



Laporan Praktikum Algoritma & Pemrograman

Semester Genap 2024/2025

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.

NIM	<71241079>
Nama Lengkap	<Natalie Neysa Jessica Soesanto>
Minggu ke / Materi	06 / Struktur Kontrol Perulangan

PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN DUTA WACANA
YOGYAKARTA
2025

BAGIAN 1: MATERI MINGGU INI (40%)

Pada bagian ini, tuliskan kembali semua materi yang telah anda pelajari minggu ini. Sesuaikan penjelasan anda dengan urutan materi yang telah diberikan di saat praktikum. Penjelasan anda harus dilengkapi dengan contoh, gambar/ilustrasi, contoh program (source code) dan outputnya. Idealnya sekitar 5-6 halaman.

MATERI 1

Definisi Perulangan

Jalannya suatu program dapat diatur secara sekuensial, percabangan, perulangan, maupun kombinasi dari ketiganya. Pengaturan tersebut biasa disebut sebagai struktur kontrol.

Perulangan digunakan apabila dalam program diperlukan untuk :

- Melakukan suatu hal yang sama beberapa kali
- Melakukan suatu hal secara bertahap, dimana setiap tahap sebenarnya memiliki langkah yang sama
- Mengakses sekumpulan data dalam suatu struktur data, misalnya : List, Tuple, Queue, Stack, dan beberapa struktur data lainnya

Di bahasa pemrograman python biasa menggunakan perulangan “for” dan “while”, namun bisa juga dengan cara rekursif.

MATERI 2

Bentuk Perulangan For

Perulangan “for” biasanya digunakan saat jumlah perulangan sudah diketahui sejak awal. Proses pembacaannya juga berurutan. Misal ada 10 file berbeda pembacaannya dimulai dari file pertama hingga terakhir, contoh :

Source Code :

```
1  for i in range(1, 101, 2):
2      print(i)
3
4  for i in range(1, 101):
5      print(i, "hello word!")
```

Output :

```

PS C:\Users\ASUS\OneDrive\Desktop> python3 ontoh1.py
1
3
5
7
9
11
13
15
17
19
21
23
25
27
29
31
33
35
37
39
41
43
45
47
49
51
53
55
57
59
61
63
65
67
69
71
73
75
77
79
81
83
85
87
89
91
93
95
97
99
101 hello word!
102 hello word!
103 hello word!
104 hello word!
105 hello word!
106 hello word!
107 hello word!
108 hello word!
109 hello word!
110 hello word!
111 hello word!
112 hello word!
113 hello word!
114 hello word!
115 hello word!
116 hello word!
117 hello word!
118 hello word!
119 hello word!
120 hello word!
121 hello word!
122 hello word!
123 hello word!
124 hello word!
125 hello word!
126 hello word!
127 hello word!
128 hello word!
129 hello word!
130 hello word!
131 hello word!
132 hello word!
133 hello word!
134 hello word!
135 hello word!
136 hello word!
137 hello word!
138 hello word!
139 hello word!
140 hello word!
141 hello word!
142 hello word!
143 hello word!
144 hello word!
145 hello word!
146 hello word!
147 hello word!
148 hello word!
149 hello word!
150 hello word!
151 hello word!
152 hello word!
153 hello word!
154 hello word!
155 hello word!
156 hello word!
157 hello word!
158 hello word!
159 hello word!
160 hello word!
161 hello word!
162 hello word!
163 hello word!
164 hello word!
165 hello word!
166 hello word!
167 hello word!
168 hello word!
169 hello word!
170 hello word!
171 hello word!
172 hello word!
173 hello word!
174 hello word!
175 hello word!
176 hello word!
177 hello word!
178 hello word!
179 hello word!
180 hello word!
181 hello word!
182 hello word!
183 hello word!
184 hello word!
185 hello word!
186 hello word!
187 hello word!
188 hello word!
189 hello word!
190 hello word!
191 hello word!
192 hello word!
193 hello word!
194 hello word!
195 hello word!
196 hello word!
197 hello word!
198 hello word!
199 hello word!
200 hello word!
PS C:\Users\ASUS\OneDrive\Desktop>

```

MATERI 3

Bentuk Perulangan While

Bentuk while biasanya digunakan pada kondisi dimana jumlah perulangan belum diketahui sebelumnya, contoh :

Source Code :

```

1  i = 1
2  while i < 11:
3      print(i, "hello world!")
4      i = i + 1

```

Output :

```

PS C:\Users\ASUS\OneDrive\Documents\Skull\Sem 2> python3 ontoh2.py
1 hello world!
2 hello world!
3 hello world!
4 hello world!
5 hello world!
6 hello world!
7 hello world!
8 hello world!
9 hello world!
10 hello world!
PS C:\Users\ASUS\OneDrive\Documents\Skull\Sem 2>

```

Contoh lainnya :

Source Code :

```

1 bilangan = 0
2 genap = False
3 while genap == False:
4     bilangan = int(input("Masukkan bilangan : "))
5     if bilangan % 2 == 0:
6         genap = True
7 print(bilangan, "yang anda masukkan adalah bilangan genap")

```

Output :

```

PS C:\Users\ASUS\OneDrive\Documents\Skull\Sem 2> python3 ontoh3.py
Masukkan bilangan : 1
Masukkan bilangan : 2
2 yang anda masukkan adalah bilangan genap
PS C:\Users\ASUS\OneDrive\Documents\Skull\Sem 2>

```

Penjelasan :

Program memaksa pengguna untuk memasukkan bilangan genap, jika yang dimasukkan pengguna bilangan ganjil maka tidak muncul apa-apa, jika yang dimasukkan bilangan genap maka program akan menampilkan output seperti di atas.

MATERI 4

Penggunaan Break dan Continue

Perulangan dapat dikontrol dengan menggunakan “break” dan “continue”. Break digunakan untuk menghentikan perulangan, sedangkan continue digunakan untuk melanjutkan perulangan ke iterasi berikutnya. Contoh :

Break

Source Code :

```

1  for i in range(1, 11):
2      if i == 5:
3          break
4      else:
5          print(i)
6  print("selesai")

```

Output :

```

● PS C:\Users\AS
  ontoh4.py"
1
2
3
4
selesai
○ PS C:\Users\AS

```

Continue

Source Code :

```

1  for i in range(1, 11):
2      if i == 5:
3          continue
4      else:
5          print(i)
6  print("selesai")

```

Output :

```

● PS C:\Users\AS
  ontoh5.py"
1
2
3
4
6
7
8
9
10
selesai
○ PS C:\Users\AS

```

MATERI 5

Konversi Bentuk For Menjadi Bentuk While

Bentuk perulangan for sebagian besar dapat dikonversi menjadi bentuk while. Beberapa hal yang ada di bentuk for dan while adalah sebagai berikut :

- Harus ada nilai awal, untuk memulai perulangan

- Harus ada nilai akhir, untuk mengakhiri perulangan
- Harus ada langkah, agar iterasi dari nilai awal dapat terus berjalan sampai mencapai nilai akhir

Contoh :

Source Code :

```
1 for i in range(1, 11):
2     print(i)
3 i = 1
4 while i <= 10:
5     print(i)
6     i = i + 1
```

Output :

1

MATERI 6

Kegiatan Praktikum : Deret Bilangan Fibonacci

Deret bilangan fibonacci adalah deret bilangan yang tersusun dari penjumlahan dua suku sebelumnya dari deret bilangan tersebut. Biasanya deret bilangan fibonacci dimulai dari 1, 1, 2, 3, ... dan seterusnya.

Source Code :

```

1 def fibo(batas):
2     bil1 = 1
3     bil2 = 1
4     # tampilkan dua suku fibonacci pertama
5     if bil1 < batas:
6         print(bil1, end='\t')
7         print(bil2, end='\t')
8     # suku-suku berikutnya dari bil1 + bil2
9     suku_baru = bil1 + bil2
10    while suku_baru < batas:
11        print(suku_baru, end='\t')
12    # geser bil1 dan bil2
13    bil1 = bil2
14    bil2 = suku_baru
15    # hitung lagi suku berikutnya
16    suku_baru = bil1 + bil2
17
18    # program utama
19    batas = int(input('Masukkan batas dari deret fibonacci: '))
20    fibo(batas)

```

Output :

```
PS C:\Users\ASUS\OneDrive\Dokumen\Skull\Sem 2\PrakAlpro\Modular Programming> python -u "c:\Users\ASUS\OneDrive\Dokumen\Skull\Sem
ontoh7.py"
Masukkan batas dari deret fibonacci: 100
1      1      2      3      5      8      13      21      34      55      89
PS C:\Users\ASUS\OneDrive\Dokumen\Skull\Sem 2\PrakAlpro\Modular Programming> python -u "c:\Users\ASUS\OneDrive\Dokumen\Skull\Sem
ontoh7.py"
Masukkan batas dari deret fibonacci: 1000
1      1      2      3      5      8      13      21      34      55      89      144      233      377      610      987
PS C:\Users\ASUS\OneDrive\Dokumen\Skull\Sem 2\PrakAlpro\Modular Programming>
```

Konvergen (Jika ganjil, maka kalikan dengan tiga, lalu tambah 1. Jika genap, bagi dengan 2)

Source Code :

```
1 def konvergen(start):
2     suku = start
3     while suku != 1:
4         print(suku)
5         if suku % 2 == 0:
6             suku = suku // 2
7         else:
8             suku = suku * 3 + 1
9     # bagian utama program
10    start = int(input('Masukkan suku pertama dari deret konvergen: '))
11    konvergen(start)
```

Output :

```
PS C:\Users\ASUS\OneDrive\Dokumen\Skull\Sem 2\Prak
ontoh8.py"
Masukkan suku pertama dari deret konvergen: 12
12
6
3
10
5
16
8
4
2
PS C:\Users\ASUS\OneDrive\Dokumen\Skull\Sem 2\Prak
ontoh8.py"
Masukkan suku pertama dari deret konvergen: 5
5
16
8
4
2
PS C:\Users\ASUS\OneDrive\Dokumen\Skull\Sem 2\Prak
```

Penjelasan :

Program tersebut meminta input dari user, jika user memasukkan bilangan ganjil maka program otomatis akan mengkalikan 3 dari input user dan tambah 1, jika input yang dimasukkan user itu genap maka akan dibagi 2, jika hasilnya ganjil maka akan kembali ke pernyataan pertama begitu juga sebaliknya sampai angka input mentok di "2".

BAGIAN 2: LATIHAN MANDIRI (60%)

Pada bagian ini anda menuliskan jawaban dari soal-soal Latihan Mandiri yang ada di modul praktikum. Jawaban anda harus disertai dengan source code, penjelasan dan screenshot output.

SOAL 1

Source Code :

```
1  def perkalian(a, b):
2      hasil = 0
3      for i in range(a):
4          hasil += b
5      return hasil
6
7  def x(a, b):
8      hasil = perkalian(a, b)
9      P = " + ".join ([str(b) for i in range(a)])
10     print(f"{a} x {b} = {P} = {hasil}")
11
12 def main():
13     tes_perkalian = [(6, 5), (7, 10)]
14
15     for a1, a2 in tes_perkalian:
16         x(a1, a2)
17
18 if __name__ == "__main__":
19     main()
```

Output :

```
PS C:\Users\ASUS\OneDrive\Dokumen\Skull\Sem 2\Praktikum\Latihan Mandiri\Soal 1\soal1.py"
● 6 x 5 = 5 + 5 + 5 + 5 + 5 + 5 = 30
  7 x 10 = 10 + 10 + 10 + 10 + 10 + 10 + 10 = 70
○ PS C:\Users\ASUS\OneDrive\Dokumen\Skull\Sem 2\Praktikum\Latihan Mandiri\Soal 1\soal1.py"
```

Penjelasan :

Program tersebut berfungsi untuk menghitung perkalian 2 bilangan. Misal $6 \times 5 = 5 + 5 + 5 + 5 + 5 + 5 = 30$. Begitu juga dengan 7×10 .

SOAL 2

Source Code :


```

1 def ganjil(bawah, atas):
2     deret = []
3     if bawah < atas:
4         for i in range(bawah, atas + 1):
5             if i % 2 != 0:
6                 deret.append(i)
7     else:
8         for i in range(bawah, atas - 1, -1):
9             if i % 2 != 0:
10                deret.append(i)
11     return deret
12
13 def main():
14     x = [(10, 30), (97, 82)]
15
16     for bawah, atas in x:
17         hasil = ganjil(bawah, atas)
18
19         if bawah < atas:
20             print(f"bawah = {bawah}, atas = {atas}. Karena bawah < atas, berarti dari kecil ke besar, maka hasilnya adalah: {' '.join(map(str, hasil))}.")
21         else:
22             print(f"bawah = {bawah}, atas = {atas}. Karena bawah > atas, berarti dari besar ke kecil, maka hasilnya adalah: {' '.join(map(str, hasil))}.")
23
24 if __name__ == "__main__":
25     main()

```

Output :

```

PS C:\Users\ASUS\OneDrive\Dokumen\Skull\Sem 2\PrakAlpro\Modular Programming> python -u "c:\Users\ASUS\OneDrive\Dokumen\Skull\Sem 2\PrakAlp
oal\soal2.py"
o bawah = 10, atas = 30. Karena bawah < atas, berarti dari kecil ke besar, maka hasilnya adalah: 11, 13, 15, 17, 19, 21, 23, 25, 27, 29.
bawah = 97, atas = 82. Karena bawah > atas, berarti dari besar ke kecil, maka hasilnya adalah: 97, 95, 93, 91, 89, 87, 85, 83.
PS C:\Users\ASUS\OneDrive\Dokumen\Skull\Sem 2\PrakAlpro\Modular Programming>

```

Penjelasan :

Program tersebut berfungsi untuk menampilkan deret bilangan ganjil sesuai input yang ada di soal, jika bawah < atas maka urutannya dari kecil ke besar, jika bawah > atas maka urutannya dari besar ke kecil.

SOAL 3

Source Code :

```

1 def hitung_ips(jumlah_matkul):
2     total_sks = 0
3     total_bobot_nilai = 0
4
5     for i in range(1, jumlah_matkul + 1):
6         nilai = input(f"Nilai MK {i}: ").upper()
7
8         if nilai == "A":
9             bobot = 4
10        elif nilai == "B":
11            bobot = 3
12        elif nilai == "C":
13            bobot = 2
14        elif nilai == "D":
15            bobot = 1
16        else:
17            print("Input tidak valid. Masukkan nilai A, B, C, atau D.")
18            continue
19
20        sks = 6
21        total_sks += sks
22        total_bobot_nilai += bobot * sks
23
24    ips = total_bobot_nilai / total_sks
25    return ips
26
27 def main():
28     print("Program penghitung IPS Mahasiswa")
29     jumlah_matkul = int(input("Berapa jumlah mata kuliah? "))
30     ips = hitung_ips(jumlah_matkul)
31     print(f"Nilai IPS anda semester ini {ips:.2f}")
32
33 if __name__ == "__main__":
34     main()

```

Output :

```

PS C:\Users\ASUS\OneDrive\Dokumen\Sku
● Program penghitung IPS Mahasiswa
Berapa jumlah mata kuliah? 6
Nilai MK 1: A
Nilai MK 2: B
Nilai MK 3: C
Nilai MK 4: A
Nilai MK 5: D
Nilai MK 6: C
Nilai IPS anda semester ini 2.67
○ PS C:\Users\ASUS\OneDrive\Dokumen\Sku

```

Penjelasan :

Program penghitung indeks mahasiswa ini untuk memunculkan masing-masing “Nilai MK” menggunakan “for i in range”, jadi kita tidak perlu membuat code yang lebih panjang untuk input “Nilai MK” nya. Program ini berfungsi untuk menghitung IPS mahasiswa, dengan cara meminta input dari user, outputnya program ini bisa menghitung IPS user tersebut.

Github : <https://github.com/babydoll-05/Laprak-6.git>