi Sa'adah S.Kom, M.Kom

Dikerjakan oleh

Nama: Muhammad Rafi Rizaldi

NRP: 3123600001

Kelas: 1 D4 IT A

PRAKTIKUM 1.1

1. Dengan menggunakan array of struct, buatlah program sbb:

Input data-data mahasiswa yang terdiri atas :

nama, nilai Tugas, nilai UTS, nilai UAS sebanyak n mahasiswa

Output:

Tampilkan dalam bentuk tabel

No, NRP, Nama Mhs, Nilai tugas, nilai UTS, nilai UAS, nilai Akhir, Grade

(Nilai Akhir = 20% tugas + 40% UTS + 40%UAS) dengan ketentuan sbb:

- Nilai akhir >= 80 Grade A
- Nilai akhir >= 70 Grade B
- Nilai akhir >= 60 Grade C
- Nilai akhir >= 50 Grade D
- Nilai akhir < 50 Grade E

Petunjuk:

- Gunakan array of struct (tanpa typedef)
- Perhatikan contoh tampilan
- Nilai tambah, gunakan array sebagai parameter fungsi (misal 2 fungsi : input() dan tampil())

```
#include <stdlib.h>
#include <string.h>
#include <stdio.h>

struct Data
{
   char nama[50];
   float nilai_tugas;
   float nilai_uts;
   float nilai_akhir;
   char grade;
```

```
void PrintLine();
void InputData(int *, struct Data **);
char GetGrade(float);
void PrintResult(int, struct Data *);
void clearScreen();
int main(int argc, char const *argv[])
  free (mahasiswa);
void InputData(int *jumlah, struct Data **data)
PEMROGRAMAN\n\n");
```

```
scanf("%f", &(*data)[i].nilai uts);
void PrintResult(int jumlah, struct Data *data)
  printf("\t\t\tMATAKULIAH KONSEP PEMROGRAMAN\n");
data[i].nama, data[i].nilai_tugas, data[i].nilai_uts,
data[i].nilai uas, data[i].nilai akhir, data[i].grade);
char GetGrade(float akhir)
```

```
void PrintLine()
{
    for (int i = 0; i < 80; i++)
    {
        printf("-");
    }
    printf("\n");
}

void clearScreen()
{
#ifdef _WIN32
    system("cls"); // For Windows
#else
    printf("\033[2J\033[H"); // For Linux/Unix
#endif
}</pre>
```

```
o eros@Rafi-Rizaldi:~/KULIAH2/ASD/prakt1.1$ ./tes
 MENGHITUNG NILAI AKHIR
 MATA KULIAH KONSEP PEMROGRAMAN
 Berapa jumlah mahasiswa ? 3
 MASUKKAN DATA MAHASISWA
 Mahasiswa ke - 1
 Nama : anas
 Nilai Tugas : 76
 Nilai UTS: 86
 Nilai UAS : 56
 Mahasiswa ke - 2
 Nama : wawa
 Nilai Tugas : 85
 Nilai UTS: 78
 Nilai UAS: 99
 Mahasiswa ke - 3
 Nama : satrio
 Nilai Tugas : 76
 Nilai UTS : 65
 Nilai UAS : 55
```

DAFTAR NILAI MATAKULIAH KONSEP PEMROGRAMAN										
No	Nama Mahasiswa	Tugas	Nilai UTS	UAS	Akhir	Grade				
1 2 3	anas wawa satrio	76.0 85.0 76.0	86.0 78.0 65.0	56.0 99.0 55.0	72.0 87.8 63.2	В А С				

- 2. Dengan menggunakan typedef, buatlah tipe data struct untuk menyimpan informasi pegawai (misal tipe pegawai) berupa : no ID, nama, tgl lahir, jenis kelamin (L/P) dan gaji/bulan.
 - di main() deklarasikan sebuah var bertipe array of pegawai (misal emp[])
 - panggil fungsi input() untuk memasukkan data-datanya dengan pengiriman parameter secara pass by reference
 - panggil fungsi tampil() untuk menampilkan semua data yang telah diinputkan dengan pengiriman parameter secara pass by reference

Petunjuk:

- -Untuk no id, gunakan auto increment (tanpa input) perhatikan contoh tampilan
- -Jika diperlukan, gunakan fungsi strcpy() dari <string.h>

```
#include <stdio.h>
#include <stdib.h>
#include <string.h>

typedef struct {
  int dd, mm, yyyy;
}Date;

typedef struct info
{
  int id;
  char nama[40];
  Date ttl;
  char gender;
  int gaji;
} Informasi;
```

```
void InputData(Informasi **, int *);
void PrintResult(Informasi *, int);
void clearScreen();
int main(int argc, char const *argv[])
void InputData(Informasi **data, int *jumlah)
  printf("Tgl lahir (dd-mm-yy)\t: ");
&(*data)[i].ttl.yyyy);
```

```
void PrintResult(Informasi *data, int jumlah)
data[i].ttl.yyyy);
data[i].gender == 'L') ? "Laki-laki " : "Perempuan");
void clearScreen()
#ifdef WIN32
#endif
```

DATA PEGAWAI

Berapa jumlah pegawai : 3

Data pegawai ke-1

Nama : anas

Tgl lahir (dd-mm-yy) : 22-5-2000

Jenis kelamin (L/P) : l

Gaji/bulan : 1000000

Data pegawai ke-2

Nama : tata

Tgl lahir (dd-mm-vy) : 23-5-1999

Jenis kelamin (L/P) : p

Gaji/bulan : 2000000

Data pegawai ke-3

Nama : masha Tgl lahir (dd-mm-yy) : 14-2-2001

Jenis kelamin (L/P) : p

Gaji/bulan : 1500000

Data pegawai yang diinputkan

No ID : 1 Nama : anas

Tgl lahir : 22-5-2000 Jenis Kelamin : Laki-laki Gaji/bulan : Rp.1000000

No ID : 2 Nama : tata

Tgl lahir : 23-5-1999 Jenis Kelamin : Perempuan Gaji/bulan : Rp.2000000

No ID : 3
Nama : masha
Tgl lahir : 14-2-2001
Jenis Kelamin : Perempuan
Gaji/bulan : Rp.1500000

- 3. Modifikasi program dari soal no 1.
 - Ubah nama tipe data dengan menggunakan typedef
 - Panggillah fungsi input() dan tampil() dengan cara pass by reference

Petunjuk:

- Untuk soal nomor 2 & 3, perhatikan cara membaca array dengan menggunakan pointer, buka kembali modul teori pointer to array

```
#include <stdlib.h>
#include <string.h>
#include <stdio.h>
Data;
void PrintLine();
void InputData(int *, Data **);
char GetGrade(float);
void PrintResult(int, Data *);
void clearScreen();
int main(int argc, char const *argv[])
 free (mahasiswa);
```

```
void InputData(int *jumlah, Data **data)
void PrintResult(int jumlah, Data *data)
```

```
data[i].nama, data[i].nilai tugas, data[i].nilai uts,
data[i].nilai uas, data[i].nilai akhir, data[i].grade);
char GetGrade(float akhir)
void PrintLine()
void clearScreen()
#ifdef WIN32
#else
#endif
```

```
o eros@Rafi-Rizaldi:~/KULIAH2/ASD/prakt1.1$ ./tes
 MENGHITUNG NILAI AKHIR
 MATA KULIAH KONSEP PEMROGRAMAN
 Berapa jumlah mahasiswa ? 3
 MASUKKAN DATA MAHASISWA
 Mahasiswa ke - 1
 Nama : anas
 Nilai Tugas : 76
 Nilai UTS: 86
 Nilai UAS : 56
 Mahasiswa ke - 2
 Nama : wawa
 Nilai Tugas : 85
 Nilai UTS: 78
 Nilai UAS : 99
 Mahasiswa ke - 3
 Nama : satrio
 Nilai Tugas : 76
 Nilai UTS : 65
 Nilai UAS : 55
```

DAFTAR NILAI MATAKULIAH KONSEP PEMROGRAMAN										
No	Nama Mahasiswa	Tugas	Nilai UTS	UAS	Akhir	Grade				
1 2 3	anas wawa satrio	76.0 85.0 76.0	86.0 78.0 65.0	56.0 99.0 55.0	72.0 87.8 63.2	B A C				

4. Untuk soal no 3 diatas, pada tampilan output, tambahkan informasi : nama & nilai akhir mahasiswa yang tertinggi (lihat contoh)

Petunjuk:

- buatlah tipe data struct baru misal dengan nama maks dengan 2 field : nama & nilai untuk menyimpan informasi mahasiswa dengan nilai tertinggi
- Sertakan <string.h> agar bisa menggunakan fungsi strcpy() untuk mengcopy nama Source Code :

```
#include <stdlib.h>
#include <string.h>
#include <stdio.h>
typedef struct data
Data;
typedef struct
void PrintLine();
void InputData(int *, Data **);
char GetGrade(float);
void PrintResult(int, Data *);
void FindTheBest(Data *, int);
int main(int argc, char const *argv[])
void InputData(int *jumlah, Data **data)
PEMROGRAMAN\n\n");
```

```
printf("Berapa jumlah mahasiswa ? ");
   printf("Nilai UAS : ");
void PrintResult(int jumlah, Data *data)
 printf("\t\t\tDAFTAR NILAI\n");
 printf("No\tNama\t\t\tNilai\t\t\tGrade\n");
```

```
data[i].nama, data[i].nilai tugas, data[i].nilai uts,
data[i].nilai uas, data[i].nilai akhir, data[i].grade);
char GetGrade(float akhir)
void FindTheBest(Data *data, int total data)
```

```
{
    printf("-");
}
printf("\n");
}
```

```
MENGHITUNG NILAI AKHIR
MATA KULIAH KONSEP PEMROGRAMAN
Berapa jumlah mahasiswa ? 3
MASUKKAN DATA MAHASISWA
Mahasiswa ke - 1
Nama : AKMAL
Nilai Tugas: 77
Nilai UTS : 88
Nilai UAS: 99
Mahasiswa ke - 2
Nama : GERUNG
Nilai Tugas : 88
Nilai UTS : 77
Nilai UAS : 56
Mahasiswa ke - 3
Nama : JERO
Nilai Tugas: 99
Nilai UTS : 77
Nilai UAS : 88
                              DAFTAR NILAI
                       MATAKULIAH KONSEP PEMROGRAMAN
                              Nilai
No
                                                     Grade
       Mahasiswa
                      Tugas
                              UTS UAS Akhir
                      77.0 88.0 99.0
88.0 77.0 56.0
                                             90.2 A
70.8 B
      AKMAL
                      88.0
2
       GERUNG
       JERO
                      99.0
                              77.0
                                      88.0
                                             85.8 A
Total mahasiswa = 3
Nilai tertinggi
Nama = AKMAL
Nilai = 90.2
```

PRAKTIKUM 1.2

1. Menggunakan fungsi sizeof() untuk menentukan ukuran objek

```
#include <stdio.h>
typedef struct employee_st
{
   char name[40];
   int id;
} Employee;
int main()
{
   int myInt;
   Employee john;
   printf("Size of int is %d\n", sizeof(myInt));
   // The argument of sizeof is an object
   printf("Size of int is %d\n", sizeof(int));
   // The argument of sizeof is a data type
   printf("Size of Employee is %d\n", sizeof(Employee));
   // The argument of sizeof is an object
   printf("Size of john is %d\n", sizeof(john));
   // The argument of sizeof is a data type
   printf("Size of char is %d\n", sizeof(char));
   printf("Size of short is %d\n", sizeof(short));
   printf("Size of int is %d\n", sizeof(int));
   printf("Size of long is %d\n", sizeof(long));
   printf("Size of double is %d\n", sizeof(double));
}
```

```
• mrrizzz@Draverezld:/mnt/d/KULIAH2-ASD/ASD/prakt1.2$ ./tes
Size of int is 4
Size of Employee is 44
Size of john is 44
Size of char is 1
Size of short is 2
Size of int is 4
Size of long is 8
Size of float is 4
Size of double is 8
```

2. Menggunakan fungsi malloc() untuk mengalokasikan memory

```
#include <stdio.h>
#include <stdib.h>
#include <string.h>
int main()
{
   char s1[] = "This is a sentence";
   char *pblok;
   pblok = (char *)malloc(strlen((s1) + 1));

   if (pblok == NULL)
      printf("Error on malloc\n");
   else
   {
      strcpy(pblok, s1);
      printf("s1: %s\n", s1);
      printf("pblok: %s\n", pblok);
   }
}
```

Output:

```
mrrizzz@Draverezld:/mnt/d/K
s1: This is a sentence
pblok: This is a sentence
```

3. Menggunakan fungsi free() untuk membebaskan kembali memory

```
#include <stdio.h>
```

```
#include <stdlib.h>
int main()
{
    char *pblok;
    pblok = (char *)malloc(500 * sizeof(char));
    if (pblok == NULL)
        puts("Error on malloc");
    else
    {
        puts("OK, alokasi memory sudah dilakukan");
        puts("----");
        free(pblok);
        pblok = NULL;
        puts("Blok memory telah dibebaskan kembali");
        puts("dan pointernya sdh digroundkan");
    }
}
```

```
• mrrizzz@Draverezld:/mnt/d/KULIAH2-ASD/ASD
OK, alokasi memory sudah dilakukan
-----
Blok memory telah dibebaskan kembali
dan pointernya sdh digroundkan
```

4. Mengalokasikan memory untuk objek struct dengan jumlah dan data yang diinputkan kemudian menampilkannya

```
#include <stdio.h>
#include <stdib.h>
#include <string.h>
typedef struct employee_st
{
   char name[40];
   int id;
} Employee;
int main()
{
   Employee *workers, *wpt;
   int num, i;
```

```
printf("How many employees do you want ? ");
workers = (Employee *)malloc(num * sizeof(Employee));
```

```
mrrizzz@Draverezld:/mnt/d/KULIAH2-ASD//
How many employees do you want ? 3
Employee's name : roger
Employee's id : 3
Employee's name : miya
Employee's id : 2
Employee's name : lolita
Employee's id : 1
Employee 3 is roger
Employee 2 is miya
Employee 1 is lolita
```

q4.

```
#include <stdio.h>
int sameDay(Date, Date);
int main(int argc, char const *argv[])
 printf("Masukkan tanggal pertama (dd-mm-yyyy): ");
&first date.year);
 printf("Masukkan tanggal kedua (dd-mm-yyyy): ");
&second date.year);
int sameDay(Date first date, Date second date)
```

```
return (first_date.day == second_date.day && first_date.month ==
second_date.month && first_date.year == second_date.year) ? 1 : 0;
}
```

```
    mrrizzz@Draverezld:/mnt/d/KULIAH2-ASD/ASD/prakt1.2$ ./tes
        Masukkan tanggal pertama (dd-mm-yyyy): 22-4-2000
        Masukkan tanggal kedua (dd-mm-yyyy): 22-4-2001
        It is not the same day
        mrrizzz@Draverezld:/mnt/d/KULIAH2-ASD/ASD/prakt1.2$ ./tes
        Masukkan tanggal pertama (dd-mm-yyyy): 22-4-2000
        Masukkan tanggal kedua (dd-mm-yyyy): 22-4-2000
        It is the same day
```

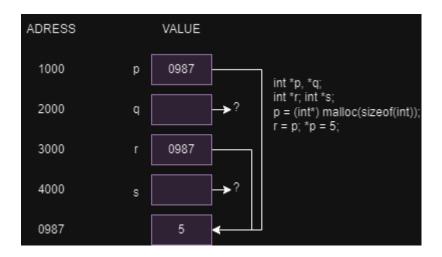
q5.

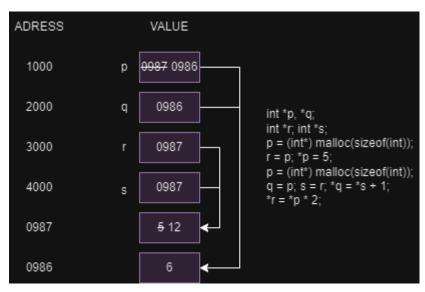
```
typedef struct
{
   int day, month, year;
} Date;
int isYounger(Date, Date);
int main(int argc, char const *argv[])
{
   Date first_student, second_student;
   printf("Input the first student's birth date (dd-mm-yyyy): ");
   scanf("%d-%d-%d", &first_student.day, &first_student.month,
&first_student.year);
   printf("Input the second student's birth date (dd-mm-yyyy): ");
   scanf("%d-%d-%d", &second_student.day, &second_student.month,
&second_student.year);
   printf(isYounger(first_student, second_student) == 1 ? "Yes, student
1 is younger than student 2\n": "No, student 1 is not younger than
student 2\n");
   return 0;
```

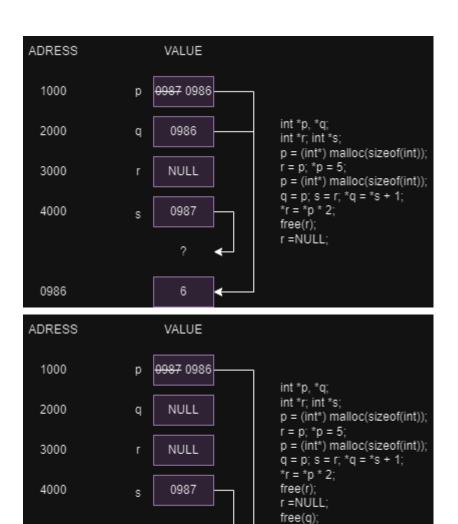
```
int isYounger(Date student1, Date student2)
{
   if (student1.year < student2.year)
     return 1;
   else if (student1.year == student2.year)
   {
     if (student1.month < student2.month)
        return 1;
     else if (student1.month == student2.month)
     {
        if (student1.day < student2.day)
           return 1;
        else
           return 0;
     }
     else
     return 0;
}</pre>
```

• mrrizzz@Draverezld:/mnt/d/KULIAH2-ASD/ASD/prakt1.2\$./tes
Input the first student's birth date (dd-mm-yyyy): 12-4-2000
Input the second student's birth date (dd-mm-yyyy): 12-3-2001
Yes, student 1 is younger than student 2

q6.







q =NÜLL;