

▼ Modul 1

Algoritma dan Code

▼ 1 Konsep

▼ 1.1 Materi

Tulis ringkasan (1 sampai dengan 4 paragraf, jika memungkinkan berikan contoh dalam Python) tentang :

1. Tipe Data
2. Algoritma (Definisi, Jenis-jenis algoritma, bagaimana menulis algoritma)

1. Tipe data atau data types adalah pengelompokan data berdasarkan dari jenis data tersebut. Beberapa jenis dari tipe data dalam pemrograman yaitu Integer, Float, String, Boolean.

Tipe data Integer atau bilangan bulat adalah tipe data numerik yang berbentuk bilangan bulat atau angka bulat yang tidak memiliki tanda koma (.), seperti 1, 30, 250, dll. Tipe data ini juga mengenal bilangan negatif seperti -5, -47, -120, dll.

Tipe data Float atau bilangan pecahan adalah tipe data numerik berbentuk bilangan desimal atau angka yang memiliki tanda koma yang memiliki perhitungan yang lebih detail, seperti 0,5; 21,7; -15,3.

Tipe data String adalah kumpulan dari karakter dan data yang dapat digunakan untuk menulis teks. String juga dapat berisi simbol dan angka, namun akan selalu diklasifikasikan sebagai teks. Ciri dari tipe data ini selalu diawali dan diakhiri tanda petik (' ') atau (" ").

Tipe data Boolean adalah tipe data yang digunakan untuk menyatakan nilai benar dan nilai salah. nilai salah (false) dipresentasikan dengan angka 0 dan nilai benar (true) dipresentasikan dengan angka 1. Boolean biasanya digunakan untuk mempertimbangkan dua nilai seperti lebih besar, lebih kecil, kurang dari, dll.

2. Algoritma merupakan metode atau tahapan-tahapan yang digunakan untuk menyelesaikan atau memecahkan suatu permasalahan. Algoritma juga didefinisikan sebagai alur pemikiran dari pemecahan masalah, terdiri atas langkah-langkah sistematis, misal seperti menghitung deret angka, bunga kredit, dll.

Jenis-jenis algoritma:

1. Algoritma Sekuensial

Algoritma sekuensial adalah algoritma yang langkah-langkahnya lurus urut dari awal sampai akhir.

2. Algoritma Perulangan

Algoritma perulangan adalah algoritma yang dalam menjalankan langkah tertentu dikerjakan secara looping atau berulang-ulang.

3. Algoritma Percabangan atau Bersyarat

Algoritma bersyarat adalah algoritma yang mengerjakan langkah selanjutnya jika syarat pada langkah sebelumnya telah terpenuhi.

Bagaimana menulis algoritma

Terdapat beberapa cara untuk menulis algoritma yaitu pseudocode, flowchart, dan kalimat deskriptif.

pseudocode atau kode palsu adalah deskripsi dari suatu algoritma pemrograman yang menggunakan struktural dari suatu bahasa pemrograman.

Flowchart atau bagan alir adalah bagan dari langkah-langkah pemecahan suatu masalah yang berbentuk diagram dan dihubungkan dengan tanda panah.

Kalimat deskriptif atau notasi alami adalah algoritma yang dituliskan dalam bentuk instruksi kalimat deskriptif menggunakan bahasa yang jelas.

Contoh Tipe Data

```
print(type(5))
print(type(1.5))
print(type('a'))
print(type(True))

<class 'int'>
<class 'float'>
<class 'str'>
<class 'bool'>
```

Contoh Algoritma

```
# Mengubah suhu dari Celcius ke Fahrenheit - Algoritma Sekuensial

C=float(input('suhu Celcius = '))
F=(9/5*C)+32
print('Konversi Fahrenheit = ',F)

suhu Celcius = 100
```

```
Konversi Fahrenheit = 212.0
```

```
# Menyebutkan urutan bilangan ganjil dengan angka maksimal - Algoritma Perulangan
```

```
number=int(input('masukkan angka maks = '))
for i in range(number):
    if i%2==1:
        temp=temp+i
        print('bilangan ganjil=',i)
```

```
masukkan angka maks = 10
bilangan ganjil= 1
bilangan ganjil= 3
bilangan ganjil= 5
bilangan ganjil= 7
bilangan ganjil= 9
```

```
# Mencari bilangan terbesar dari 3 bilangan - Algoritma Pencabangan
```

```
a=int(input('a = '))
b=int(input('b = '))
c=int(input('c = '))
if a>b :
    max=a
else :
    max=b
if max>c :
    print('bilangan paling besar adalah',max)
else :
    print('bilangan paling besar adalah',c)

a = 7
b = 5
c = 9
bilangan paling besar adalah 9
```

▼ 1.2 Algoritma-1

Tuliskan algoritma (tiap tahap, diberikan nomor tahapan) untuk menyelesaikan Persamaan Matematika :

$$y = \sum_{i=1}^n (i^2 + 2i)$$

, dimana n adalah inputan dari user

▼ Tahapan

1. buat variabel n dengan n merupakan inputan dari user
2. lakukan looping menggunakan for dengan range mulai dari 1 sampai n+1
3. tulis rumus sigma tersebut dalam variabel y
4. print y atau hasil dari rumus tersebut

5. lalu buat variabel i nilainya bertambah 1 dalam setiap perulangan
6. buat variabel temp lalu temp ditambah nilai dari y dalam setiap perulangan untuk menghitung hasil
7. print hasil keseluruhan perulangan dengan temp diluar syntax for

```
n=int(input('n = '))
temp=0
```

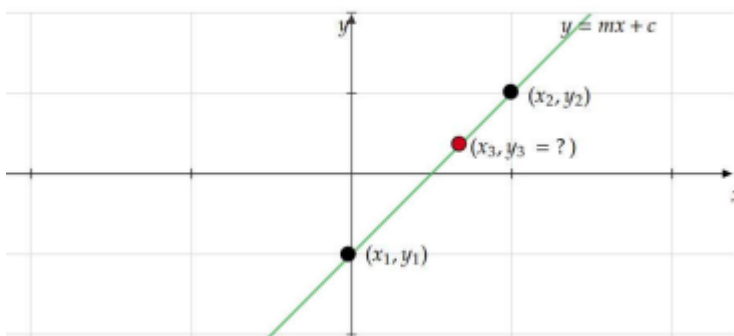
```
for i in range(1,n+1):
    y=i**2+2*i
    print(y)
    i=i+1
    temp=temp+y
print('y = ',temp)
```

```
n = 5
3
8
15
24
35
y = 85
```

▼ 1.3 Algoritma-2

Tuliskan algoritma (tiap tahap, diberikan nomor tahapan) untuk permasalahan berikut ini :

1. Terdapat (Diketahui) dua buah titik, yaitu (x_1, y_1) dan (x_2, y_2)
2. Jika terdapat titik baru yang hanya diketahui nilai koordinat x nya, misal (x_3, \dots) , maka cari nilai y_3
3. Petunjuk : Gunakan Persamaan Garis seperti yang ditunjukkan pada Gambar 1, dan $m = \Delta y / \Delta x$



Gambar 1: Persamaan Garis $y = mx + c$

Tahapan

1. diketahui titik (x_1, y_1) dan (x_2, y_2)
2. masukkan ke rumus $y - y_1 / y_2 - y_1 = x - x_1 / x_2 - x_1$

3. dari rumus tersebut akan menghasilkan $ax+by+c=0$
4. lalu substitusi nilai x_3 yang diketahui ke persamaan tersebut
5. nilai dari y_3 tersebut akan dapat ditemukan

▼ 1.4 Algoritma-3

Tulis tahapan-tahapan (algoritma) untuk menampilkan sejumlah n bilangan ganjil, dimana n adalah inputan dari user, dan jumlahkan semua bilangan ganjil tersebut.

▼ Tahapan

1. buat variabel a sebagai angka awal dimulai dari 0
2. buat variabel counter sebagai angka awal dimulainya perulangan dimulai dari 1
3. buat variabel temp untuk nantinya menghitung total
4. buat variabel n dengan n inputan dari user
5. buat perulangan menggunakan while dengan $\text{counter} \leq n$ yang artinya perulangan akan terus berlanjut selama nilai counter masih kurang dari sama dengan nilai n
6. buat percabangan dengan if. lalu jika nilai a di modulus 2 adalah 1 maka print a tersebut, lalu counter bertambah 1 dan temp bertambah nilai a .
7. syntax akan dilakukan jika if tersebut bernilai True
8. buat variabel a bertambah 1 di setiap perulangan, letakkan diluar if agar tetap bertambah di setiap perulangan/tidak dipengaruhi if
9. print jumlah dengan variabel temp yang tadi dibuat, letakkan diluar while agar output yang keluar adalah jumlah akhir dari semua perulangan.

```
a=0
counter=1
temp=0

n=int(input('n = '))

while counter<=n:
    if a%2==1:
        print(a)
        counter+=1
        temp+=a
    a+=1
print('jumlah =',temp)

n = 5
1
3
5
7
9
jumlah = 25
```

▼ 1.5 Algoritma-4

Tulis algoritma untuk menampilkan sejumlah bilangan dan total jumlah dari bilangan-bilangan tersebut, berdasarkan persamaan deret aritmatika.

$$U_n = a + (n - 1)b$$

$$S_n = n/2(2a + (n - 1)b)$$

dimana

U_n : suku ke-n (n input dari user)

a : suku awal(input dari user)

b : beda(input dari user), and

S_n : Total jumlah seluruh suku, mulai dari suku pertama sampai suku ke-n

▼ Tahapan

1. buat variabel nMax sebagai variabel suku terakhir dengan nMax inputan dari user
2. buat variabel a dengan a inputan dari user
3. buat variabel b dengan b inputan dari user
4. buat variabel n=1 sebagai suku pertama untuk nanti dimasukkan pada rumus saat proses looping
5. buat looping menggunakan syntax for dengan range mulai dari angka 1 dan angka terakhirnya adalah nilai nMax+1
6. tulis rumus deret tersebut ke dalam variabel U_n
7. print suku ke-n dan nilainya adalah hasil dari U_n
8. buat nilai dari variabel n bertambah 1 disetiap perulangan
9. tulis rumus S_n diluar for/looping, ubah n dalam rumus menjadi nMax agar yang masuk rumus adalah suku n ke terakhirnya
10. print hasil S_n

```
nMax=int(input('n='))
a=int(input('a='))
b=int(input('b='))
n=1

for i in range(1,nMax+1):
    Un=a+(n-1)*b
    print('suku ke-',n,'adalah',Un)
    n+=1

Sn=nMax/2*(2*a+(nMax-1)*b)
print('jumlah seluruh suku dari suku pertama sampai suku ke-',nMax,'adalah',Sn)

n=3
```

```

a=4
b=6
suku ke- 1 adalah 4
suku ke- 2 adalah 10
suku ke- 3 adalah 16
jumlah seluruh suku dari suku pertama sampai suku ke- 3 adalah 30.0

```

▼ 2 Implementasi

▼ 2.1 Progran Bilangan Ganjil

Konversi algoritma yang sudah ditulis sebelumnya dengan menggunakan bahasa Pemrograman Python.

Contoh hasil eksekusi dapat dilihat di Gambar.2a or Gambar.2b :

```

masukkan jumlah bilangan = 5
Bilangan ke- 1 : 1
Bilangan ke- 2 : 3
Bilangan ke- 3 : 5
Bilangan ke- 4 : 7
Bilangan ke- 5 : 9
Total = 25

```

(a) n = 5

```

masukkan jumlah bilangan = 10
Bilangan ke- 1 : 1
Bilangan ke- 2 : 3
Bilangan ke- 3 : 5
Bilangan ke- 4 : 7
Bilangan ke- 5 : 9
Bilangan ke- 6 : 11
Bilangan ke- 7 : 13
Bilangan ke- 8 : 15
Bilangan ke- 9 : 17
Bilangan ke- 10 : 19
Total = 100

```

(b) n = 10

Gambar 2: Bilangan Ganjil dan Jumlahnya

```

a=0
counter=1
temp=0
n=1

jum=int(input('masukkan jumlah bilangan = '))

while counter<=jum:
    if a%2==1:
        print('Bilangan ke- ',n,':',a )
        counter+=1
        temp+=a
        n+=1
    a+=1
print('Total =',temp)

```

```

masukkan jumlah bilangan = 10
Bilangan ke- 1 : 1
Bilangan ke- 2 : 3
Bilangan ke- 3 : 5
Bilangan ke- 4 : 7
Bilangan ke- 5 : 9
Bilangan ke- 6 : 11
Bilangan ke- 7 : 13
Bilangan ke- 8 : 15
Bilangan ke- 9 : 17

```

Bilangan ke- 10 : 19
Total = 100

▼ 2.2 Program Deret Aritmatika

Konversi algoritma yang sudah ditulis sebelumnya dengan menggunakan bahasa Pemrograman Python, seperti yang ditunjukkan pada Gambar 3a or Gambar3b.

<pre> masukkan nilai n = 3 masukkan nilai a = 4 masukkan nilai b = 6 u- 1 = 4 u- 2 = 10 u- 3 = 16 sn= 30.0 </pre>	<pre> masukkan nilai n = 5 masukkan nilai a = 6 masukkan nilai b = 7 u- 1 = 6 u- 2 = 13 u- 3 = 20 u- 4 = 27 u- 5 = 34 sn= 100.0 </pre>
(a) $n = 3, a = 4, b = 6$	(b) $n = 5, a = 6, b = 7$

Gambar 3: Arithmetic Series

```

nMax=int(input('masukkan nilai n='))
a=int(input('a='))
b=int(input('b='))
n=1

```

```

for i in range(1,nMax+1):
    Un=a+(n-1)*b
    print('u- ',n,'=',Un)
    n+=1

```

```

Sn=nMax/2*(2*a+(nMax-1)*b)
print('sn=',Sn)

```

```

masukkan nilai n=3
a=4
b=6
u- 1 = 4
u- 2 = 10
u- 3 = 16
sn= 30.0

```


[Produk berbayar Colab](#) - [Batalan kontrak di sini](#)

