

E-UTS PRAKTIKUM ALGORITMA DAN STRUKTUR DATA



Nama : Daffa Putra Alwansyah
NIM : L200190031
Kelas : B
Prodi : Informatika

**Fakultas Komunikasi dan Informatika Universitas
Muhammadiyah Surakarta**

1. Buatlah fungsi untuk menghitung luas bangun datar berikut ini :

a) Persegi

code:

```
#Persegi
def luas_persegi(): #membuat fungsi luas persegi
    print("Program menghitung luas Persegi")
    sisi = float(input("Input sisi persegi: ")) #variabel sisi untuk menginput nilai sisi, type data float
    luas = sisi * sisi #variabel menghitung luas persegi
    print ("Luas Persegi: ", str(luas), "satuan luas") #menampilkan hasil Luas Persegi dan mengconvert ke string
```

output:

```
>>> luas_persegi()
Program menghitung luas Persegi
Input sisi persegi: 5
Luas Persegi: 25.0 satuan luas
>>> |
```

b) Lingkaran

code:

```
#Lingkaran
def luas_lingkaran(): #membuat fungsi luas lingkaran
    print("Program menghitung luas Lingkaran")
    phi = 3.14 #variabel phi dengan nilai 3.14
    jari = float(input("Input jari-jari lingkaran: ")) #variabel jari untuk menginput nilai jari-jari, type data float
    luas = phi * jari * jari #variabel untuk menghitung Luas Lingkaran
    print("Luas lingkaran: ", str(luas), "satuan luas") # menampilkan hasil Luas Lingkaran dan mengkonvert ke string
```

output:

```
>>> luas_lingkaran()
Program menghitung luas Lingkaran
Input jari-jari lingkaran: 14
Luas lingkaran: 615.44 satuan luas
>>> |
```

c) Segitiga samasisi

code:

```
#Segitiga sama sisi
def luas_segsamasisi(): #Membuat fungsi luas segitiga sama sisi
    print("Program menghitung luas Segitiga Sama Sisi")
    alas = float(input("Input alas: ")) #variabel alas untuk menginput nilai alas, type data float
    tinggi = float(input("Input tinggi: ")) #variabel tinggi untuk menginput input tinggi, type data float
    luas = (1/2) * alas * tinggi #variabel untuk menghitung luas Segitiga Sama Sisi
    print("Luas Segitiga Sama Sisi: ", str(luas), "satuan luas") #menampilkan hasil Luas Segitiga Sama Sisi dan mengkonvert ke string
```

output:

```
>>> luas_segsamasisi()
Program menghitung luas Segitiga Sama Sisi
Input alas: 5
Input tinggi: 10
Luas Segitiga Sama Sisi: 25.0 satuan luas
```

d) Belah ketupat

code:

```
#BelahKetupat
def luas_belahketupat(): #Membuat fungsi luas belah ketupat
    print("Program menghitung luas Belah Ketupat")
    d1 = float(input("Masukan diagonal 1 Belah Ketupat: ")) #variabel d1 untuk menginput nilai diagonal 1, type data float
    d2 = float(input("Masukan diagonal 2 Belah Ketupat: ")) #variabel d2 untuk menginput nilai diagonal 2, type data float
    luas = (1/2) * d1 * d2 #variabel untuk menghitung luas Belah Ketupat
    print("Luas Belah Ketupat: ", str(luas), "satuan luas") #menampilkan hasil Luas Belah Ketupat dan mengkonvert ke string
```

output:

```
>>> luas_belahketupat()
Program menghitung luas Belah Ketupat
Masukan diagonal 1 Belah Ketupat: 6
Masukan diagonal 2 Belah Ketupat: 4
Luas Belah Ketupat:  12.0 satuan luas
>>> |
```

2. Buatlah program fungsi perkalian dua buah matriks! Matrik A dengan ordo (1x2) dan matriks B dengan ordo (2x3) maka hasilnya matriks C dengan ordo (1x3)

a) code:

```
#A
A = [[1,2]] #ordo 1x2
B = [[1,2,3], [1,2,3]] #ordo 2x3
def kali_matriks(matriks1,matriks2): #membuat fungsi dengan nama kali_matriks
    p = [] #menampung baris matriks
    for a in range(len(matriks1)): #iterasi matriks A
        #menampung kolom perbaris
        new = []
        for b in range(len(matriks2[0])): #iterasi matriks B
            #perkalian periteem pada kolom
            x = 0
            for c in range(len(matriks2)):
                x += matriks1[a][c] * matriks2[c][b]
            #hasil perkalian dimasukan ke dalam kolom baris new
            new.append(x)
        p.append(new) #hasil dimasukan kedalam list p
    print("Hasil dari perkalian ordo:" , p)
```

output:

```
>>> kali_matriks(A,B)
Hasil dari perkalian ordo: [[3, 6, 9]]
>>> |
```

b) Buatlah program matriks identitas dengan ordo (7 x 7)

code:

```
#B
def matriks_identitas(a):
    print("matriks identitas ordo: "+str(a)+"x"+str(a)) #menampilkan ordo
    #variabel matriks_id berisi iterasi untuk membuat kolom dan baris pada matriks
    matriks_id = [[1 if j == C else 0 for j in range(a)] for C in range(a)]
    for x in matriks_id: #iterasi didalam matriks
        print(x) #menampilkan hasil
```

Output:

```
>>> matriks_identitas(7)
matriks identitas ordo: 7x7
[1, 0, 0, 0, 0, 0, 0]
[0, 1, 0, 0, 0, 0, 0]
[0, 0, 1, 0, 0, 0, 0]
[0, 0, 0, 1, 0, 0, 0]
[0, 0, 0, 0, 1, 0, 0]
[0, 0, 0, 0, 0, 1, 0]
[0, 0, 0, 0, 0, 0, 1]
>>> |
```

3. Tambahkan 3 metode / fungsi untuk menampilkan dataNama, dataUmur, dan data warnaKulit. Dari class tersebut, tambahkan minimal 10 objek (data personal)

code:

```
##Nomor 3
class MhsTIF(object): #membuat kelas mhsTIF
    def __init__(self,nama,umur,warnaKulit): #membuat inisialisasi properti
        self.nama = nama #inisialisasi self.nama
        self.umur = umur #inisialisasi self.umur
        self.warnaKulit = warnaKulit #inisialisasi self.warnaKulit

#membuat 11 data pada kelas mhsTIF
c0 = MhsTIF('Daffa', 20, "Sawo Matang")
c1 = MhsTIF('Putra', 21, "Kuning Langsung")
c2 = MhsTIF('Alwansyah', 19, "Hitam")
c3 = MhsTIF('Ammar', 23, "Putih")
c4 = MhsTIF('Markotong', 22, "Hitam")
c5 = MhsTIF('Noval', 20, "Sawo Matang")
c6 = MhsTIF('Haris', 24, "Putih")
c7 = MhsTIF('Pandu', 25, "Sawo Matang")
c8 = MhsTIF('Khasan', 23, "Sawo Matang")
c9 = MhsTIF('Hanafi', 21, "Hitam")
c10 = MhsTIF('Dimas', 20, "Kuning Langsung")
#memasukkan var c0-c10 kedalam Daftar
Daftar=[c0,c1,c2,c3,c4,c5,c6,c7,c8,c9,c10]

##Fungsi Tampil Nama
def dataNama(x): #membuat fungsi dataNama
    for i in x: #melakukan iterasi
        print(i.nama) #setelah iterasi menampilkan nama

##Fungsi Tampil Umur
def dataUmur(x): #membuat fungsi dataUmur
    for i in x: #melakukan iterasi
        print(i.umur) #setelah iterasi menampilkan umur

##Fungsi Tampil Warna Kulit
def warnaKulit(x):
    for i in x: #melakukan iterasi
        print(i.warnaKulit) #setelah iterasi menampilkan warna kulit
```

Output:

```
>>> dataNama(Daftar)
Daffa
Putra
Alwansyah
Ammar
Markotong
Noval
Haris
Pandu
Khasan
Hanafi
Dimas
>>> dataUmur(Daftar)
20
21
19
23
22
20
24
25
23
21
20
>>> warnaKulit(Daftar)
Sawo Matang
Kuning Langsung
Hitam
Putih
Hitam
Sawo Matang
Putih
Sawo Matang
Sawo Matang
Hitam
Kuning Langsung
>>> |
```

4. Berdasarkan soal No.3, buatlah program untuk menampilkan daftar orang yang mempunyai kulit sawo matang (gunakan algoritma **pencarian**)!

code:

```
##Nomor 4
#Linier Search
def cari_kulit(x): #fungsi mencari kulit
    for i in range(len(Daftar)): #iterasi pada Daftar
        if x == Daftar[i].warnaKulit: #jika key == warna kulit didaftar, maka
            #menampilkan nama dan warna kulit pada iterasi Daftar
            print(Daftar[i].nama, "memiliki warna kulit: ", Daftar[i].warnaKulit)
```

output:

```
>>> cari_kulit("Sawo Matang")
Daffa memiliki warna kulit: Sawo Matang
Noval memiliki warna kulit: Sawo Matang
Pandu memiliki warna kulit: Sawo Matang
Khasan memiliki warna kulit: Sawo Matang
>>> |
```

5. Berdasarkan soal No.3, buatlah program untuk mengurutkan data tersebut dari umuryang termuda sampai tertua menggunakan algoritma **pengurutan!**

code:

```
#Nomor 5
#Fungsi swap yang digunakan untuk menukar posisi
def swap(i,j,k):
    temp = i[j]
    i[j]=i[k]
    i[k]=temp

#Fungsi cari umur dari terkecil
#Bubble Sort
def urut_umur(i):
    p = len(i)
    #melakukan iterasi pada Daftar - 1
    for x in range(p-1):
        #melakukan iterasi Daftar - iterasi - 1
        for y in range(p-x-1):
            #jika data current lebih besar dari selanjutnya maka akan dilakukan swap
            if i[y].umur > i[y+1].umur:
                swap(i,y,y+1)

#Fungsi untuk mengecek Daftar
def cek_umur(Daftar):
    for z in Daftar: #iterasi pada Daftar
        print(z.nama, z.umur, z.warnaKulit) #menampilkan isi pada Daftar
```

Output:

```
>>> cek_umur(Daftar)
Daffa 20 Sawo Matang
Putra 21 Kuning Langsung
Alwansyah 19 Hitam
Ammar 23 Putih
Markotong 22 Hitam
Noval 20 Sawo Matang
Haris 24 Putih
Pandu 25 Sawo Matang
Khasan 23 Sawo Matang
Hanafi 21 Hitam
Dimas 20 Kuning Langsung
>>> urut_umur(Daftar)
>>> cek_umur(Daftar)
Alwansyah 19 Hitam
Daffa 20 Sawo Matang
Noval 20 Sawo Matang
Dimas 20 Kuning Langsung
Putra 21 Kuning Langsung
Hanafi 21 Hitam
Markotong 22 Hitam
Ammar 23 Putih
Khasan 23 Sawo Matang
Haris 24 Putih
Pandu 25 Sawo Matang
>>> |
```