



Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

NIM	71231058
Nama Lengkap	Michael Chandra Mahanaim
Minggu ke / Materi	06 / Percabangan dan Perulangan Kompleks

SAYA MENYATAKAN BAHWA LAPORAN PRAKTIKUM INI SAYA BUAT DENGAN USAHA SENDIRI TANPA MENGGUNAKAN BANTUAN ORANG LAIN. SEMUA MATERI YANG SAYA AMBIL DARI SUMBER LAIN SUDAH SAYA CANTUMKAN SUMBERNYA DAN TELAH SAYA TULIS ULANG DENGAN BAHASA SAYA SENDIRI.

SAYA SANGGUP MENERIMA SANKSI JIKA MELAKUKAN KEGIATAN PLAGIASI, TERMASUK SANKSI TIDAK LULUS MATA KULIAH INI.

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA

BAGIAN 1: MATERI MINGGU INI (40%)

MATERI 1: Percabangan Kompleks

Percabangan kompleks adalah dimana 2 atau lebih percabangan digunakan dalam sebuah kode untuk membuat sebuah kondisi spesifik. Contoh percabangan kompleks adalah seperti berikut:

```
n = 0
m = 2
if n == 1:
    if m == 2:
        print("Yes")
    else:
        print("No")
else:
    print("Definitely no")
```

Gambar contoh percabangan komplks

Dari contoh diatas diberi 2 variabel: n dan m. Dalam percabangan diberi kondisi jika n == 1 maka akan masuk if dan diberi kondisi lagi jika m == 2 maka print("Yes"). Dengan 2 percabangan ini sudah dapat muncul 3 kemungkinan output yang berdasarkan dalam 2 input saja. Ada juga yang lebih rumit jika menggunakan else dari if pertama sebagai cabang lagi seperti di bawah ini:

```
n = 0
m = 2
if n == 1:
    if m == 2:
        print("Yes")
    else:
        print("No")
else:
    if m != 4:
        print("Absolutely No")
    else:
        print("Maybe Yes")
```

Gambar penggunaan lebih rumit else sebagai tambahan percabangan

Dengan menggunakan else dari if pertama sebagai kondisi baru yaitu: $m \neq 4$ maka `print("Absolutely No")` dan `else print("Maybe Yes")`. Maka kemungkinan output sudah ada 4. jika diteruskan maka kemungkinan output dapat bertumpuk lebih banyak.

MATERI 2: Perulangan Kompleks

Perulangan kompleks sama seperti materi sebelumnya yaitu Struktur perulangan kompleks dan membahas tentang `break` dan `continue`. Sebagai recap `break` dan `continue` digunakan untuk Memotong dan Men-skip sebuah perulangan. Contohnya sebagai berikut:

```
for i in range(0, 100):  
    if i == 10:  
        continue  
    elif i == 20:  
        break  
    print(i, end=" ")
```

0 1 2 3 4 5 6 7 8 9 11 12 13 14 15 16 17 18 19

Contoh penggunaan `break` dan `continue` serta outputnya

Dari kode tersebut dapat dilihat jika $i == 10$ maka `continue` akan digunakan dan kode dibawahnya akan di skip dan itu terlihat dari outputnya yang tidak menampilkan angka 10. Sedangkan `break` akan full stop memotong for loop saat $i == 20$ walaupun for loop seharusnya berjalan terus dari 1 sampai 100. Dari contoh tersebut diketahui bahwa `break` dan `continue` sangatlah berguna untuk memotong sebuah jika suatu kondisi terpenuhi ataupun melompati sebuah langkah jika sebuah kondisi terpenuhi.

MATERI 3: Perulangan Bertingkat

Perulangan bertingkat seperti namanya adalah perulangan di dalam perulangan. Dapat berupa campuran for dan while loop ataupun eksklusif salah satu saja. Contohnya sebagai berikut:

```
for i in range(10):
    for j in range(i):
        print(i, end=" ")
    print("")
```

Contoh gambar penggunaan perulangan bertingkat

Dari contoh di atas ada 2 perulangan yang digunakan, yaitu perulangan luar dan dalam. Tugas perulangan luar adalah untuk membuat baris baru sebanyak 10 kali, sedangkan tugas perulangan dalam adalah untuk mengeluarkan output berupa angka i sebanyak i kali secara mendatar. Output yang dihasilkan adalah seperti berikut ini:

```
1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
6 6 6 6 6 6
7 7 7 7 7 7 7
8 8 8 8 8 8 8 8
9 9 9 9 9 9 9 9 9
```

Gambar contoh output dari perulangan bertingkat

Karena bertingkat, maka outputnya juga bertingkat. perulangan dalam akan print i sebanyak i kali dan perulangan luar bertugas sebagai patokan perulangan perulangan dalam dalam print i nya. Perulangan bertingkat ini sangat berguna untuk membuat output yang berbentuk suatu gambar ataupun sebuah pola seperti piramida ataupun bentuk kompleks lain seperti jajargenjang.

BAGIAN 2: LATIHAN MANDIRI (60%)

SOAL 1

A. Source Code

```
Welcome | Question1.py U x | Question2.py U | Question3.py U | Sr
Question1.py > primechk
1 def primechk(n):
2     if n < 2:
3         return False
4     for i in range(2, round(n**(1/2))+1):
5         if n % i == 0:
6             return False
7     return True
8 def primebelow(n):
9     if n <= 2:
10        print("No prime below 2")
11        return None
12
13    for i in range(n, 1, -1):
14        if primechk(i):
15            return i
16
17
18 num = input("Input a number: ")
19 try:
20     num = int(num)
21 except:
22     print("Please use numbers, not words")
23     exit()
24 prime = primebelow(num)
25 print(f"Prime below {num} is {prime}")
```

B. Output Result

PS C:\Users\micha\Doc s/College/2SMT/TPAP1/ Input a number: 68 Prime below 68 is 67	PS C:\Users\micha\Doc s/College/2SMT/TPAP1/ Input a number: 99 Prime below 99 is 97
--	--

C. Explanation

Kode ini adalah kode yang digunakan untuk mencari bilangan prima bawah terdekat dari nomor yang diinputkan oleh user. Kode ini menggunakan komponen: input, try except, fungsi, percabangan, dan perulangan. Fungsi primechk() menggunakan percabangan dan perulangan untuk mengecek faktor yang dimiliki angka input user, karena faktor terpecah jadi 2 (faktor besar dan faktor kecil), maka akan lebih efisien jika mengecek faktor kecil saja dari input user dan

menggunakan akar bilangan itu sebagai pemisah faktor kecil dan faktor besar. `primebelow()` hanya digunakan untuk mengecek jika input kurang dari 2 dan menghitung kebawah untuk mengecek angka prima terdekat serta mengulangi fungsi `primechk()`. `Try except` digunakan untuk mengatasi kesalahan input user.

SOAL 2

A. Source Code

```
Welcome | Question1.py U | Question2.py U X | Question3.py U | S
Question2.py > ...
1 def factorial(n):
2     r = 1
3     for i in range(n, 1, -1):
4         r *= i
5     return r
6 def printout(n):
7     for j in range(n, 0, -1):
8         result = factorial(j)
9         print(result, end="")
10        for i in range(j, 0, -1):
11            print(f" {i}", end="")
12        print("")
13
14 num = input("Input a number: ")
15
16 try:
17     num = int(num)
18 except:
19     print("Please use numbers, not words")
20     exit()
21
22 printout(num)
```

B. Output Result

```
PS C:\Users\micha\Documents\College/2SMT/TPAP1/tugas>
Input a number: 9
362880 9 8 7 6 5 4 3 2 1
40320 8 7 6 5 4 3 2 1
5040 7 6 5 4 3 2 1
720 6 5 4 3 2 1
120 5 4 3 2 1
24 4 3 2 1
6 3 2 1
2 2 1
1 1

PS C:\Users\micha\Documents\College/2SMT/TPAP1/tugas>
Input a number: 7
5040 7 6 5 4 3 2 1
720 6 5 4 3 2 1
120 5 4 3 2 1
24 4 3 2 1
6 3 2 1
2 2 1
1 1
```

C. Explanation

Kode ini menggunakan perulangan dan percabangan kompleks untuk membuat output berupa hasil faktorial diikuti dengan angka yang digunakan untuk hasil faktorial mulai dari angka input user sampai angka 1. Kode ini menggunakan komponen: input, try except, fungsi, dan perulangan saja. fungsi factorial() digunakan untuk menghitung hasil faktorial angka yang diinginkan dan fungsi printout() digunakan untuk mengeluarkan output sesuai dengan output result. Output menggunakan for i in range() untuk output mendatar dan for j in range() untuk output menurun. Try except seperti biasa digunakan untuk menangani kesalahan input user.

SOAL 3

A. Source Code

```
Welcome X Question1.py U Question2.py U Question3.py U X S
Question3.py > printout
1 def printout(x, y):
2     count = 1
3     for i in range(y):
4         for j in range(x):
5             print(count, end=" ")
6             count += 1
7         print("")
8
9 height = input("Input height: ")
10 width = input("Input width: ")
11
12 try:
13     height = int(height)
14     width = int(width)
15 except:
16     print("Please use numbers, not words")
17     exit()
18
19 printout(width, height)
```

B. Output Result

```
PS C:\Users\micha
s/College/2SMT/TP
Input height: 5
Input width: 5
1 2 3 4 5
6 7 8 9 10
11 12 13 14 15
16 17 18 19 20
21 22 23 24 25

PS C:\Users\micha\Documents\Co
s/College/2SMT/TPAP1/tugaspra
Input height: 3
Input width: 10
1 2 3 4 5 6 7 8 9 10
11 12 13 14 15 16 17 18 19 20
21 22 23 24 25 26 27 28 29 30
```

C. Explanation

Kode ini menggunakan 2 input user berupa lebar dan tinggi untuk mengeluarkan output seperti pada Output Result. Kode ini memiliki komponen: input, try except, fungsi, dan perulangan. `for i in range()` digunakan untuk output tinggi dan `for j in range()` digunakan untuk output mendatar. Try except sekali lagi digunakan untuk mengatasi kesalahan input user.