

**LAPORAN PRAKTIKUM  
PEMROGRAMAN I  
MODUL 4**



**Loop**

**Oleh:**

**Ashma Salimah**

**NIM. 2410817220015**

**PROGRAM STUDI TEKNOLOGI INFORMASI  
FAKULTAS TEKNIK  
UNIVERSITAS LAMBUNG MANGKURAT  
NOVEMBER 2024**

**LEMBAR PENGESAHAN**  
**LAPORAN PRAKTIKUM PEMROGRAMAN I**  
**MODUL 4**

Laporan Praktikum Pemrograman I Modul 4 : Loop ini disusun sebagai syarat lulus mata kuliah Praktikum Pemrograman I. Laporan Praktikum ini dikerjakan oleh:

Nama Praktikan : Ashma Salimah  
NIM : 2410817220015

Menyetujui,  
Asisten Praktikum

Mengetahui,  
Dosen Penanggung Jawab Praktikum

Muhammad Ryan Rizky Rahmadi  
NIM. 2210817310001

Helda Yunita, S.Kom., M.Kom.  
NIP. 199106192024062001

## DAFTAR ISI

LEMBAR PENGESAHAN .....	2
DAFTAR ISI .....	3
DAFTAR GAMBAR.....	5
DAFTAR TABEL .....	6
SOAL 1 .....	7
A.    Source Code .....	8
•    C.....	8
•    Python .....	8
B.    Output Program.....	8
•    C.....	8
•    Python .....	9
C.    Pembahasan.....	9
•    C.....	9
•    Python .....	10
SOAL 2.....	12
A.    Source Code .....	12
•    C.....	12
•    Python .....	13
B.    Output Program.....	13
•    C.....	13
•    Python .....	14
C.    Pembahasan.....	14
•    C.....	14
•    Python .....	15
SOAL 3.....	17
A.    Source Code .....	17
•    C.....	17
•    Python .....	18
B.    Output Program.....	19

• C.....	19
• Python .....	19
C. Pembahasan.....	20
• C.....	20
• Python .....	21
SOAL 4.....	22
A. Source Code .....	24
• C.....	24
• Python .....	25
B. Output Program.....	26
• C.....	26
• Python .....	26
C. Pembahasan.....	27
• C.....	27
• Python .....	28
SOAL 5.....	29
A. Source Code .....	30
• C.....	30
• Python .....	30
B. Output Program.....	31
• C.....	31
• Python .....	31
C. Pembahasan.....	32
• C.....	32
• Python .....	33

## DAFTAR GAMBAR

Gambar 1 Screenshot 1 Hasil Jawaban Soal 1 .....	8
Gambar 2 Screenshot 2 Hasil Jawaban Soal 1 .....	9
Gambar 3 Screenshot 1 Hasil Jawaban Soal 2 .....	13
Gambar 4 Screenshot 2 Hasil Jawaban Soal 2 .....	14
Gambar 5 Screenshot 1 Hasil Jawaban Soal 3 .....	19
Gambar 6 Screenshot 2 Hasil Jawaban Soal 3 .....	19
Gambar 7 Screenshot 1 Hasil Jawaban Soal 4 .....	26
Gambar 8 Screenshot 2 Hasil Jawaban Soal 4 .....	26
Gambar 9 Screenshot 1 Hasil Jawaban Soal 5 .....	31
Gambar 10 Screenshot 2 Hasil Jawaban Soal 5 .....	31

## DAFTAR TABEL

Table 1 Source Code 1 Soal 1 .....	8
Table 2 Source Code 2 Soal 1 .....	8
Table 3 Source Code 1 Soal 2 .....	12
Table 4 Source 2 Code Soal 2 .....	13
Table 5 Source Code 1 Soal 3 .....	17
Table 6 Source Code 2 Soal 3 .....	18
Table 7 Source Code 1 Soal 4 .....	24
Table 8 Source Code 2 Soal 4 .....	25
Table 9 Source Code 1 Soal 5 .....	30
Table 10 Source Code 2 Soal 5 .....	30

## SOAL 1

1. Pak Dengklek merupakan seorang guru TK Suka Berhitung. Hari ini Pak Dengklek mengajarkan murid-muridnya bilangan kelipatan dengan cara bermain. Setiap murid diminta untuk berjejer dan menyebutkan angka sesuai urutannya kecuali bilangan kelipatan yang harus disebut dengan suatu simbol. Misalnya simbol yang harus disebutkan adalah bintang (\*) pada kelipatan 3 maka urutan yang disebut pada tiap anak menjadi: 1 2 \* 4 5 \* dan seterusnya. Buatlah program untuk membantu Pak Dengklek agar permainan dapat berjalan dengan baik jika jumlah anak didiknya adalah 50 anak.

Info:

Input pertama merupakan bilangan kelipatan yang dirubah menjadi simbol

Input kedua merupakan simbol yang akan menggantikan bilangan tersebut

Output merupakan bilangan 1-50 dengan bilangan kelipatan dirubah menjadi simbol

Input	Output
6 *	1 2 3 4 5 * 7 8 9 10 11 * 13 14 15 16 17 * 19 20 21 22 23 * 25 26 27 28 29 * 31 32 33 34 35 * 37 38 39 40 41 * 43 44 45 46 47 * 49 50
3 #	1 2 # 4 5 # 7 8 # 10 11 # 13 14 # 16 17 # 19 20 # 22 23 # 25 26 # 28 29 # 31 32 # 34 35 # 37 38 # 40 41 # 43 44 # 46 47 # 49 50
11 &	1 2 3 4 5 6 7 8 9 10 & 12 13 14 15 16 17 18 19 20 21 & 23 24 25 26 27 28 29 30 31 32 & 34 35 36 37 38 39 40 41 42 43 & 45 46 47 48 49 50

## A. Source Code

- C

```
1 #include <stdio.h>
2 int main() {
3     int a;
4     char b;
5     scanf("%d %c", &a, &b);
6     int i;
7     for(i=1; i<51; i++) {
8         if(i%a==0) {printf("%c ", b);}
9         else {printf("%d ", i); } }
10    return 0;
11 }
```

Table 1 Source Code 1 Soal 1

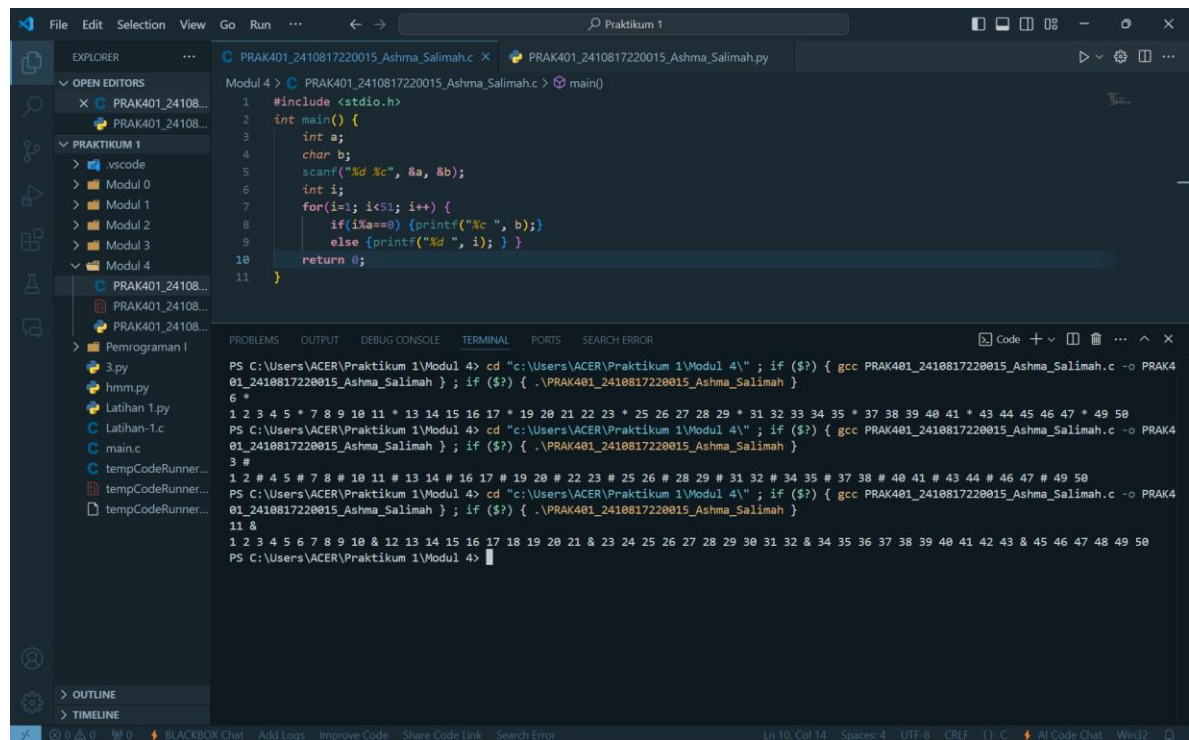
- Python

```
1 a, b = (input().split())
2 a = int(a)
3 for x in range(1, 51) :
4     if x%a==0 : print(b, end=' ')
5     else : print(x, end=' ')
```

Table 2 Source Code 2 Soal 1

## B. Output Program

- C



Gambar 1 Screenshot 1 Hasil Jawaban Soal 1



- Python

The screenshot shows a VS Code editor with a Python file named `PRAK401_2410817220015_Ashma_Salimah.py`. The code in the editor is as follows:

```

1 a, b = (input().split())
2 a = int(a)
3 for x in range(1, 51):
4     if x%a==0 : print(b, end=' ')
5     else : print(x, end=' ')

```

The terminal output shows the execution of the program, which prints a sequence of numbers from 1 to 50, with spaces between numbers that are multiples of 'a' and the number 'b'.

```

PS C:\Users\ACER\Praktikum 1\Modul 4> python -u "c:\Users\ACER\Praktikum 1\Modul 4\PRAK401_2410817220015_Ashma_Salimah.py"
6 *
1 2 3 4 5 * 7 8 9 10 11 * 13 14 15 16 17 * 19 20 21 22 23 * 25 26 27 28 29 * 31 32 33 34 35 * 37 38 39 40 41 * 43 44 45 46 47 * 49 50
PS C:\Users\ACER\Praktikum 1\Modul 4> python -u "c:\Users\ACER\Praktikum 1\Modul 4\PRAK401_2410817220015_Ashma_Salimah.py"
3 #
1 2 # 4 5 # 7 8 # 10 11 # 13 14 # 16 17 # 19 20 # 22 23 # 25 26 # 28 29 # 31 32 # 34 35 # 37 38 # 40 41 # 43 44 # 46 47 # 49 50
PS C:\Users\ACER\Praktikum 1\Modul 4> python -u "c:\Users\ACER\Praktikum 1\Modul 4\PRAK401_2410817220015_Ashma_Salimah.py"
11 &
1 2 3 4 5 6 7 8 9 10 & 12 13 14 15 16 17 18 19 20 21 & 23 24 25 26 27 28 29 30 31 32 & 34 35 36 37 38 39 40 41 42 43 & 45 46 47 48 49 50
PS C:\Users\ACER\Praktikum 1\Modul 4>

```

Gambar 2 Screenshot 2 Hasil Jawaban Soal 1

## C. Pembahasan

- C

Pada baris pertama, terdapat `#include <stdio.h>` yang memiliki fungsi menyertakan library standar, juga sebagai kode input maupun output yang akan dilibatkan dalam program C.

Pada baris kedua, terdapat `int main ()`. Int berfungsi sebagai penunjuk bahwa `main()` mengembalikan nilai integer dan bukan sebuah angka desimal. Fungsi inilah yang menjadi titik awal dan titik akhir eksekusi program.

Pada baris 2 dan 11 terdapat tanda kurung kurawal `{}`. Berfungsi untuk mendefinisikan blok kode, sebagai pendefinisian tubuh fungsi. `{` berfungsi sebagai awal eksekusi program dan `}` berfungsi sebagai akhir tubuh atau akhir eksekusi program.

`int` berfungsi untuk memasukkan fungsi nilai integer atau bilangan bulat.

`char` berfungsi untuk menyimpan fungsi karakter, baik itu huruf, angka, atau simbol.

`scanf("")`; berfungsi untuk membaca string pada input. Tanda `&` sebelum variabel berfungsi untuk memberi alamat memori dari input akan dibaca dan disimpan pada variabel.

Loop `for` berfungsi untuk melakukan pengulangan blok kode. Pada `i=x` , variabel `i` diinisialisasikan dengan variabel `x`, berarti pengulangan dimulai dari variabel `x`. Kemudian pada `i<y` , loop akan terus berjalan selama `i` kurang dari variabel `y`, jika telah mencapai variabel `y`, loop akan berhenti.

Pada `i++`, loop akan meningkat sebanyak satu unit.

Pada `i--`, loop akan berkurang satu unit.

`if()`, `else if()`, dan `else` adalah fungsi kondisional yang berguna untuk mengontrol alur program berdasarkan kondisi tertentu.

`if` untuk memeriksa kondisi. Jika kondisi tersebut benar, maka kode di dalamnya akan dieksekusi.

`else if` digunakan untuk memeriksa kondisi tambahan jika kondisi sebelumnya salah.

`else` berfungsi untuk menangani semua kasus yang tidak terpenuhi oleh kondisi yang ada di `if` dan `else if`.

`printf("")` ; berfungsi untuk menampilkan suatu keluaran atau mencetak output pada layar.

`\n` berfungsi untuk menambahkan baris baru pada output. Jika tidak ada `\n` sebelum tanda petik maka output akan tetap berada pada baris yang sama walaupun dalam inputan berada pada baris baru.

`;` berfungsi untuk menandai akhir dari setiap pernyataan atau statement dan bukan sebagai pemisah antara dua pernyataan.

`return 0;` berfungsi sebagai akhir dari eksekusi fungsi, juga sebagai konvensi yang menunjukkan bahwa program dijalankan tanpa kesalahan.

- **Python**

`input().split()` berfungsi untuk membaca dan menyimpan input. Memisahkan string berdasarkan spasi.

`int` berfungsi untuk memasukkan fungsi nilai integer atau bilangan bulat.

`for x in range(x,y) :` , adalah loop untuk menghasilkan angka tertentu, disini, loop awalnya dimulai dari angka pada variabel `x` dan berakhir di angka pada variabel `y`.

`print("")` berfungsi untuk menampilkan suatu keluaran atau mencetak output pada layar.

`if`, `elif`, dan `else` adalah fungsi kondisional yang berguna untuk mengontrol alur program berdasarkan kondisi tertentu.

`if` untuk memeriksa kondisi. Jika kondisi tersebut benar, maka kode di dalamnya akan dieksekusi.

`elif` digunakan untuk memeriksa kondisi tambahan jika kondisi sebelumnya salah.

`else` berfungsi untuk menangani semua kasus yang tidak terpenuhi oleh kondisi yang ada di `if` dan `elif`.

`end=" "`, berfungsi untuk mengeluarkan output yang dipisahkan dengan spasi.

## SOAL 2

2. Hari ini Pak Dengklek tidak bisa mengajar anak didiknya yang ada di TK Suka Berhitung karena sedang sakit sehingga harus digantikan oleh Bu Dengklek. Pak Dengklek berpesan kepada Bu Dengklek agar mengajarkan materi bilangan genap dan ganjil kepada muridnya. Bu Dengklek mempunyai metode agar materi bilangan ganjil genap dapat disampaikan dengan cara yang menyenangkan. Metodenya adalah dengan cara mengurutkan bilangan ganjil dari 1 sampai batas tertentu dan menyebutkan bilangan genapnya secara terbalik. Buatlah program untuk mempermudah Bu Dengklek menghitung bilangan tersebut!

Info:

Input merupakan batas maksimal dari bilangan yang akan dihitung

Output baris pertama merupakan urutan bilangan ganjil dari 1 sampai batas maksimal

Output baris kedua merupakan urutan bilangan genap dari batas maksimal sampai 2

Input	Output
10	1 3 5 7 9 10 8 6 4 2
25	1 3 5 7 9 11 13 15 17 19 21 23 25 24 22 20 18 16 14 12 10 8 6 4 2
6	1 3 5 6 4 2

### A. Source Code

- C

1	#include <stdio.h>
2	int main() {
3	int a;
4	scanf("%d", &a);
5	int i;
6	for(i=1; i<=a; i++) {
7	if(i%2!=0) {printf("%d ", i);} }
8	printf("\n");
9	for(i=a; i>=1; i--) {
10	if(i%2==0) {printf("%d ", i);}}
11	return 0;
12	}

Table 3 Source Code 1 Soal 2

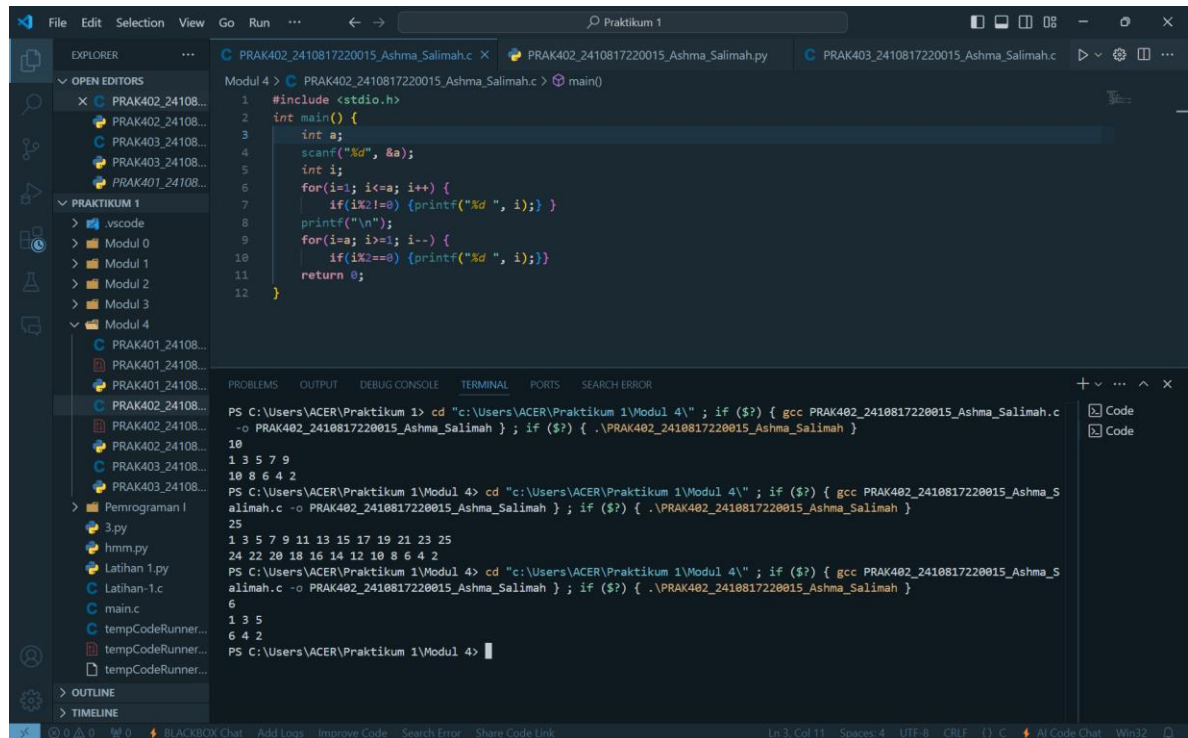
- Python

1	<code>a = int(input(""))</code>
2	<code>for y in range (1, a+1) :</code>
3	<code>    if y%2!=0 : print(y , end=" ")</code>
4	<code>print(" ")</code>
5	<code>for x in range (a,0, -1) :</code>
6	<code>    if x%2==0 : print(x , end=" ")</code>

Table 4 Source 2 Code Soal 2

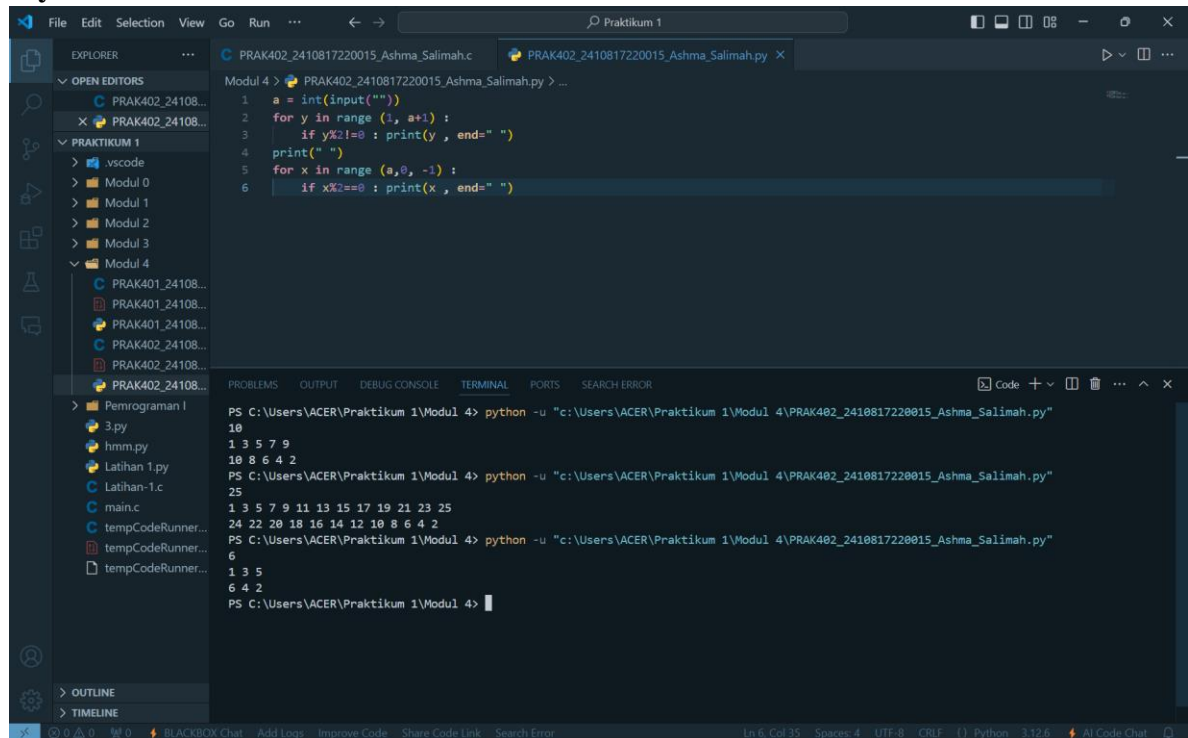
## B. Output Program

- C



Gambar 3 Screenshot 1 Hasil Jawaban Soal 2

- **Python**



Gambar 4 Screenshot 2 Hasil Jawaban Soal 2

### C. Pembahasan

- **C**

Pada baris pertama, terdapat `#include <stdio.h>` yang memiliki fungsi menyertakan library standar, juga sebagai kode input maupun output yang akan dilibatkan dalam program C.

Pada baris kedua, terdapat `int main ()`. Int berfungsi sebagai penunjuk bahwa `main()` membalikkan nilai integer dan bukan sebuah angka desimal. Fungsi inilah yang menjadi titik awal dan titik akhir eksekusi program.

Pada baris 2 dan 12 terdapat tanda kurung kurawal `{}`. Berfungsi untuk mendefinisikan blok kode, sebagai pendefinisian tubuh fungsi. `{` berfungsi sebagai awal eksekusi program dan `}` berfungsi sebagai akhir tubuh atau akhir eksekusi program.

`int` berfungsi untuk memasukkan fungsi nilai integer atau bilangan bulat.

`char` berfungsi untuk menyimpan fungsi karakter, baik itu huruf, angka, atau simbol.

`scanf("");` berfungsi untuk membaca string pada input. Tanda `&` sebelum variabel berfungsi untuk memberi alamat memori dari input akan dibaca dan disimpan pada variabel.

Loop `for` berfungsi untuk melakukan pengulangan blok kode. Pada `i=x`, variabel `i` diinisialisasikan dengan variabel `x`, berarti pengulangan dimulai dari variabel `x`. Kemudian pada `i<y`, loop akan terus berjalan selama `i` kurang dari variabel `y`, jika telah mencapai variabel `y`, loop akan berhenti.

Pada `i++`, loop akan meningkat sebanyak satu unit.

Pada `i--`, loop akan berkurang satu unit.

`if()`, `else if()`, dan `else` adalah fungsi kondisional yang berguna untuk mengontrol alur program berdasarkan kondisi tertentu.

`if` untuk memeriksa kondisi. Jika kondisi tersebut benar, maka kode di dalamnya akan dieksekusi.

`else if` digunakan untuk memeriksa kondisi tambahan jika kondisi sebelumnya salah.

`else` berfungsi untuk menangani semua kasus yang tidak terpenuhi oleh kondisi yang ada di `if` dan `else if`.

`printf("")`; berfungsi untuk menampilkan suatu keluaran atau mencetak output pada layar.

`\n` berfungsi untuk menambahkan baris baru pada output. Jika tidak ada `\n` sebelum tanda petik maka output akan tetap berada pada baris yang sama walaupun dalam inputan berada pada baris baru.

`;` berfungsi untuk menandai akhir dari setiap pernyataan atau statement dan bukan sebagai pemisah antara dua pernyataan.

`return 0;` berfungsi sebagai akhir dari eksekusi fungsi, juga sebagai konvensi yang menunjukkan bahwa program dijalankan tanpa kesalahan.

- **Python**

`Int(input())`, berfungsi untuk membaca dan menyimpan input. Memisahkan string berdasarkan spasi.

`int` berfungsi untuk memasukkan fungsi nilai integer atau bilangan bulat.

`for x in range(x,y) :` adalah loop untuk menghasilkan angka tertentu, disini, loop awalnya dimulai dari angka pada variabel `x` dan berakhir di angka pada variabel `y`.

`print("")` berfungsi untuk menampilkan suatu keluaran atau mencetak output pada layar.

`if`, `elif`, dan `else` adalah fungsi kondisional yang berguna untuk mengontrol alur program berdasarkan kondisi tertentu.

`if` untuk memeriksa kondisi. Jika kondisi tersebut benar, maka kode di dalamnya akan dieksekusi.

`elif` digunakan untuk memeriksa kondisi tambahan jika kondisi sebelumnya salah.

`else` berfungsi untuk menangani semua kasus yang tidak terpenuhi oleh kondisi yang ada di `if` dan `elif`.

`end=" "`, berfungsi untuk mengeluarkan output yang dipisahkan dengan spasi.



### SOAL 3

3. Pak Samson adalah seorang pengusaha sekaligus programmer ternama di Kota Xam. Pak Samson kesulitan membagi waktu antara membuat program dan mengurus usahanya, lalu Pak Samson menyuruh anda membuat program yang diminta oleh clientnya, permintaannya sebagai berikut : Buatlah program angka yang bersilangan, dan input terdiri dari angka pertama dan angka kedua, setiap angka yang bersilangan dibatasi dengan simbol . jika angka pertama lebih besar dari angka kedua maka program akan menampilkan dari terkecil ke terbesar dari batasan angka kedua. jika angka kedua lebih besar dari angka pertama maka program akan menampilkan dari terbesar ke terkecil dari batasan angka kedua.

Untuk lebih jelasnya lihat tabel input output berikut :

Input	Output
3 7	3 7 - 4 6 - 5 5 - 6 4 - 7 3
7 3	7 3 - 6 4 - 5 5 - 4 6 - 3 7
95 100	95 100 - 96 99 - 97 98 - 98 97 - 99 96 - 100 95
23 17	23 17 - 22 18 - 21 19 - 20 20 - 19 21 - 18 22 - 17 23

#### A. Source Code

- C

```
1  #include <stdio.h>
2  int main() {
3      int a, b;
4      scanf("%d %d", &a, &b);
5      int i, j;
6      for (i = a, j = b; i<=b, j>=a; i++, j--) {
7          printf("%d %d ", i, j);
8          if (i<b || j>a) {printf("- ");}}
9      for (i = a, j = b; i>=b, j<=a; i--, j++) {
10         printf("%d %d ", i, j);
11         if (i>b || j<a) {printf("- ");}
12     }
13
14     return 0;
15 }
```

Table 5 Source Code 1 Soal 3

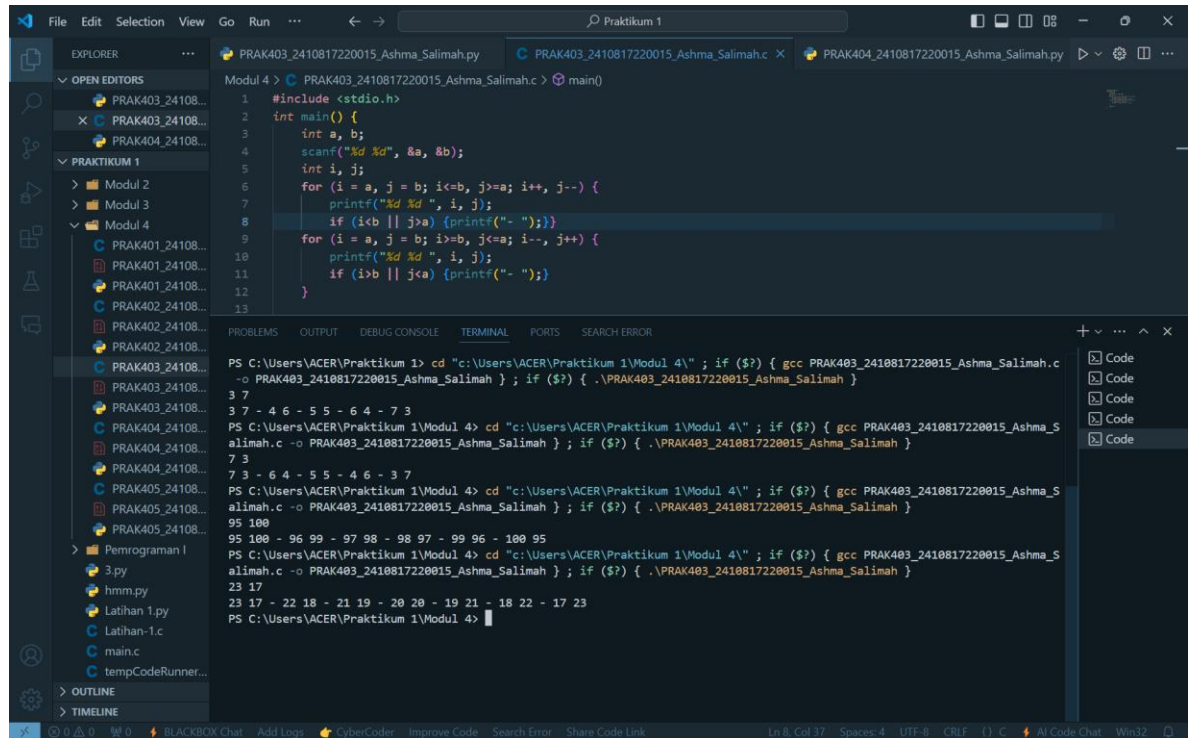
- Python

```
1 a, b = map(int,input().split())
2 x, y = a, b
3 if a <= b :
4     while a<=b :
5         print(f"{a} {b}", end=' ')
6         if a <= b :
7             print(" - ", end=' ')
8             a += 1
9             b -= 1
10    while a <= y :
11        print(f"{a} {b}", end=' ')
12        if a < y:
13            print(" - ", end=' ')
14            a += 1
15            b -= 1
16 else :
17     while a>=b:
18         print(f"{a} {b}", end=' ')
19         if a >= b :
20             print("-", end=' ')
21             a -= 1
22             b += 1
23     while a >= y :
24         print(f"{a} {b}", end=' ')
25         if a > y :
26             print("-", end=' ')
27             a -= 1
28             b += 1
```

Table 6 Source Code 2 Soal 3

## B. Output Program

- C

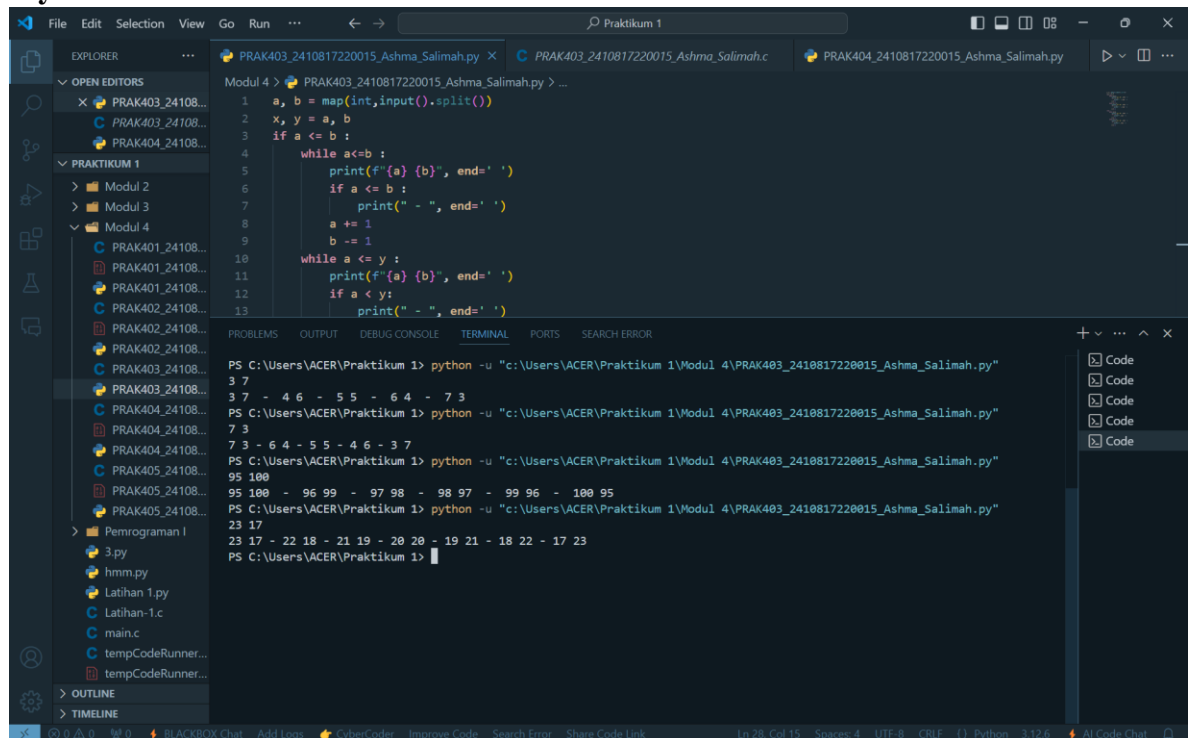


The screenshot shows the Visual Studio Code interface with a C program open in the editor. The program is a simple loop that prints numbers in a specific pattern. The terminal shows the execution of the program, which produces the following output:

```
PS C:\Users\ACER\Praktikum 1> cd "c:\Users\ACER\Praktikum 1\Modul 4"; if ($?) { gcc PRAK403_2410817220015_Ashma_Salimah.c -o PRAK403_2410817220015_Ashma_Salimah }; if ($?) { .\PRAK403_2410817220015_Ashma_Salimah }
3 7
3 7 - 4 6 - 5 5 - 6 4 - 7 3
PS C:\Users\ACER\Praktikum 1\Modul 4> cd "c:\Users\ACER\Praktikum 1\Modul 4"; if ($?) { gcc PRAK403_2410817220015_Ashma_Salimah.c -o PRAK403_2410817220015_Ashma_Salimah }; if ($?) { .\PRAK403_2410817220015_Ashma_Salimah }
7 3
7 3 - 6 4 - 5 5 - 4 6 - 3 7
PS C:\Users\ACER\Praktikum 1\Modul 4> cd "c:\Users\ACER\Praktikum 1\Modul 4"; if ($?) { gcc PRAK403_2410817220015_Ashma_Salimah.c -o PRAK403_2410817220015_Ashma_Salimah }; if ($?) { .\PRAK403_2410817220015_Ashma_Salimah }
95 100
95 100 - 96 99 - 97 98 - 98 97 - 99 96 - 100 95
PS C:\Users\ACER\Praktikum 1\Modul 4> cd "c:\Users\ACER\Praktikum 1\Modul 4"; if ($?) { gcc PRAK403_2410817220015_Ashma_Salimah.c -o PRAK403_2410817220015_Ashma_Salimah }; if ($?) { .\PRAK403_2410817220015_Ashma_Salimah }
23 17
23 17 - 22 18 - 21 19 - 20 20 - 19 21 - 18 22 - 17 23
PS C:\Users\ACER\Praktikum 1\Modul 4>
```

Gambar 5 Screenshot 1 Hasil Jawaban Soal 3

- Python



The screenshot shows the Visual Studio Code interface with a Python program open in the editor. The program is a simple loop that prints numbers in a specific pattern. The terminal shows the execution of the program, which produces the following output:

```
PS C:\Users\ACER\Praktikum 1> python -u "c:\Users\ACER\Praktikum 1\Modul 4\PRAK403_2410817220015_Ashma_Salimah.py"
3 7
3 7 - 4 6 - 5 5 - 6 4 - 7 3
PS C:\Users\ACER\Praktikum 1> python -u "c:\Users\ACER\Praktikum 1\Modul 4\PRAK403_2410817220015_Ashma_Salimah.py"
7 3
7 3 - 6 4 - 5 5 - 4 6 - 3 7
PS C:\Users\ACER\Praktikum 1> python -u "c:\Users\ACER\Praktikum 1\Modul 4\PRAK403_2410817220015_Ashma_Salimah.py"
95 100
95 100 - 96 99 - 97 98 - 98 97 - 99 96 - 100 95
PS C:\Users\ACER\Praktikum 1> python -u "c:\Users\ACER\Praktikum 1\Modul 4\PRAK403_2410817220015_Ashma_Salimah.py"
23 17
23 17 - 22 18 - 21 19 - 20 20 - 19 21 - 18 22 - 17 23
PS C:\Users\ACER\Praktikum 1>
```

Gambar 6 Screenshot 2 Hasil Jawaban Soal 3

## C. Pembahasan

- C

Pada baris pertama, terdapat `#include <stdio.h>` yang memiliki fungsi menyertakan library standar, juga sebagai kode input maupun output yang akan dilibatkan dalam program C.

Pada baris kedua, terdapat `int main ()`. Int berfungsi sebagai penunjuk bahwa `main()` membalikkan nilai integer dan bukan sebuah angka desimal. Fungsi inilah yang menjadi titik awal dan titik akhir eksekusi program.

Pada baris 2 dan 15 terdapat tanda kurung kurawal (`{}`). Berfungsi untuk mendefinisikan blok kode, sebagai pendefinisian tubuh fungsi. `{` berfungsi sebagai awal eksekusi program dan `}` berfungsi sebagai akhir tubuh atau akhir eksekusi program.

`int` berfungsi untuk memasukkan fungsi nilai integer atau bilangan bulat.

`char` berfungsi untuk menyimpan fungsi karakter, baik itu huruf, angka, atau simbol.

`scanf("")`; berfungsi untuk membaca string pada input. Tanda `&` sebelum variabel berfungsi untuk memberi alamat memori dari input akan dibaca dan disimpan pada variabel.

Loop `for` berfungsi untuk melakukan pengulangan blok kode. Pada `i=x`, variabel `i` diinisialisasikan dengan variabel `x`, berarti pengulangan dimulai dari variabel `x`.

Kemudian pada `i<y`, loop akan terus berjalan selama `i` kurang dari variabel `y`, jika telah mencapai variabel `y`, loop akan berhenti.

Pada `i++`, loop akan meningkat sebanyak satu unit.

Pada `i--`, loop akan berkurang satu unit.

`if()`, `else if()`, dan `else` adalah fungsi kondisional yang berguna untuk mengontrol alur program berdasarkan kondisi tertentu.

`if` untuk memeriksa kondisi. Jika kondisi tersebut benar, maka kode di dalamnya akan dieksekusi.

`else if` digunakan untuk memeriksa kondisi tambahan jika kondisi sebelumnya salah.

`else` berfungsi untuk menangani semua kasus yang tidak terpenuhi oleh kondisi yang ada di `if` dan `else if`.

`printf("")`; berfungsi untuk menampilkan suatu keluaran atau mencetak output pada layar.

`\n` berfungsi untuk menambahkan baris baru pada output. Jika tidak ada `\n` sebelum tanda petik maka output akan tetap berada pada baris yang sama walaupun dalam inputan berada pada baris baru.

`;` berfungsi untuk menandai akhir dari setiap pernyataan atau statement dan bukan sebagai pemisah antara dua pernyataan.

`return 0;` berfungsi sebagai akhir dari eksekusi fungsi, juga sebagai konvensi yang menunjukkan bahwa program dijalankan tanpa kesalahan.

- **Python**

`Int(input())`, berfungsi untuk membaca dan menyimpan input. Memisahkan string berdasarkan spasi.

`int` berfungsi untuk memasukkan fungsi nilai integer atau bilangan bulat.

**while** adalah loop untuk menghasilkan angka tertentu, disini, loop awalnya dimulai dari angka pada variabel x dan berakhir di angka pada variabel y.

`print("")` berfungsi untuk menampilkan suatu keluaran atau mencetak output pada layar.

`if, elif, dan else` adalah fungsi kondisional yang berguna untuk mengontrol alur program berdasarkan kondisi tertentu.

`if` untuk memeriksa kondisi. Jika kondisi tersebut benar, maka kode di dalamnya akan dieksekusi.

`elif` digunakan untuk memeriksa kondisi tambahan jika kondisi sebelumnya salah.

`else` berfungsi untuk menangani semua kasus yang tidak terpenuhi oleh kondisi yang ada di `if` dan `elif`.

**end=" "**, berfungsi untuk mengeluarkan output yang dipisahkan dengan spasi.

## SOAL 4

4. Pa Jack mencari seorang programmer yang bisa membuatkan program kalkulator sederhana untuk dirinya, karena dia kesulitan menghitung hasil penjualannya. Buatlah sebuah kalkulator sederhana yang di minta sesuai dengan yang Pa Jack inginkan dengan output sebagai berikut :

Pilih program

1. Penjumlahan
2. Pengurangan
3. Perkalian
4. Pembagian
5. Exit

Masukkan Pilihan :

Masukkan nilai pertama :

Masukkan nilai kedua :

Hasil **Pilihan** antara **NilaiPertama** dengan **NilaiKedua** adalah **Hasil**

- Selama program belum memasukkan pilihan angka 5, maka program akan terus berjalan.
- Jika memasukkan angka 5 maka selanjutnya program selesai dan tampilkan : Terimakasih, telah menggunakan kalkulator NAMAANDA
- Jika memasukkan angka selain dari angka yang tertera mulai ulang programnya dan tampilkan : Input anda salah, silahkan coba lagi.

**Note : Lebih jelasnya untuk input output lihat dari link:**

<https://bit.ly/PenjelasanSoalNo4>

✓ yang bertanda merah diganti dengan yang sesuai dengan inputan, misal: **Masukkan Pilihan : 2 , Nilai Pertama : 4 , dan Nilai Kedua : 2 . maka outputnya sebagai berikut = Hasil Pengurangan antara 4.00 dengan 2.00 adalah 2.00**

✓ **Ketelitian 2 angka dibelakang koma.**

Input	Output
Pilih program	Pilih program
1. Penjumlahan	1. Penjumlahan
2. Pengurangan	2. Pengurangan
3. Perkalian	3. Perkalian
4. Pembagian	4. Pembagian
5. Exit	5. Exit

Masukkan Pilihan : 3 Masukkan nilai pertama : 12 Masukkan nilai kedua : 5	Masukkan Pilihan : 3 Masukkan nilai pertama :12 Masukkan nilai kedua :5 Hasil perkalian antara 12.00 dengan 5.00 adalah 60.00
Pilih program 1. Penjumlahan 2. Pengurangan 3. Perkalian 4. Pembagian 5. Exit  Masukkan Pilihan : 13	Pilih program 1. Penjumlahan 2. Pengurangan 3. Perkalian 4. Pembagian 5. Exit  Masukkan Pilihan : 13 Input anda salah, silahkan coba lagi
Pilih program 1. Penjumlahan 2. Pengurangan 3. Perkalian 4. Pembagian 5. Exit  Masukkan Pilihan : 5	Pilih program 1. Penjumlahan 2. Pengurangan 3. Perkalian 4. Pembagian 5. Exit  Masukkan Pilihan : 5 Terimakasih, telah menggunakan kalkulator <b>NAMAANDA</b>

## A. Source Code

### • C

1	#include <stdio.h>
2	int main() {
3	int x;
4	float y, z;
5	do {
6	printf("Pilih program\n 1. Penjumlahan\n 2. Pengurangan\n 3. Perkalian\n 4. Pembagian\n 5. Exit\n");
7	printf("Masukkan Pilihan : ");
8	scanf("%d", &x);
9	if (x>=1 && x<=4) { printf("Masukkan nilai pertama : ");
10	scanf("%f", &y);
11	printf("Masukkan nilai kedua : ");
12	scanf("%f", &z); }
13	if (x == 1) { float hasil_jumlah = y + z;
14	printf("Hasil penjumlahan antara %.2f dengan %.2f adalah %.2f \n\n", y, z, hasil_jumlah); }
15	else if (x == 2) { float hasil_kurang = y - z;
16	printf("Hasil pengurangan antara %.2f dengan %.2f adalah %.2f \n\n", y, z, hasil_kurang); }
17	else if (x == 3) { float hasil_kali = y * z;
18	printf("Hasil perkalian antara %.2f dengan %.2f adalah %.2f \n\n", y, z, hasil_kali); }
19	else if (x == 4) { float hasil_bagi = y / z;
20	if (z != 0) { printf("Hasil perkalian antara %.2f dengan %.2f adalah %.2f \n\n", y, z, hasil_bagi); }
21	else {printf("Error! = Pembagian tidak boleh 0!");} }
22	else if (x == 5) { printf("Terimakasih, telah menggunakan kalkulator ASHMA SALIMAH \n\n"); }
23	else { printf("Input anda salah, silahkan coba lagi \n\n"); }
24	} while (x!=5);
25	return 0;
26	}

Table 7 Source Code 1 Soal 4



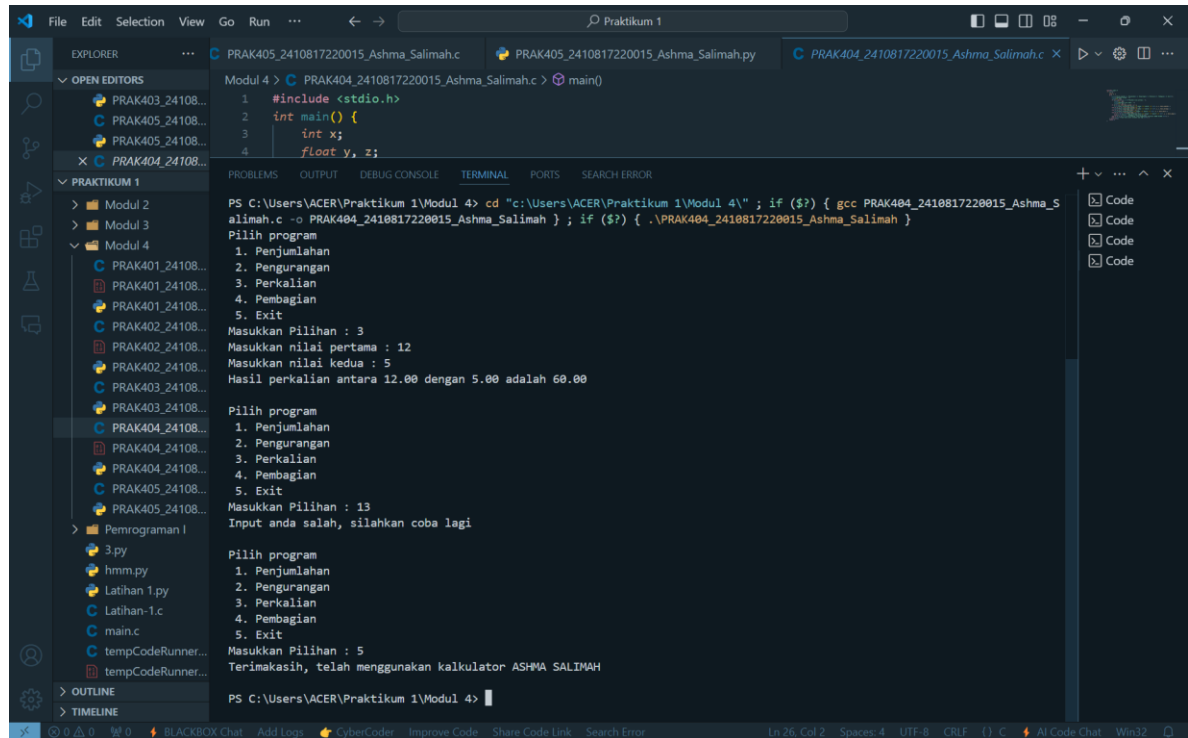
- Python

1	<b>while</b> True :
2	print("Pilih program \n1. Penjumlahan \n2. Pengurangan \n3. Perkalian \n4. Pembagian \n5. Exit")
3	x = int(input("Masukkan Pilihan : "))
4	
5	<b>if</b> 1 <= x <= 4 :
6	y = float(input("Masukkan nilai pertama : "))
7	z = float(input("Masukkan nilai kedua : "))
8	
9	<b>if</b> x == 1 :
10	hasil_jumlah = y + z
11	print(f"Hasil penjumlahan antara {y:.2f} dengan {z:.2f} adalah {hasil_jumlah:.2f}\n")
12	<b>elif</b> x == 2 :
13	hasil_kurang = y - z
14	print(f"Hasil pengurangan antara {y:.2f} dengan {z:.2f} adalah {hasil_kurang:.2f}\n")
15	<b>elif</b> x == 3 :
16	hasil_kali = y * z
17	print(f"Hasil perkalian antara {y:.2f} dengan {z:.2f} adalah {hasil_kali:.2f}\n")
18	<b>elif</b> x == 4 :
19	<b>if</b> z !=0 <b>and</b> y!=0 :
20	hasil_bagi = y / z
21	print(f"Hasil pembagian antara {y:.2f} dengan {z:.2f} adalah {hasil_bagi:.2f}\n")
22	<b>else</b> :
23	print("Error! = Pembagian tidak boleh 0!\n")
24	
25	<b>elif</b> x == 5 :
26	print("Terimakasih, telah menggunakan kalkulator ASHMA SALIMAH \n")
27	<b>break</b>
28	
29	<b>else</b> :
30	print("Input anda salah, silahkan coba lagi\n")

Table 8 Source Code 2 Soal 4

## B. Output Program

- C



```
Modul 4 > C PRAK404_2410817220015_Ashma_Salimah.c > main()
1 #include <stdio.h>
2 int main() {
3     int x;
4     float y, z;

PS C:\Users\ACER\Praktikum 1\Modul 4> cd "c:\Users\ACER\Praktikum 1\Modul 4\"; if ($?) { gcc PRAK404_2410817220015_Ashma_S
alimah.c -o PRAK404_2410817220015_Ashma_Salimah }; if ($?) { .\PRAK404_2410817220015_Ashma_Salimah }
Pilih program
1. Penjumlahan
2. Pengurangan
3. Perkalian
4. Pembagian
5. Exit
Masukkan Pilihan : 3
Masukkan nilai pertama : 12
Masukkan nilai kedua : 5
Hasil perkalian antara 12.00 dengan 5.00 adalah 60.00

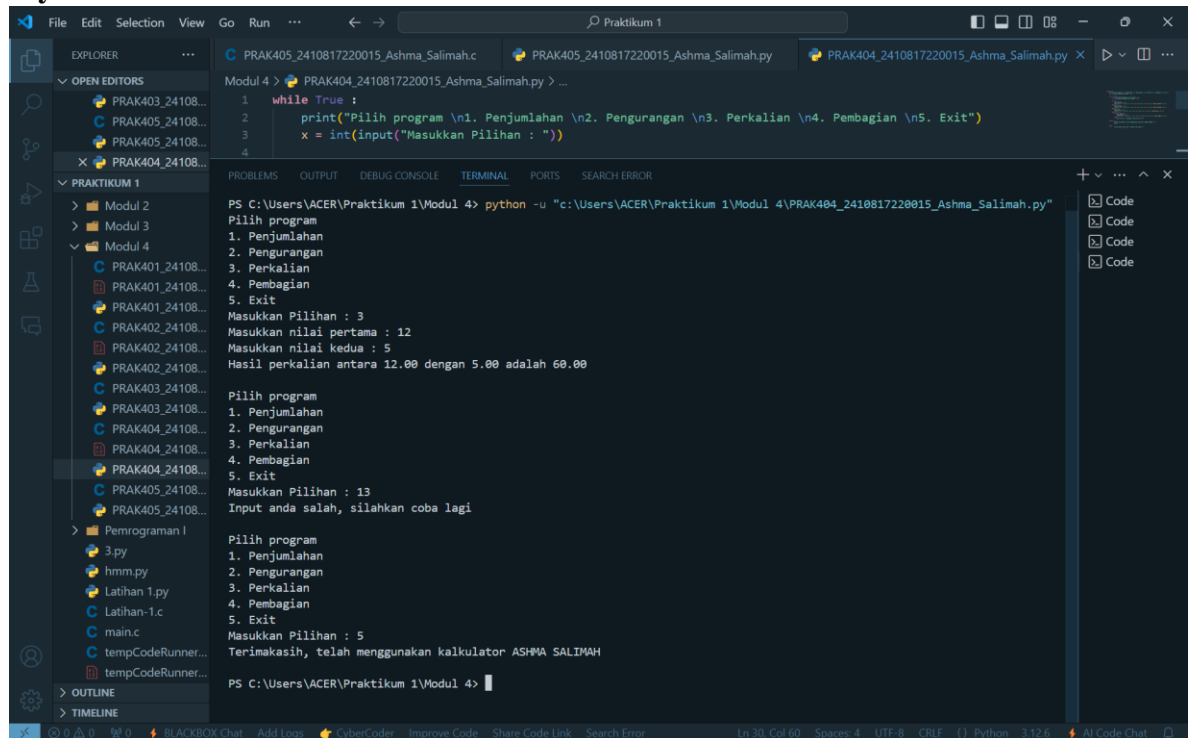
Pilih program
1. Penjumlahan
2. Pengurangan
3. Perkalian
4. Pembagian
5. Exit
Masukkan Pilihan : 13
Input anda salah, silahkan coba lagi

Pilih program
1. Penjumlahan
2. Pengurangan
3. Perkalian
4. Pembagian
5. Exit
Masukkan Pilihan : 5
Terimakasih, telah menggunakan kalkulator ASHMA SALIMAH

PS C:\Users\ACER\Praktikum 1\Modul 4>
```

Gambar 7 Screenshot 1 Hasil Jawaban Soal 4

- Python



```
Modul 4 > PRAK404_2410817220015_Ashma_Salimah.py > ...
1 while True :
2     print("Pilih program \n1. Penjumlahan \n2. Pengurangan \n3. Perkalian \n4. Pembagian \n5. Exit")
3     x = int(input("Masukkan Pilihan : "))
4

PS C:\Users\ACER\Praktikum 1\Modul 4> python -u "c:\Users\ACER\Praktikum 1\Modul 4\PRAK404_2410817220015_Ashma_Salimah.py"
Pilih program
1. Penjumlahan
2. Pengurangan
3. Perkalian
4. Pembagian
5. Exit
Masukkan Pilihan : 3
Masukkan nilai pertama : 12
Masukkan nilai kedua : 5
Hasil perkalian antara 12.00 dengan 5.00 adalah 60.00

Pilih program
1. Penjumlahan
2. Pengurangan
3. Perkalian
4. Pembagian
5. Exit
Masukkan Pilihan : 13
Input anda salah, silahkan coba lagi

Pilih program
1. Penjumlahan
2. Pengurangan
3. Perkalian
4. Pembagian
5. Exit
Masukkan Pilihan : 5
Terimakasih, telah menggunakan kalkulator ASHMA SALIMAH

PS C:\Users\ACER\Praktikum 1\Modul 4>
```

Gambar 8 Screenshot 2 Hasil Jawaban Soal 4

## C. Pembahasan

- C

Pada baris pertama, terdapat `#include <stdio.h>` yang memiliki fungsi menyertakan library standar, juga sebagai kode input maupun output yang akan dilibatkan dalam program C.

Pada baris kedua, terdapat `int main ()`. Int berfungsi sebagai penunjuk bahwa `main()` membalikkan nilai integer dan bukan sebuah angka desimal. Fungsi inilah yang menjadi titik awal dan titik akhir eksekusi program.

Pada baris 2 dan 15 terdapat tanda kurung kurawal (`{}`). Berfungsi untuk mendefinisikan blok kode, sebagai pendefinisian tubuh fungsi. `{` berfungsi sebagai awal eksekusi program dan `}` berfungsi sebagai akhir tubuh atau akhir eksekusi program.

`int` berfungsi untuk memasukkan fungsi nilai integer atau bilangan bulat.

`char` berfungsi untuk menyimpan fungsi karakter, baik itu huruf, angka, atau simbol.

`scanf("")`; berfungsi untuk membaca string pada input. Tanda `&` sebelum variabel berfungsi untuk memberi alamat memori dari input akan dibaca dan disimpan pada variabel.

Loop `for` berfungsi untuk melakukan pengulangan blok kode. Pada `i=x`, variabel `i` diinisialisasikan dengan variabel `x`, berarti pengulangan dimulai dari variabel `x`.

Kemudian pada `i<y`, loop akan terus berjalan selama `i` kurang dari variabel `y`, jika telah mencapai variabel `y`, loop akan berhenti.

Pada `i++`, loop akan meningkat sebanyak satu unit.

Pada `i--`, loop akan berkurang satu unit.

`if()`, `else if()`, dan `else` adalah fungsi kondisional yang berguna untuk mengontrol alur program berdasarkan kondisi tertentu.

`if` untuk memeriksa kondisi. Jika kondisi tersebut benar, maka kode di dalamnya akan dieksekusi.

`else if` digunakan untuk memeriksa kondisi tambahan jika kondisi sebelumnya salah.

`else` berfungsi untuk menangani semua kasus yang tidak terpenuhi oleh kondisi yang ada di `if` dan `else if`.

`printf("")`; berfungsi untuk menampilkan suatu keluaran atau mencetak output pada layar.

`\n` berfungsi untuk menambahkan baris baru pada output. Jika tidak ada `\n` sebelum tanda petik maka output akan tetap berada pada baris yang sama walaupun dalam inputan berada pada baris baru.

`;` berfungsi untuk menandai akhir dari setiap pernyataan atau statement dan bukan sebagai pemisah antara dua pernyataan.

`return 0;` berfungsi sebagai akhir dari eksekusi fungsi, juga sebagai konvensi yang menunjukkan bahwa program dijalankan tanpa kesalahan.

- **Python**

`int(input(""))` berfungsi untuk membaca dan menyimpan input yang akan disimpan sebagai fungsi integer.

`print("")` berfungsi untuk menampilkan suatu keluaran atau mencetak output pada layar.

**while** `True` adalah loop yang akan berjalan terus menerus, untuk menghrntikan loop tersebut bisa menggunakan perintah **break**.

`if`, `elif`, dan `else` adalah fungsi kondisional yang berguna untuk mengontrol alur program berdasarkan kondisi tertentu.

`if` untuk memeriksa kondisi. Jika kondisi tersebut benar, maka kode di dalamnya akan dieksekusi.

`elif` digunakan untuk memeriksa kondisi tambahan jika kondisi sebelumnya salah.

`else` berfungsi untuk menangani semua kasus yang tidak terpenuhi oleh kondisi yang ada di `if` dan `elif`.

## SOAL 5

5. Buatlah sebuah program untuk menghitung jumlah kelipatan pada tiap baris seperti contoh kasus.

Info:

Input baris pertama, banyaknya n.

Input baris kedua, kelipatan.

Output adalah hasil perhitungan dari masing-masing kelipatan pada tiap baris dan output baris terakhir merupakan jumlah pada masing-masing baris.

Input	Output
3 2	$(1 * 2) = 2$ $(2 * 2) + (1 * 2) = 6$ $(3 * 2) + (2 * 2) + (1 * 2) = 12$ 20
5 3	$(1 * 3) = 3$ $(2 * 3) + (1 * 3) = 9$ $(3 * 3) + (2 * 3) + (1 * 3) = 18$ $(4 * 3) + (3 * 3) + (2 * 3) + (1 * 3) = 30$ $(5 * 3) + (4 * 3) + (3 * 3) + (2 * 3) + (1 * 3) = 45$ 105
2 3	$(1 * 3) = 3$ $(2 * 3) + (1 * 3) = 9$ 12

## A. Source Code

### • C

```
1  #include <stdio.h>
2  int main() {
3      int a, k, h = 0;
4      scanf("%d %d", &a, &k);
5      for (int i = 1; i <= a; i++) {
6          int b = 0;
7          for (int j = i; j >= 1; j--) {
8              b += j * k;
9              printf("(%d * %d)", j, k);
10             if (j > 1) { printf(" + "); } }
11             printf(" = %d \n", b);
12             h += b; }
13     printf("%d \n", h);
14     return 0;
15 }
```

Table 9 Source Code 1 Soal 5

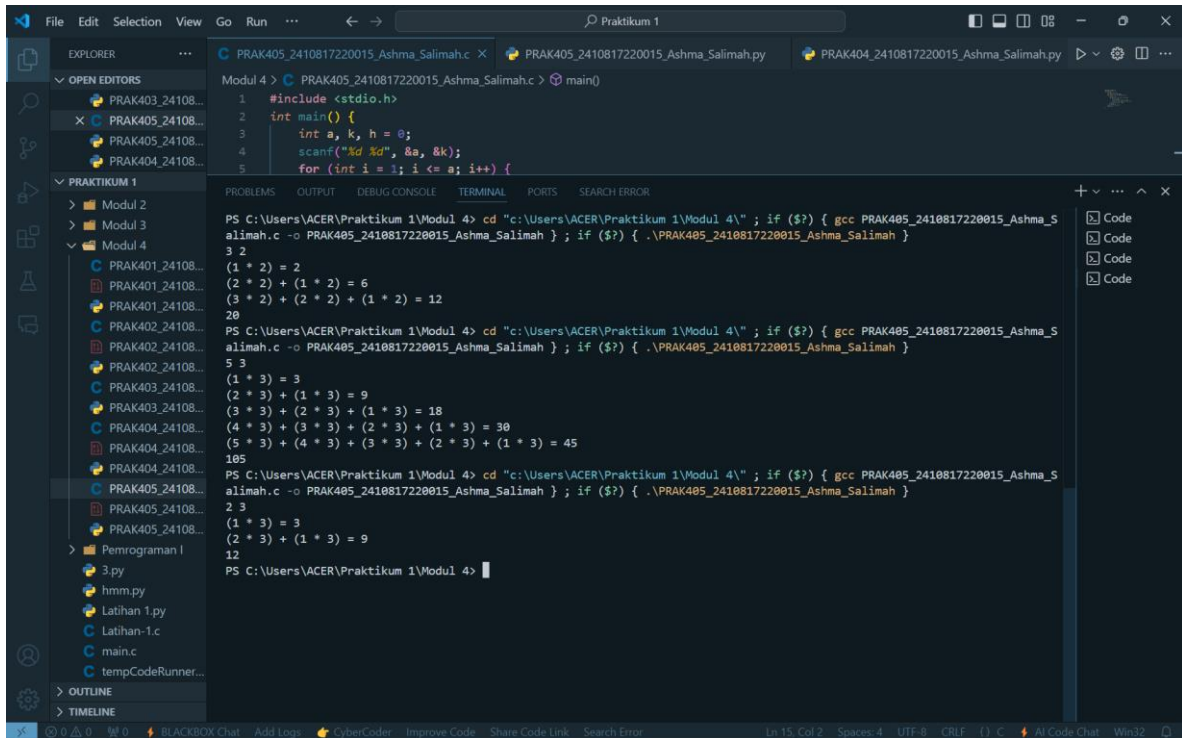
### • Python

```
1  a, k = input().split()
2  a = int(a)
3  k = int(k)
4  h = 0
5  for x in range (1, a + 1) :
6      b = 0
7      for y in range (1, x + 1) :
8          b += y * k
9          print(f"({y} * {k})", end=' ')
10         if y < x :
11             print(" + ", end=' ')
12         h += b
13         print(f" = {b}")
14     print(f"{h}")
```

Table 10 Source Code 2 Soal 5

## B. Output Program

- C



The screenshot shows the Visual Studio Code interface with a C program being executed. The Explorer pane on the left shows the project structure with folders for 'Modul 2', 'Modul 3', and 'Modul 4'. The 'Modul 4' folder is expanded, showing files like 'PRAK401\_24108...', 'PRAK402\_24108...', 'PRAK403\_24108...', 'PRAK404\_24108...', and 'PRAK405\_24108...'. The main editor displays the C code for 'PRAK405\_2410817220015\_Ashma\_Salimah.c', which includes a loop to calculate the sum of squares of numbers from 1 to 'a'. The Output pane at the bottom shows the execution results for 'Modul 4' with the command 'gcc PRAK405\_2410817220015\_Ashma\_Salimah.c -o PRAK405\_2410817220015\_Ashma\_Salimah' and the output of the program, which is the sum of squares for 'a' values 2, 3, 4, and 5.

```
1 #include <stdio.h>
2 int main() {
3     int a, k, h = 0;
4     scanf("%d %d", &a, &k);
5     for (int i = 1; i <= a; i++) {
```

PS C:\Users\ACER\Praktikum 1\Modul 4> cd "c:\Users\ACER\Praktikum 1\Modul 4\" ; if (\$?) { gcc PRAK405\_2410817220015\_Ashma\_Salimah.c -o PRAK405\_2410817220015\_Ashma\_Salimah } ; if (\$?) { .\PRAK405\_2410817220015\_Ashma\_Salimah }

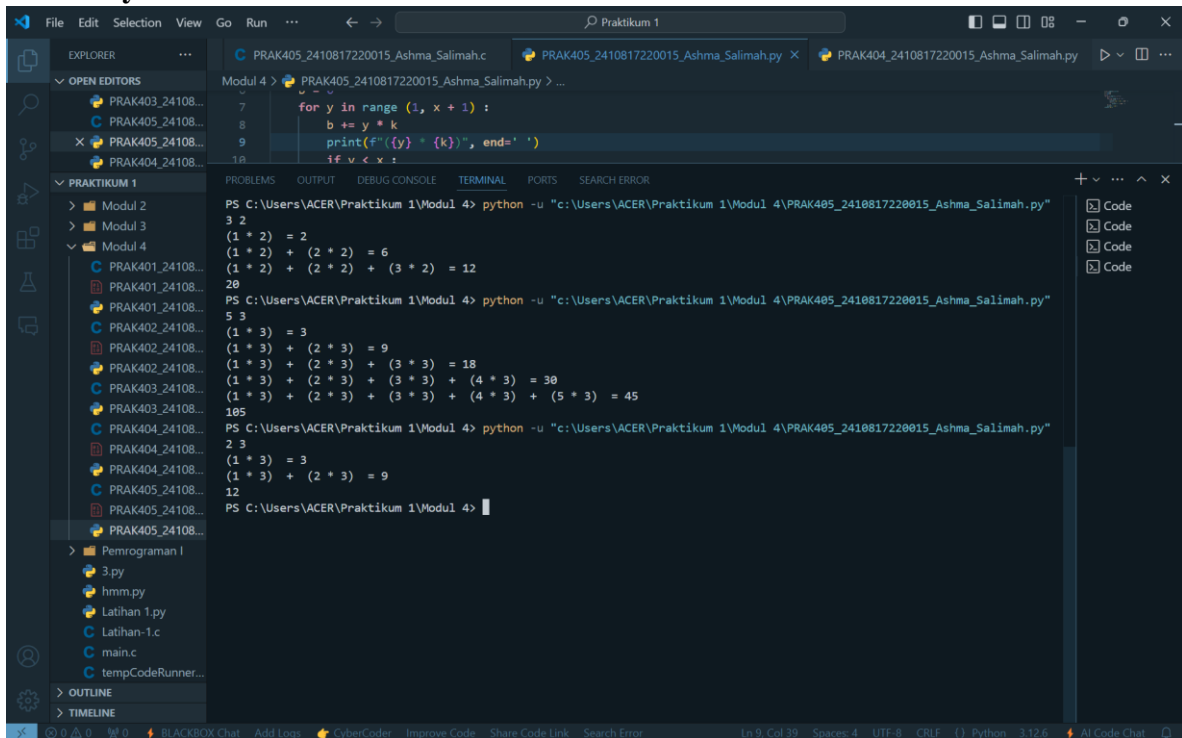
3 2  
(1 \* 2) = 2  
(2 \* 2) + (1 \* 2) = 6  
(3 \* 2) + (2 \* 2) + (1 \* 2) = 12  
20  
PS C:\Users\ACER\Praktikum 1\Modul 4> cd "c:\Users\ACER\Praktikum 1\Modul 4\" ; if (\$?) { gcc PRAK405\_2410817220015\_Ashma\_Salimah.c -o PRAK405\_2410817220015\_Ashma\_Salimah } ; if (\$?) { .\PRAK405\_2410817220015\_Ashma\_Salimah }

5 3  
(1 \* 3) = 3  
(2 \* 3) + (1 \* 3) = 9  
(3 \* 3) + (2 \* 3) + (1 \* 3) = 18  
(4 \* 3) + (3 \* 3) + (2 \* 3) + (1 \* 3) = 30  
(5 \* 3) + (4 \* 3) + (3 \* 3) + (2 \* 3) + (1 \* 3) = 45  
105  
PS C:\Users\ACER\Praktikum 1\Modul 4> cd "c:\Users\ACER\Praktikum 1\Modul 4\" ; if (\$?) { gcc PRAK405\_2410817220015\_Ashma\_Salimah.c -o PRAK405\_2410817220015\_Ashma\_Salimah } ; if (\$?) { .\PRAK405\_2410817220015\_Ashma\_Salimah }

2 3  
(1 \* 3) = 3  
(2 \* 3) + (1 \* 3) = 9  
12  
PS C:\Users\ACER\Praktikum 1\Modul 4> |

Gambar 9 Screenshot 1 Hasil Jawaban Soal 5

- Python



The screenshot shows the Visual Studio Code interface with a Python program being executed. The Explorer pane on the left shows the project structure with folders for 'Modul 2', 'Modul 3', and 'Modul 4'. The 'Modul 4' folder is expanded, showing files like 'PRAK401\_24108...', 'PRAK402\_24108...', 'PRAK403\_24108...', 'PRAK404\_24108...', and 'PRAK405\_24108...'. The main editor displays the Python code for 'PRAK405\_2410817220015\_Ashma\_Salimah.py', which includes a loop to calculate the sum of squares of numbers from 1 to 'x'. The Output pane at the bottom shows the execution results for 'Modul 4' with the command 'python -u "c:\Users\ACER\Praktikum 1\Modul 4\PRAK405\_2410817220015\_Ashma\_Salimah.py"' and the output of the program, which is the sum of squares for 'x' values 2, 3, 4, and 5.

```
7     for y in range(1, x + 1):
8         b += y * k
9         print(f'({y}) * {k})', end=' ')
10    if v < x:
```

PS C:\Users\ACER\Praktikum 1\Modul 4> python -u "c:\Users\ACER\Praktikum 1\Modul 4\PRAK405\_2410817220015\_Ashma\_Salimah.py"

3 2  
(1 \* 2) = 2  
(1 \* 2) + (2 \* 2) = 6  
(1 \* 2) + (2 \* 2) + (3 \* 2) = 12  
20  
PS C:\Users\ACER\Praktikum 1\Modul 4> python -u "c:\Users\ACER\Praktikum 1\Modul 4\PRAK405\_2410817220015\_Ashma\_Salimah.py"

5 3  
(1 \* 3) = 3  
(1 \* 3) + (2 \* 3) = 9  
(1 \* 3) + (2 \* 3) + (3 \* 3) = 18  
(1 \* 3) + (2 \* 3) + (3 \* 3) + (4 \* 3) = 30  
(1 \* 3) + (2 \* 3) + (3 \* 3) + (4 \* 3) + (5 \* 3) = 45  
105  
PS C:\Users\ACER\Praktikum 1\Modul 4> python -u "c:\Users\ACER\Praktikum 1\Modul 4\PRAK405\_2410817220015\_Ashma\_Salimah.py"

2 3  
(1 \* 3) = 3  
(1 \* 3) + (2 \* 3) = 9  
12  
PS C:\Users\ACER\Praktikum 1\Modul 4> |

Gambar 10 Screenshot 2 Hasil Jawaban Soal 5

## C. Pembahasan

- C

Pada baris pertama, terdapat `#include <stdio.h>` yang memiliki fungsi menyertakan library standar, juga sebagai kode input maupun output yang akan dilibatkan dalam program C.

Pada baris kedua, terdapat `int main ()`. Int berfungsi sebagai penunjuk bahwa `main()` membalikkan nilai integer dan bukan sebuah angka desimal. Fungsi inilah yang menjadi titik awal dan titik akhir eksekusi program.

Pada baris 2 dan 15 terdapat tanda kurung kurawal (`{}`). Berfungsi untuk mendefinisikan blok kode, sebagai pendefinisian tubuh fungsi. `{` berfungsi sebagai awal eksekusi program dan `}` berfungsi sebagai akhir tubuh atau akhir eksekusi program.

`int` berfungsi untuk memasukkan fungsi nilai integer atau bilangan bulat.

`char` berfungsi untuk menyimpan fungsi karakter, baik itu huruf, angka, atau simbol.

`scanf("")`; berfungsi untuk membaca string pada input. Tanda `&` sebelum variabel berfungsi untuk memberi alamat memori dari input akan dibaca dan disimpan pada variabel.

Loop `for` berfungsi untuk melakukan pengulangan blok kode. Pada `i=x`, variabel `i` diinisialisasikan dengan variabel `x`, berarti pengulangan dimulai dari variabel `x`.

Kemudian pada `i<y`, loop akan terus berjalan selama `i` kurang dari variabel `y`, jika telah mencapai variabel `y`, loop akan berhenti.

Pada `i++`, loop akan meningkat sebanyak satu unit.

Pada `i--`, loop akan berkurang satu unit.

`if()`, `else if()`, dan `else` adalah fungsi kondisional yang berguna untuk mengontrol alur program berdasarkan kondisi tertentu.

`if` untuk memeriksa kondisi. Jika kondisi tersebut benar, maka kode di dalamnya akan dieksekusi.

`else if` digunakan untuk memeriksa kondisi tambahan jika kondisi sebelumnya salah.

`else` berfungsi untuk menangani semua kasus yang tidak terpenuhi oleh kondisi yang ada di `if` dan `else if`.

`printf("")`; berfungsi untuk menampilkan suatu keluaran atau mencetak output pada layar.



`\n` berfungsi untuk menambahkan baris baru pada output. Jika tidak ada `\n` sebelum tanda petik maka output akan tetap berada pada baris yang sama walaupun dalam inputan berada pada baris baru.

`;` berfungsi untuk menandai akhir dari setiap pernyataan atau statement dan bukan sebagai pemisah antara dua pernyataan.

`return 0;` berfungsi sebagai akhir dari eksekusi fungsi, juga sebagai konvensi yang menunjukkan bahwa program dijalankan tanpa kesalahan.

- **Python**

`int(input(""))` berfungsi untuk membaca dan menyimpan input yang akan disimpan sebagai fungsi integer.

`print("")` berfungsi untuk menampilkan suatu keluaran atau mencetak output pada layar.

`if`, `elif`, dan `else` adalah fungsi kondisional yang berguna untuk mengontrol alur program berdasarkan kondisi tertentu.

`if` untuk memeriksa kondisi. Jika kondisi tersebut benar, maka kode di dalamnya akan dieksekusi.

`elif` digunakan untuk memeriksa kondisi tambahan jika kondisi sebelumnya salah.

`else` berfungsi untuk menangani semua kasus yang tidak terpenuhi oleh kondisi yang ada di `if` dan `elif`.