



# Laporan Praktikum Algoritma dan Pemrograman

Semester Genap 2023/2024

<b>NIM</b>	<b>71230972</b>
<b>Nama Lengkap</b>	<b>Oktavian Christ Putranto</b>
<b>Minggu ke / Materi</b>	<b>03 / Kontrol percabangan</b>

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  
YOGYAKARTA  
2024

## Boolean Expression dan Operator

Boolean Expression sederhananya adalah kondisi benar dan salah atau sering disebut dengan True/False Dimana hal ini dapat membantu kita untuk menentukan jalan nya program sehingga bisa menampilkan 2/lebih kemungkinan output yang berbeda beda sesuai kondisional yang kita tentukan.

sebagai contoh sederhana apabila kita diminta untuk membuat sebuah program untuk sebuah toko baju yang menentukan syarat sebuah voucher diskon 30 % dapat dipakai jika minimum pembelian RP.100.000 Rupiah sehingga terdapat sebuah syarat yang harus dipenuhi yaitu pembelian minimumnya RP. 100.000 Rupiah:

```
voucher.py > ...  
1 pembelian=int(input("total belanjaan anda:"))  
2 if pembelian >= 100000:  
3     print("true")  
4 else:  
5     print("false")  
6
```

```
total belanjaan anda:100000  
true  
PS C:\Users\ASUS VivoBook\Downloads\UG_2> & "C:/Users/ASUS VivoBook/s64/mingw64/bin/python.exe" "c:/Users/ASUS VivoBook/  
total belanjaan anda:45000  
false
```

Dari code tersebut kita dapat menentukan 2 kondisi apakah pembelian minimal atau  $\geq 100.000$  (true/false) terpenuhi atau tidak yang mana hal ini akan memengaruhi hasil output yang dikeluarkan seperti contoh bila input yang dimasukan adalah 100.000 maka keluar hasil true yang berarti sudah memenuhi persyaratan dan bila kita input 45.000 akan keluar hasil false karena 45.000 tidak lebih dari 100.000 yang membuat nya tidak memenuhi persyaratan.

Untuk dapat membuat kondisional atau percabangan yang biasa dipakai dalam pemrograman maka kita harus bisa memahami operator yang tepat untuk tiap persyaratan tersebut seperti contohnya pada soal tadi kita diminta untuk membuat persyaratan berupa minimum pembelian maka operator yang kita pakai ialah  $\geq$  (lebih dari samadengan) karena ketika pembelian diangka 100.000 atau lebih dari 100.000 maka voucher dapat berlaku, selain tanda  $\geq$  masih ada banyak operator lain diantaranya ialah :

**Tabel 3.1: Operator-operator perbandingan (comparison).**

<b>Operator</b>	<b>Keterangan</b>
<code>x == y</code>	Apakah x sama dengan y?
<code>x != y</code>	Apakah x tidak sama dengan y?
<code>x &gt; y</code>	Apakah x lebih besar dari y?
<code>x &gt;= y</code>	Apakah x lebih besar atau sama dengan y?
<code>x &lt; y</code>	Apakah x lebih kecil dari y?
<code>x &lt;= y</code>	Apakah x lebih kecil atau sama dengan y?
<code>x is y</code>	Apakah x sama dengan y?
<code>x is not y</code>	Apakah x tidak sama dengan y?

Selain operator-operator tersebut ada juga operator logikal yang ditandai dengan **and**, **or**, **not** ketiga operator logikal ini kegunaannya sama seperti operator logika matematika. Dimana apabila ada 2 kondisional yang dihubungkan dengan operator logikal **and** maka ketika salah satu dari dua kondisional itu false keseluruhan kondisional menjadi false, bila dihubungkan dengan **or** maka ketika salah satu dari 2 kondisional bernilai true maka keseluruhan nya akan menjadi true untuk contoh penggunaan logikal operator dapat kita lihat sebagai berikut :

Dalam sebuah wahana kora-kora terdapat sebuah aturan dimana kora-kora hanya dapat dinaiki oleh penumpang dengan usia minimal 12 tahun dan tinggi badan minimal 115 cm.

Kedua persyaratan tersebut dapat dinyatakan dalam bentuk sebagai berikut:

`usia >= 12 and tinggi >= 115`

diskon diberikan kepada member atau jumlah pembelian lebih dari Rp. 500.000 persyaratan tersebut dapat ditulis dalam code menjadi seperti ini:

`member == true or pembelian > 500000`

Berikut adalah hal-hal yang perlu kita perhatikan saat menyusun bentuk boolean expression:

- Boolean expression hasilnya hanya ada dua, yaitu True atau False.
- Perhatikan kata-kata khusus seperti minimum, maksimum, tidak lebih dari, tidak kurang dari, tidak sama, tidak berbeda.
- Perhatikan dengan seksama dan tentukan variabel yang perlu dibandingkan dengan benar sesuai dengan permasalahan.

Contoh dari Boolean expression :

Tabel 3.2: Operator-operator perbandingan (comparison).

Contoh masalah	Boolean expression
Untuk lulus dibutuhkan IPK minimum 2.25	<code>ipk &gt;= 2.25</code>
Golden Button hanya diberikan untuk Youtuber dengan subscriber lebih dari 1 juta	<code>subscriber &gt; 1000000</code>
Pengendara dengan kecepatan lebih dari 90 km/jam akan mendapatkan tilang	<code>kecepatan &gt; 90</code>
Wahana Rollercoaster hanya bisa dinaiki oleh mereka yang tinggi badannya lebih dari 110 cm	<code>tinggi &gt; 110</code>
Nilai ujian Hanna adalah 75 sedangkan Robby mendapatkan nilai 75. Apakah nilai keduanya sama?	<code>hanna is robbby</code>
Junaedi memiliki 10 sepatu, Ricky punya 15 sepatu dan Arnold punya 20 sepatu. Apakah gabungan sepatu Junaedi dan Ricky lebih banyak dari sepatu milik Arnold?	<code>junaedi + ricky &gt; arnold</code>

## Bentuk bentuk percabangan

Percabangan pada Python secara umum dibagi menjadi tiga bentuk, yaitu: conditional, alternative dan chained conditional.

Conditional :

```
nilai=int(input("input nilai anda:"))

if nilai > 75:
    print("anda lulus dan dapat sertifikat")
```

```
input nilai anda:78
anda lulus dan dapat sertifikat
PS C:\Users\ASUS VivoBook\Downloads\UG_2> & "C:/Users/ASUS VivoBook/Documents/kuliah/msys64/mingw64/bin/python.exe" "c:/Users/ASUS VivoBook/Downloads/UG_2/percabangan.py"
input nilai anda:25
```

Apabila input nilai sesuai dengan kondisi yang ditetapkan maka akan muncul output sesuai yang ditentukan yaitu "anda lulus dan dapat sertifikat" sedangkan ketika input nilai tidak memenuhi kondisi yang ditentukan maka tidak terjadi apa apa/output tidak keluar.

Alternative:

```

percabangan.py > ...
1  nilai=int(input("input nilai anda:"))
2
3  √ if nilai > 75:
4      |   print("anda lulus dan dapat sertifikat")
5  √ else:
6      |   print("anda di Do oleh kampus")
7

```

```

input nilai anda:85
anda lulus dan dapat sertifikat
PS C:\Users\ASUS VivoBook\Downloads\UG_2> & "C:/Users/ASUS VivoBook/Documents/kuliah/msys64/mingw64/bin/python.exe" "c:/Users/ASUS VivoBook/Downloads/UG_2/percabangan.py"
input nilai anda:25
anda di Do oleh kampus
PS C:\Users\ASUS VivoBook\Downloads\UG_2>

```

Pada bentuk percabangan alternative maka ketika conditional yang ditentukan tidak terpenuhi maka akan mengeluarkan output yang berbeda seperti dalam contoh ketika input nilai sebesar 85 yang memenuhi conditional ( $\text{nilai} > 75$ ) maka output yang keluar ialah "anda lulus dan dapat sertifikat", sedangkan ketika input nilai sebesar 25 (tidak memenuhi conditional  $\text{nilai} > 75$ ) maka output yang dikeluarkan ialah "anda di DO oleh kampus".

Chained conditional:

```

percabangan.py > ...
1  nilai=int(input("input nilai anda:"))
2  if nilai > 75:
3      |   print("anda lulus dan dapat sertifikat")
4  elif nilai > 60 and nilai <=70:
5      |   print("anda lulus tetapi tidak dapat sertifikat")
6  elif nilai > 50 and nilai <= 60:
7      |   print("anda tidak lulus tetapi anda boleh remidial")
8  elif nilai > 40 and nilai <= 50:
9      |   print("silahkan ulang tahun depan")
10 else:
11     |   print("anda di Do oleh kampus")
12

```

```

input nilai anda:80
anda lulus dan dapat sertifikat
PS C:\Users\ASUS VivoBook\Downloads\UG_2> & "C:/Users/ASUS VivoBook/Documents/kuliah/msy
s64/mingw64/bin/python.exe" "c:/Users/ASUS VivoBook/Downloads/UG_2/percabangan.py"
input nilai anda:45
silahkan ulang tahun depan
PS C:\Users\ASUS VivoBook\Downloads\UG_2> & "C:/Users/ASUS VivoBook/Documents/kuliah/msy
s64/mingw64/bin/python.exe" "c:/Users/ASUS VivoBook/Downloads/UG_2/percabangan.py"
input nilai anda:65
anda lulus tetapi tidak dapat sertifikat
PS C:\Users\ASUS VivoBook\Downloads\UG_2> & "C:/Users/ASUS VivoBook/Documents/kuliah/msy
s64/mingw64/bin/python.exe" "c:/Users/ASUS VivoBook/Downloads/UG_2/percabangan.py"
input nilai anda:12
anda di Do oleh kampus

```

Pada bentuk chained conditional sederhanaanya ialah terdapat beberapa kondisi yang masing masing akan mengeluarkan output yang berbeda beda seperti yang terlihat ketika input nilai 80 maka yang memenuhi conditional ( $\text{nilai} > 75$ ) maka output yang keluar ialah "anda lulus dan dapat sertifikat" kemudian ketika input nilai 45 (memenuhi conditional  $40 \leq 50$ ) maka outputnya ialah "silahkan mengulang tahun depan, dan ini berlaku untuk tiap tiap conditional lain nya termasuk saat tidak ada satupun conditional yang memenuhi maka akan keluar outpt "anda di DO oleh kampus".

### Penanganan kesalahan input

Ketika terdapat input yang harus dilakukan oleh pengguna maka kita harus memperhatikan adanya potensi kesalahan pengguna ketika menginput data agar tidak terjadi eror oleh sebab itu kita harus bisa mengantisipasi kemungkinan tersebut dengan menetapkan penanganan kesalahan, salah satu cara penanganan kesalahan dalam python ialah dengan memakai try dan except contoh :

Contoh program tanpa adanya penanganan kesalahan:

```

percabangan.py > ...
1  nilai=int(input("input nilai anda:"))
2  if nilai > 75:
3      print("anda lulus dan dapat sertifikat")
4  elif nilai > 40 and nilai <= 50:
5      print("silahkan ulang tahun depan")
6  else:
7      print("anda di Do oleh kampus")
8

```

```

input nilai anda:duapuluh
Traceback (most recent call last):
  File "c:\Users\ASUS VivoBook\Downloads\UG_2\percabangan.py", line 1, in <module>
    nilai=int(input("input nilai anda:"))
ValueError: invalid literal for int() with base 10: 'duapuluh'

```

Ketika terjadi salah input oleh user maka akan terdapat eror di program karena diminta memasukan dalam angka bukan nya huruf.

Contoh pemakaian try and except :

```

percabangan.py > ...
1  nilai=(input("input nilai anda:"))
2  try:
3      int(nilai)
4      if nilai > 75:
5          print("anda lulus dan dapat sertifikat")
6      elif nilai > 40 and nilai <= 50:
7          print("silahkan ulang tahun depan")
8      else:
9          print("anda di Do oleh kampus")
10 except :
11     print("terdapat kesalahan input, input nilai dalam range(1-100)")
12

```

```

input nilai anda:dua puluh
terdapat kesalahan input, input nilai dalam range(1-100)

```

Ketika program sudah diberikan penanganan kesalahan maka ketika terdapat input user yang tidak sesuai dengan ketentuan maka output dari program bukanlah eror melainkan berupa pesan “terdapat kesalahan input, input nilai dalam range (1-100) “

## 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

Pada soal pertama kita diminta untuk memodifikasi 3 program dan mengaplikasikan penanganan kesalahan memakai try and except dari ketiga program tersebut sudah ada di modul pembelajaran

Input:

```
soal_1_1.py > ...
1  user = input("Masukkan suhu tubuh: ")
2  try :
3      suhu = int(user)
4      if suhu >= 38:
5          print("Anda demam")
6
7      else:
8          print("Anda tidak demam")
9  except :
10     print("Anda memasukan inputan yang salah")
```

Output:

```
Masukkan suhu tubuh: 39
Anda demam
PS C:\Users\ASUS VivoBook\Downloads\lap_prak_3_vian> & "C:/Users/ASUS VivoBook/Documents
/kuliah/msys64/mingw64/bin/python.exe" "c:/Users/ASUS VivoBook/Downloads/lap_prak_3_vian
/soal_1_1.py"
Masukkan suhu tubuh: tigapuluh
Anda memasukan inputan yang salah
```

Input:



```
suiii = input("Masukkan suatu bilangan: ")
try:
    bilangan = int(suiii)
    if bilangan > 0:
        print("Positif")
    elif bilangan < 0:
        print("Negatif")
    elif bilangan == 0:
        print("Nol")
except :
    print("anda memasukan inputan yang salah, cobalah memasukan angka")
```

Output:

```
Masukkan suatu bilangan: -1
Negatif
PS C:\Users\ASUS VivoBook\Downloads\lap_prak_3_vian> & "C:/Users//
/kuliah/msys64/mingw64/bin/python.exe" "c:/Users/ASUS VivoBook/Dov
/soal_1_2.py"
Masukkan suatu bilangan: tiga
anda memasukan inputan yang salah, cobalah memasukan angka
```

Input:

```
soal_1_3.py > ...
1  # input a, b dan c
2  prtm = input("Masukkan bilangan pertama: ")
3  kdua = input("Masukkan bilangan kedua: ")
4  tga = input("Masukkan bilangan ketiga: ")
5  # secara berurutan tulis kriteria untuk a, b, dan c
6  try :
7      a = int(prtm)
8      b = int(kdua)
9      c = int(tga)
10     if a > b and a > c:
11         print("Terbesar: ", a)
12     elif b > a and b > c:
13         print("Terbesar: ", b)
14     elif c > a and c > b:
15         print("Terbesar: ", c)
16 except :
17     print("terdapat kesalahan input,cobalah input ulang")
```

Output:

```
Masukkan bilangan pertama: tiga
Masukkan bilangan kedua: 23
Masukkan bilangan ketiga: 1
terdapat kesalahan input,cobalah input ulang
```

Setelah diterapkan penanganan kesalahan dalam ketiga program tersebut maka hasil yang dikeluarkan ketika user melakukan input yang salah bukan lagi sebuah error melainkan pesan yang kita buat

Cara kerja program tersebut ialah dengan mengaplikasikan input dengan beragam tipe data dan ketika tipe data nya sesuai dengan ketentuan nya (dalam contoh diminta data int/integer) maka program conditional / percabangan if akan mengecek input data yang user lakukan yang nantinya akan mengeluarkan output yang sesuai dengan conditionalnya namun ketika tipe data yang di input oleh user tidak sesuai dengan yang diinginkan maka outputnya ialah “terdapat kesalahan input”

## SOAL 2

Pada soal kedua kita diminta untuk mengubah bentuk program chained conditional biasa menjadi bentuk ternary yang ada dalam contoh, bentuk ternary sendiri merupakan salah satu contoh penulisan conditional Dimana hanya terdiri dari satu line.

Input:

```
soal_2.py > ...
1  bilangan = int(input("Masukkan suatu bilangan: "))
2  print ("positif") if bilangan > 0 else print( "negatif" ) if bilangan < 0 else print( "nol" )
```

Output:

```
Masukkan suatu bilangan: 12
positif
PS C:\Users\ASUS VivoBook\Downloads
xe" "c:/Users/ASUS VivoBook/Downloa
Masukkan suatu bilangan: 0
nol
PS C:\Users\ASUS VivoBook\Downloads
xe" "c:/Users/ASUS VivoBook/Downloa
Masukkan suatu bilangan: -90
negatif
```

Bentuk ternary dari conditipnal yang dibuat dalam program bertujuan untuk mengecek apakah sebuah bilangan termasuk positif, negative, atau nol oleh karena itu maka penulisannya dapat dibuat menjadi bentuk ternary yang mana hanya terdiri dari 1 line program saja namun tetap bisa menghasilkan output yang sama dengan chained conditional biasanya.

### SOAL 3

Pada soal ketiga kita diminta untuk dapat menentukan jumlah hari dalam suatu bulan di tahun 2020 dan kita diminta untuk tetap mengaplikasikan penanganan kesalahan dalam program tersebut

Input:

```
soal_3.py > ...
1 masukan=input("masukan bulan(1-12):")
2 try:
3     bulan = int(masukan)
4     if bulan == 1 or bulan == 3 or bulan==5 or bulan==8 or bulan==7 or bulan==10 or bulan==12:
5         hari = 31
6         print(f"jadi total hari : {hari}")
7     elif bulan == 4 or bulan==6 or bulan==9 or bulan==11 :
8         hari = 30
9         print(f"jadi total hari : {hari}")
10    elif bulan ==2:
11        hari = 29
12        print(f"jadi total hari : {hari}")
13    else:
14        print ("bulan tersebut tidak valid")
15 except:
16    print("terdapat salah input, cobalah masukan input bulan dengan angka antara 1-12")

masukan bulan(1-12):2
jadi total hari : 29
PS C:\Users\ASUS VivoBook\Downloads\lap_prak_3_vian> & "C:/Users/ASUS VivoBook/xe" "c:/Users/ASUS VivoBook/Downloads/lap_prak_3_vian/soal_3.py"
masukan bulan(1-12):februari
terdapat salah input, cobalah masukan input bulan dengan angka antara 1-12
PS C:\Users\ASUS VivoBook\Downloads\lap_prak_3_vian> & "C:/Users/ASUS VivoBook/xe" "c:/Users/ASUS VivoBook/Downloads/lap_prak_3_vian/soal_3.py"
masukan bulan(1-12):12
jadi total hari : 31
PS C:\Users\ASUS VivoBook\Downloads\lap_prak_3_vian> & "C:/Users/ASUS VivoBook/xe" "c:/Users/ASUS VivoBook/Downloads/lap_prak_3_vian/soal_3.py"
masukan bulan(1-12):13
bulan tersebut tidak valid
```

dalam code ini kita diminta untuk dapat menampilkan jumlah hari pada bulan tertentu sesuai dengan inputan user dan tetap menerapkan penanganan kesalahan input, cara kerja code:

pertama tama user diminta untuk melakukan input untuk menentukan nilai dari variable bulan dan apabila inputan user sesuai dengan ketentuan yaitu interger maka input akan di cek oleh chained conditional yang kita buat Dimana ketika bulan bernilai 1 or 3 or 5 or 7 or 8 or 10 or 12 maka output nya adalah 31 hari dan ketika input buln bernilai 2 maka outputnya ialah 29 dan ketika input bulan bernilai 4 or 6 or 9 or 11 maka output nya akan bernilai 30. Dan ketika value bulan bernilai lebih dari 12 atau kurang dari 1 maka outputnya adalah "bulan tersebut tidak valid".

Ketika input oleh user tidak bertipe int maka output nya akan langsung mengeluarkan "terdapat salah input, cobalah masukan input bulan dengan angka antara 1-12"

## SOAL 4

Input:

```
soal_4.py > ...
1  #3 sisi segitiga
2  input_1= input("masukan sisi pertama segitiga:")
3  input_2= input("masukan sisi kedua segitiga:")
4  input_3= input("masukan sisi ketiga segitiga:")
5  try:
6      s_1= int(input_1)
7      s_2= int(input_2)
8      s_3= int(input_3)
9      if s_1+s_2>=s_3 and s_1+s_3>=s_2 and s_2+s_3>=s_1:
10         print ("merupakan segitiga")
11         if s_1 == s_2 and s_1 == s_3 and s_2 == s_3:
12             print("dengan ketiga sisi yang sama")
13         elif s_1 == s_2 and s_1 != s_3 or s_2 == s_3 and s_3 != s_1 or s_1==s_3 and s_2!=s_1:
14             print("dengan dua sisi yang sama")
15         else:
16             print("tidak ada sisi yang sama")
17     else:
18         print("bukan segitiga")
19 except:
20     print("terdapat salah inputan cobalah masukan input yang lain")
```

Output:

```
masukan sisi pertama segitiga:12
masukan sisi kedua segitiga:13
masukan sisi ketiga segitiga:12
merupakan segitiga
dengan dua sisi yang sama
```

```
masukan sisi pertama segitiga:13
masukan sisi kedua segitiga:13
masukan sisi ketiga segitiga:1
merupakan segitiga
dengan dua sisi yang sama
```

```
masukan sisi pertama segitiga:1
masukan sisi kedua segitiga:1
masukan sisi ketiga segitiga:24
bukan segitiga
```

```
masukan sisi pertama segitiga:1
masukan sisi kedua segitiga:2
masukan sisi ketiga segitiga:3
merupakan segitiga
tidak ada sisi yang sama
```

```
masukan sisi pertama segitiga:dua
masukan sisi kedua segitiga:lima
masukan sisi ketiga segitiga:4
terdapat salah inputan cobalah masukan input yang lain
```

Penjelasan code:

Dalam code ini pertama tama user diminta untuk melakukan 3 input Dimana nanti akan menentukan segitiga tersebut valid atau tidak, dan berapa sisi segitiga tersebut yang sama panjangnya, cara kerja program:

Pertama tama user diminta untuk memasukan input untuk 3 bilangan yang mewakili sisi sisi segitiga

Apabila input sesuai dengan ketentuan(integer) maka data akan di proses menuju compound conditional Dimana pada kondisi pertama ketiga value akan dibandingkan dengan rumus  $a + b >= c$  and  $a + c >= b$  and  $c + b >= a$  code ini dipakai untuk menentukan apakah ketiga sisi yang di input merupakan sebuah segitiga atau tidak, karena sifat segitiga ialah jumlah dua sisi segitiga harus lebih besar dari sisi Panjang segitiganya jadi ketika sifat ini tidak terpenuhi maka akan muncul output “bukan segitiga” ketika syarat pertama terpenuhi maka akan muncul output “merupakan segitiga” yang kemudian masuk ke dalam chained conditional lain yang berfungsi untuk menguji jumlah sisi yang sama dalam segitiga itu pada conditional pertama ialah saat sisi 1 sisi 2 dan sisi 3 panjangnya sama yang dirumuskan dengan  $s_1 == s_2$  and  $s_1 == s_3$  and  $s_2 == s_3$  ketika kondisi ini terpenuhi maka akan menampilkan output”dengan ketiga sisi yang sama”. Ada juga conditional kedua yang berguna untuk mengecek apakah ada 2 sisi yang sama dalam segitiga tersebut yang dirumuskan dengan  $s_1 == s_2$  and  $s_1 != s_3$  or  $s_2 == s_3$  and  $s_3 != s_1$  or  $s_1 == s_3$  and  $s_2 != s_1$  dalam conditional ini maka apabila ada dua sisi yang sama akan muncul output “dengan dua sisi yang sama”, terakhir apabila tidak masuk dalam conditional-conditional diatasnya maka outputnya ialah “tidak ada sisi yang sama”

Untuk kesalahan input maka output yang keluar adalah “terdapat salah inputan, cobalah masukan input yang lain”

Source code : [https://github.com/vianchr/tugas\\_3\\_alpro\\_vian](https://github.com/vianchr/tugas_3_alpro_vian)