

Nama : Minhajul Yusri Khairi

NIM : H071221006

Grup Praktikum B

Asistensi 13

A.) Fibonacci Sequence 19 (10 Poin)

Input:

```
3  a = int(input(""))
4
5  deret1 = 0
6  deret2 = 1
7
8  for i in range(a):
9      print(deret1, end=" ")
10     rumus_fibonacci = deret1 + deret2
11     deret1 = deret2
12     deret2 = rumus_fibonacci
```

Output:

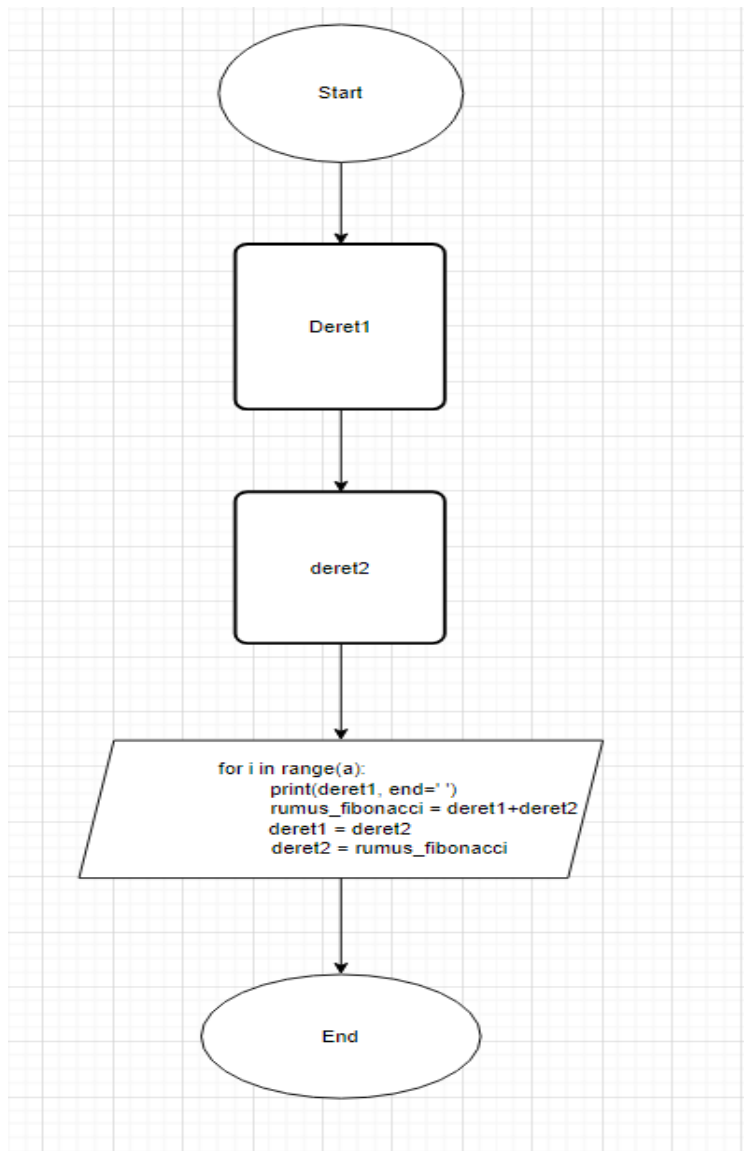
```
8
0 1 1 2 3 5 8 13
PS D:\TUGAS\PENG PEMROGRAMAN
```

Penjelasan:

1. Program menerima inputan berupa bilangan bulat.
2. Kemudian kita membuat variabel *deret1* dan *deret2* untuk menampung angka pertama dan kedua dari deret Fibonacci yang selalu bernilai 0 dan 1.
3. Kemudian kita membuat perulangan dari range pada inputan.
4. Lalu kita mencetak programnya seperti pada *line 9* agar inputan bisa horizontal.

5. Selanjutnya kita membuat variabel untuk rumus Fibonacci yaitu penjumlahan dari dua angka sebelumnya.
6. Kemudian di *line 11* artinya *deret1* akan diperbaharui dengan *deret2*.
7. Kemudian *deret2* akan diubah menjadi *rumus_fibonacci*.

Flowchart:



B. Pola Kotak (25 Poin)

Input:

```
4
5     n = int(input())
6     print("#"*n)
7     for i in range(n-2):
8         print("#"+"-"*(n-2)+"#")
9     print("#"*n)
10
```

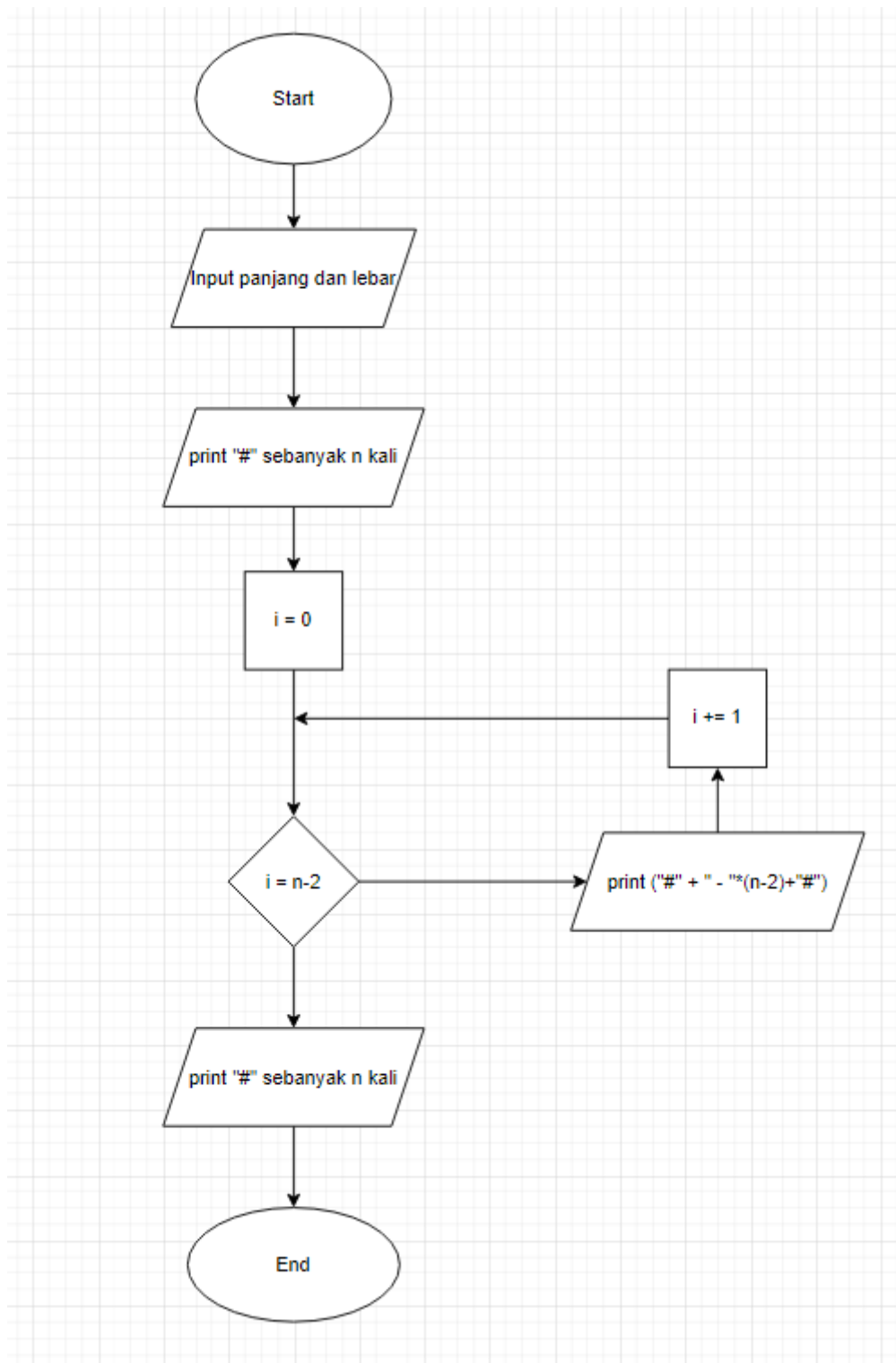
Output:

```
n/kotak.py"
4
####
# - - #
# - - #
####
PS D:\TUGAS\PENG PEMROGRAMAN\I
```

Penjelasan:

1. Program menerima inputan berupa bilangan bulat yang merupakan ukuran persegi.
2. Kemudian kita mencetak bagian atas persegi.
3. Kemudian kita mencetak bagian tengah kotak menggunakan perulangan.
4. Terakhir kita mencetak bagian bawah persegi.

Flowchart:



C. Lampu Sederhana (15 Poin)

Input:

```
1  import math
2
3  def kpk(a, b):
4      atas = a * b
5      bawah = math.gcd(a, b)
6
7      hasil = atas / bawah
8
9      return int(hasil)
10
11 n = list(map(int, input().split()))
12 a = n[0]
13 b = n[1]
14 print(kpk(a, b))
```

Output:

```
2 3
6
PS D:\TUGAS\PENG PEMROGRAMAN\L
```

Penjelasan:

1. Pertama kita mengimpor library math untuk melakukan perhitungan matematika.
2. Kemudian kita membuat function kpk yang berisi parameter a dan b.
3. Kita mengalikan kedua bilangan yang kita uji.
4. Kemudian kita mencari kpk menggunakan method gcd yang di dapat dari import math yang sudah dipanggil modulnya.
5. Kemudian kita mengembalikan nilai kpk nya menggunakan keyword "return".

6. Kemudian kita menginput bilangan bulat yang dimasukkan kedalam list kemudian diubah kembali menjadi bentuk biasa. Split berfungsi untuk menginput 2 atau lebih elemen yang dipisah dengan tanda spasi.
7. Kemudian kita membuat variabel baru yaitu “a” dan “b” untuk menguji bilangan untuk mencari kpk.

Flowchart:

