**PRAKTIKUM ALGORITMA STRUKTUR DATA**

**MODUL 7**

***Stacks and Queues***

****

**Disusun oleh:**

**Adinda Aulia Hapsari**

**L200220037**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS KOMUNIKASI DAN INFORMATIKA**

**UNIVERSITAS MUHAMMADIYAH SURAKARTA**

**TAHUN 2024**

**8.8 Soal-soal untuk Mahasiswa**

**Stacks**

1. Buatlah program untuk mengubah representasi suatu bilangan dari basis sepuluh kebasis dua. Berikut ini contoh pemanggilannya.

1 >>> cetakHexa(12)

2 ’C’

3 >>> cetakHexa(31)

4 ’1F’

5 >>> cetakHexa(229)

6 ’E5’

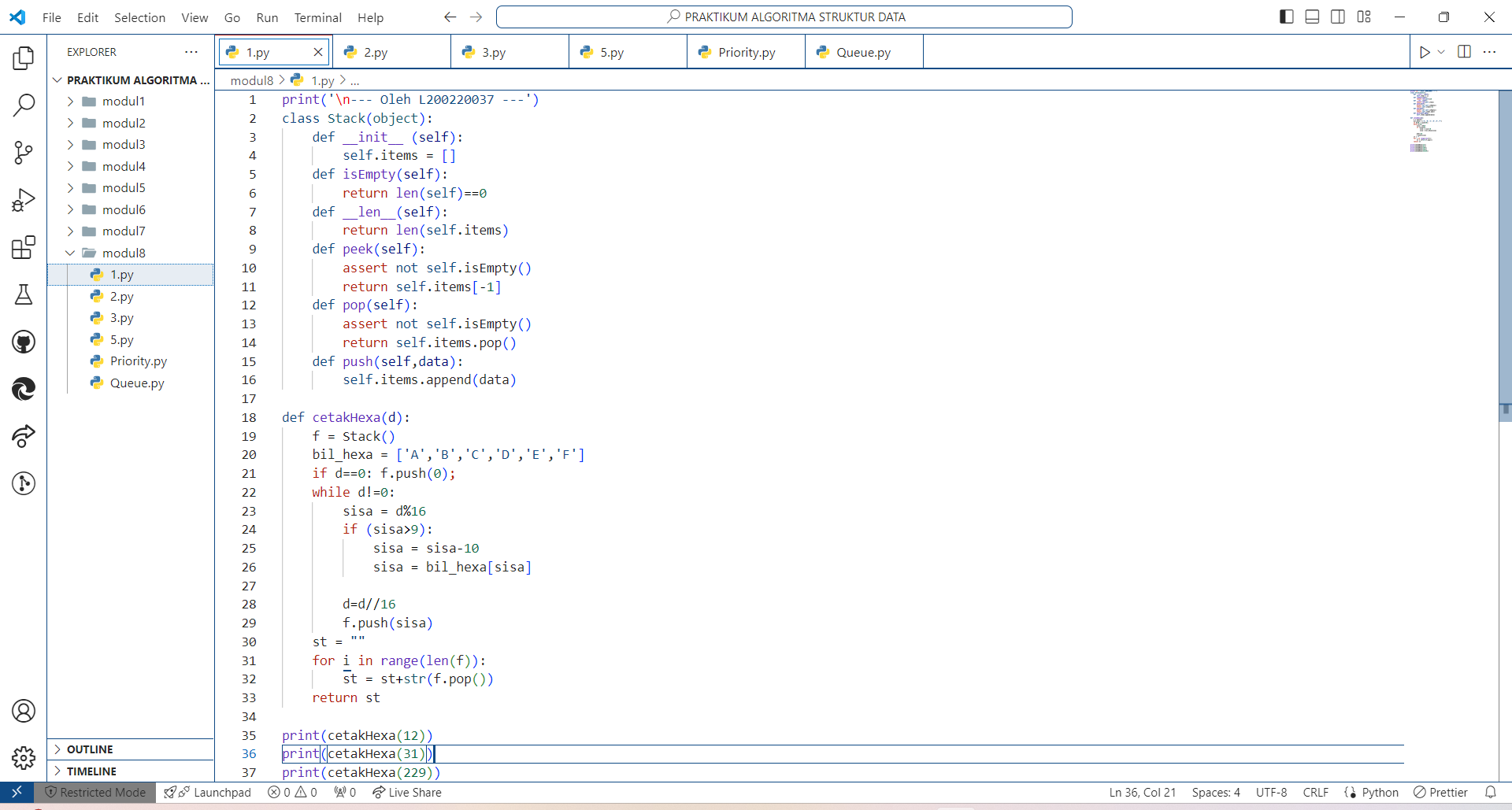
7 >>> cetakHexa(255)

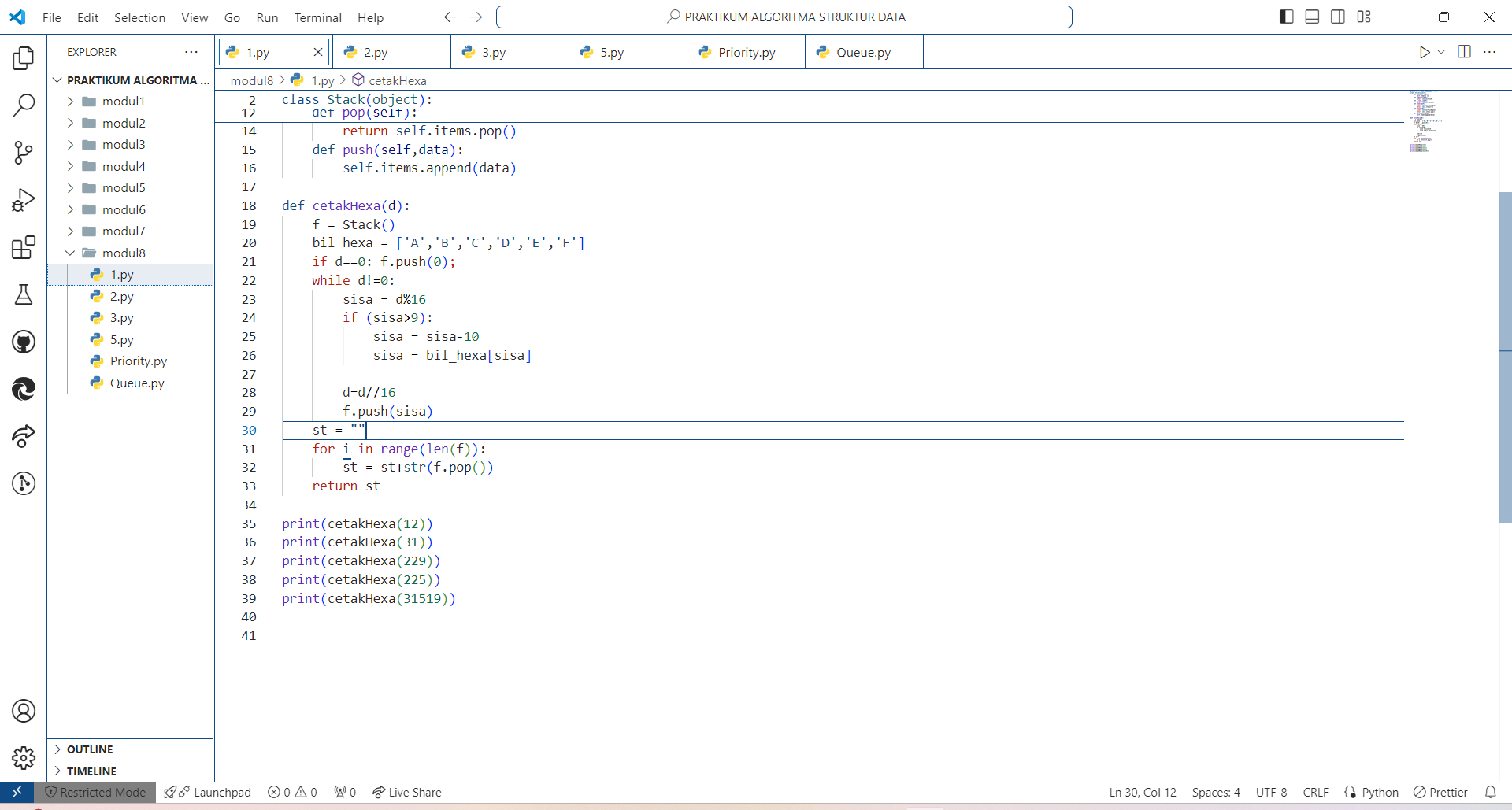
8 ’FF’

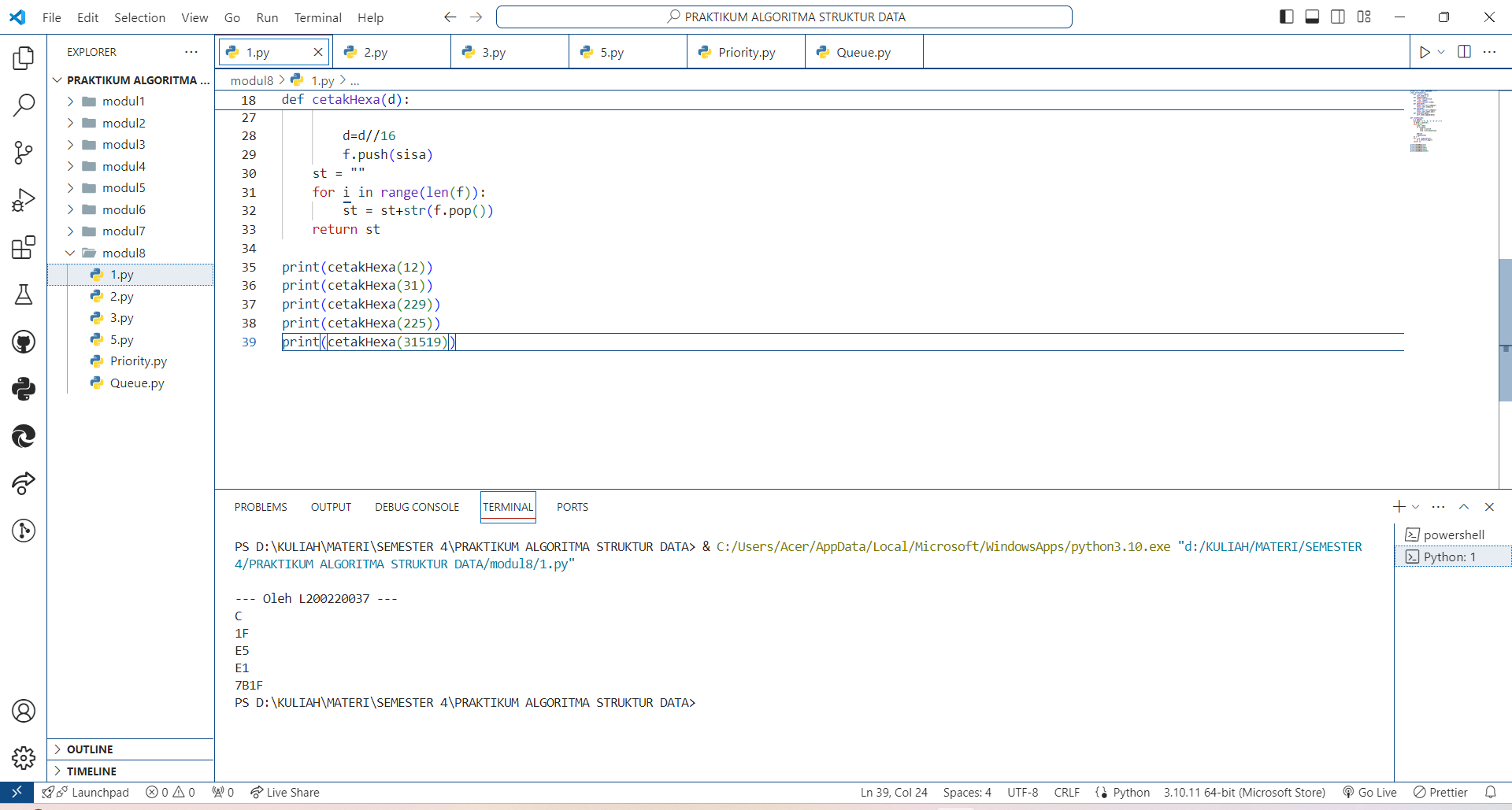
9 >>> cetakHexa(31519)

10 ’7B1F’

Perhatikan bahwa sisa pembagian tidak hanya 0 dan 1, namun bisa 0 sampai 9 dan bahkan 10, 11, 12, 13, 14, 15. Kamu harus memetakan angka-angka yang lebih dari 9 kelambang A, B, C, D, E, dan F.







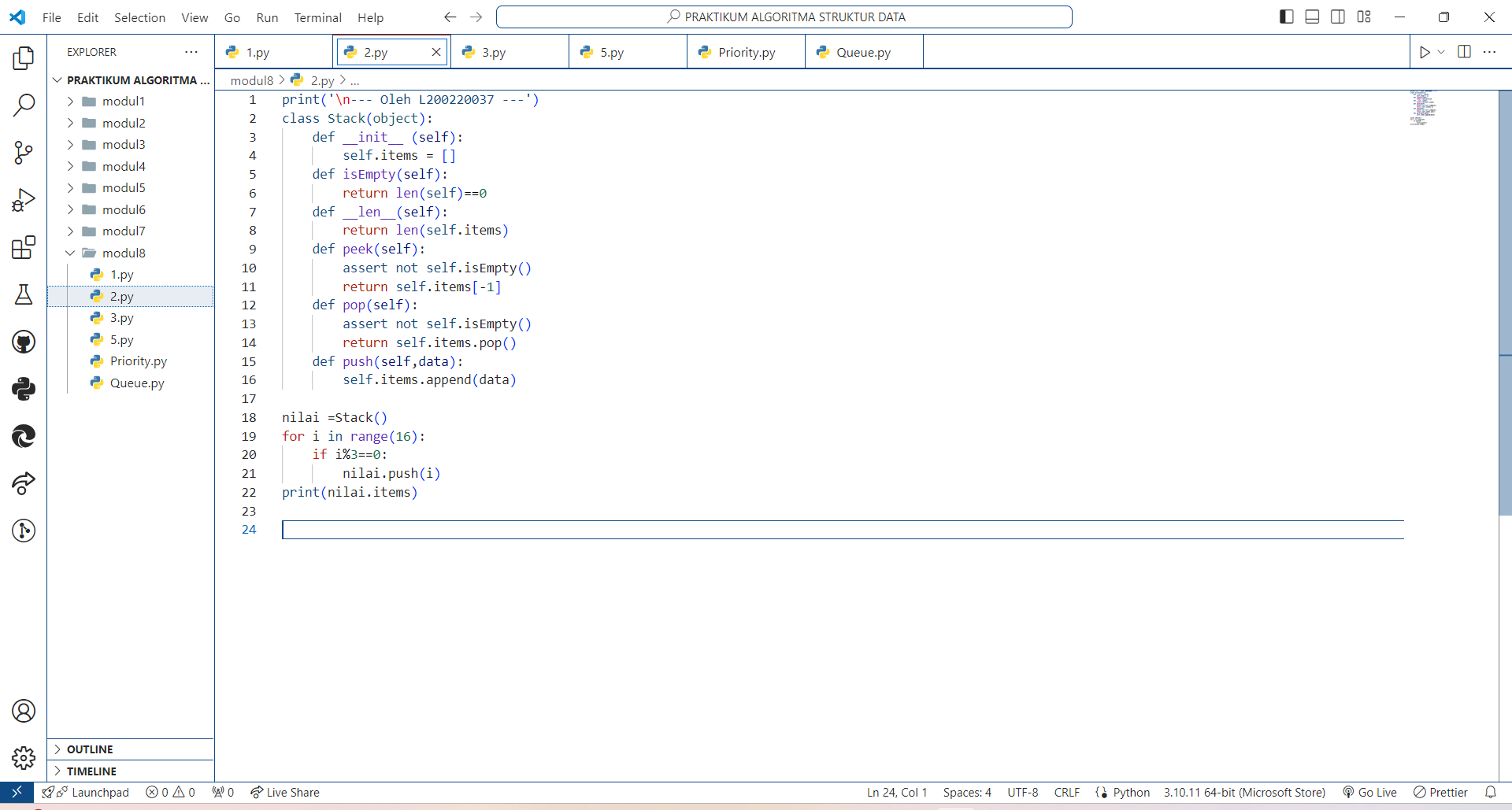
1. Eksekusi program berikut dengan pensil dan kertas, dan tunjukkan isi stack-nya pada setiap langkah.

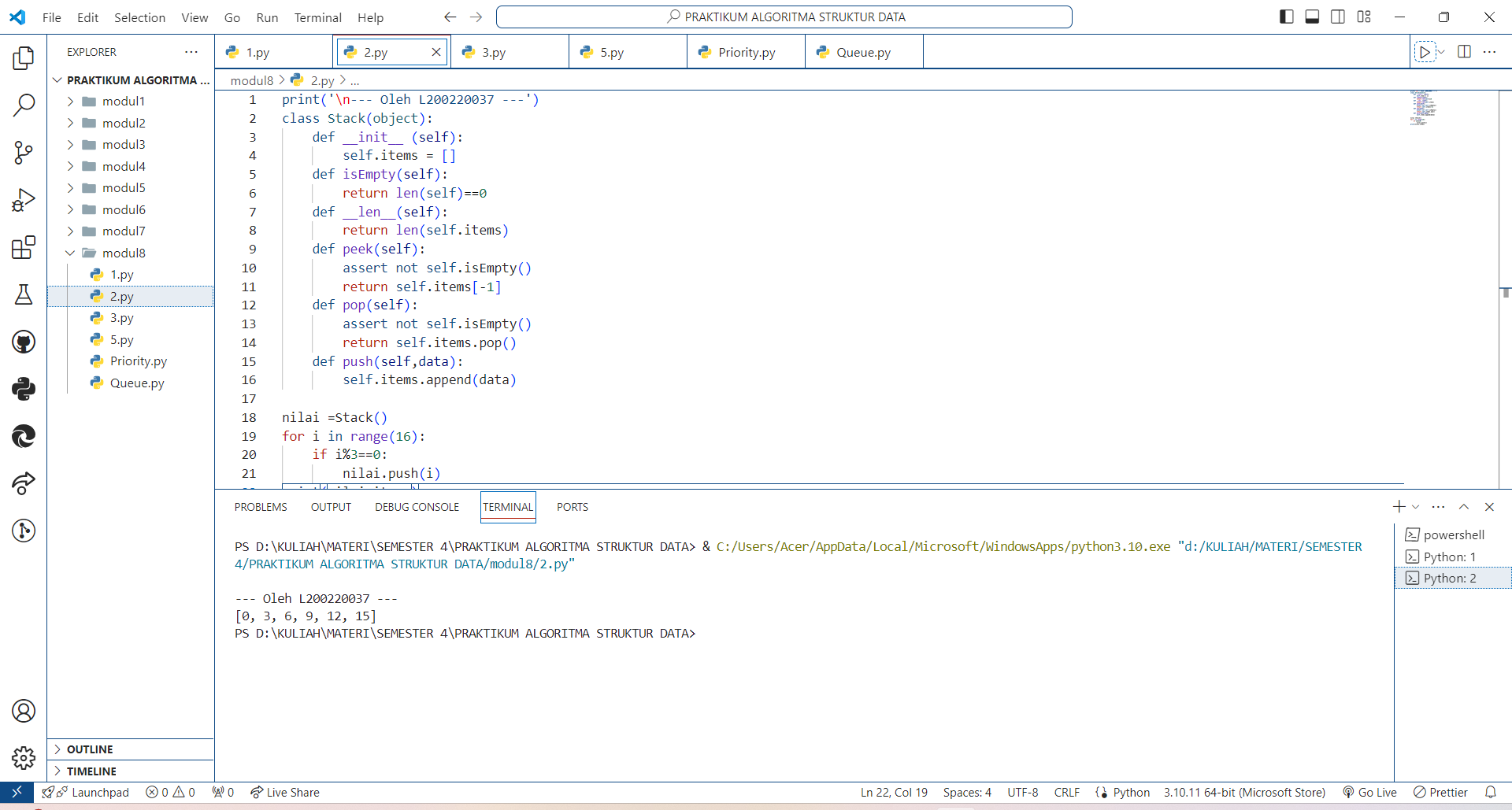
1 nilai = Stack()

2 for i in range(16 )

3 if i % 3 == 0:

4 nilai.push( i )





1. Eksekusi program berikut dengan pensil dan kertas, dan tunjukkan isi stack-nya pada setiap langkah.

1 nilai = Stack()

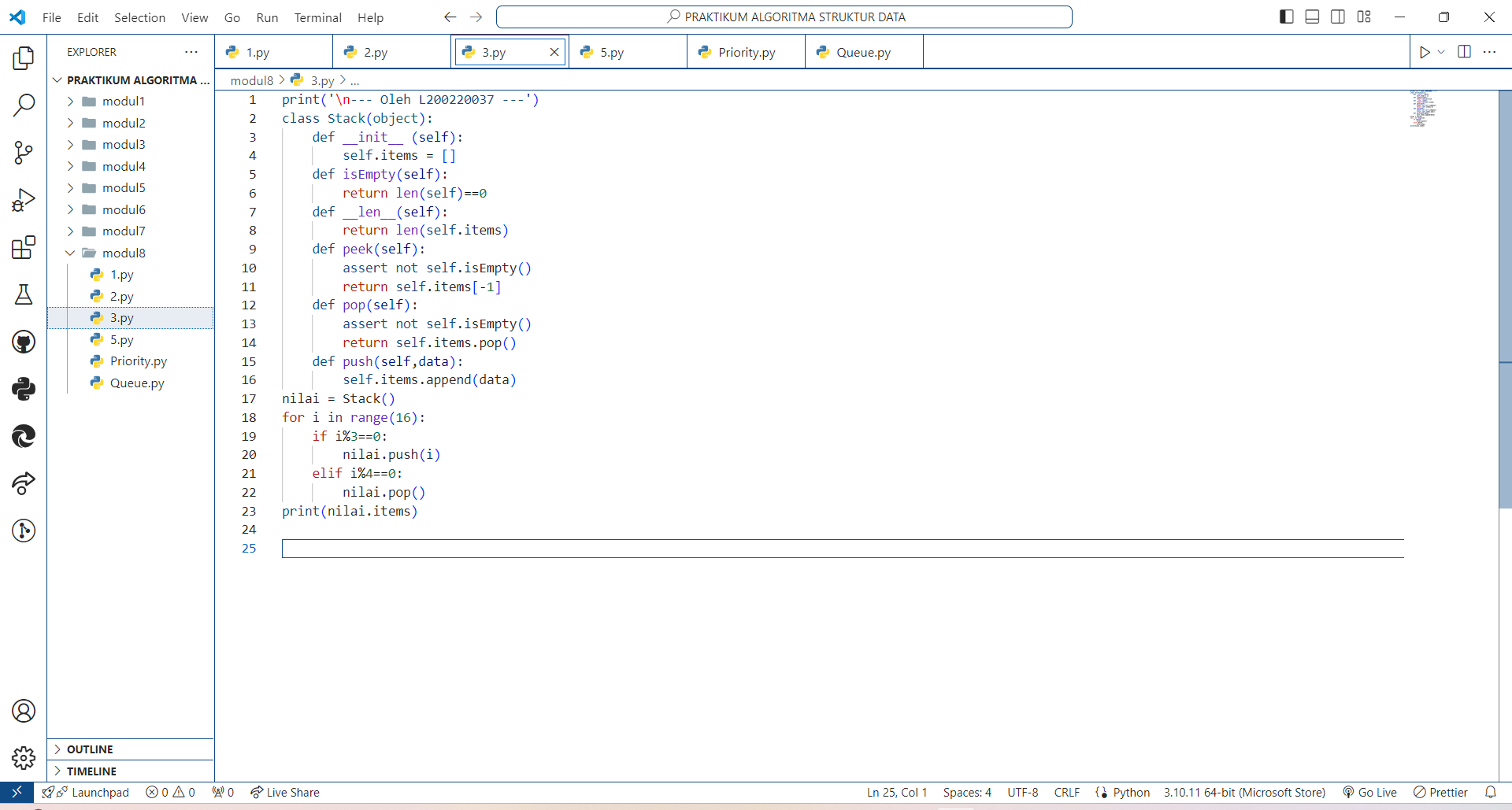
2 for i in range( 16 ) :

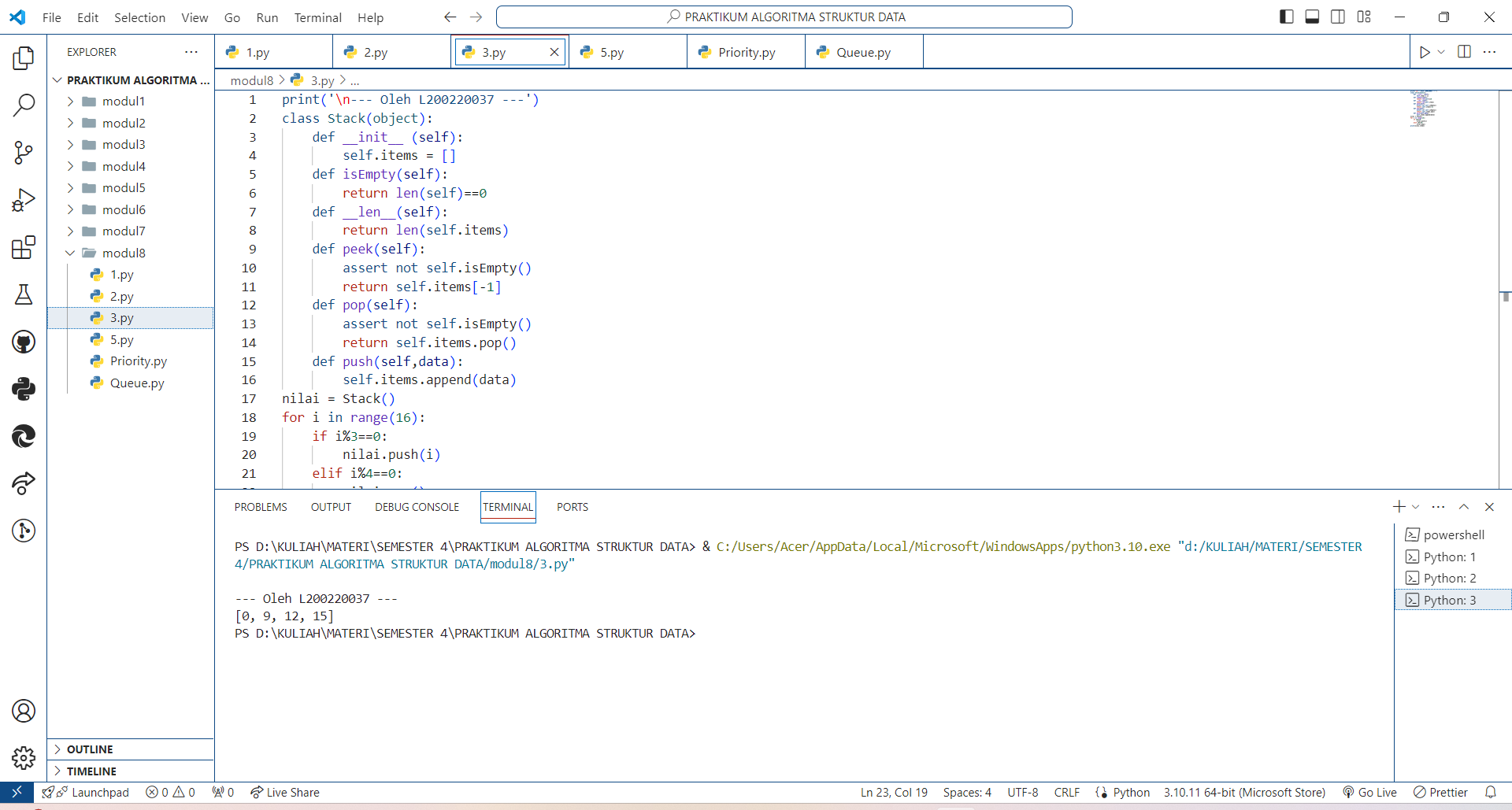
3 if i % 3 == 0 :

4 nilai.push( i )

5 elif i % 4 == 0 :

6 nilai.pop()





**Queues**

1. Tulis dua metode berikut keclass Queue dan class Priority Queue diatas.

• Metode untuk mengetahui item yang paling depan tanpa menghapusnya

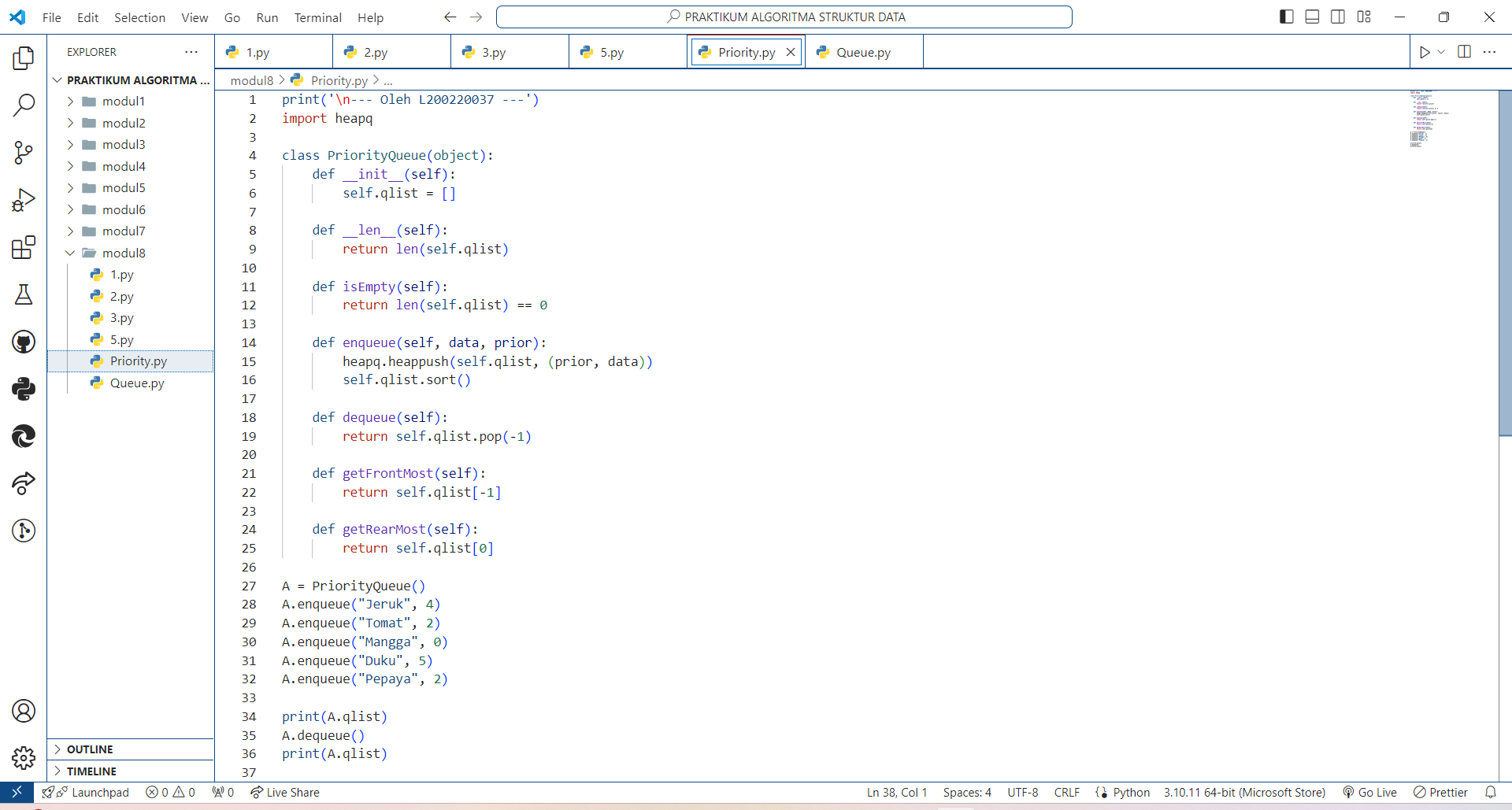
def getFrontMost(self) :

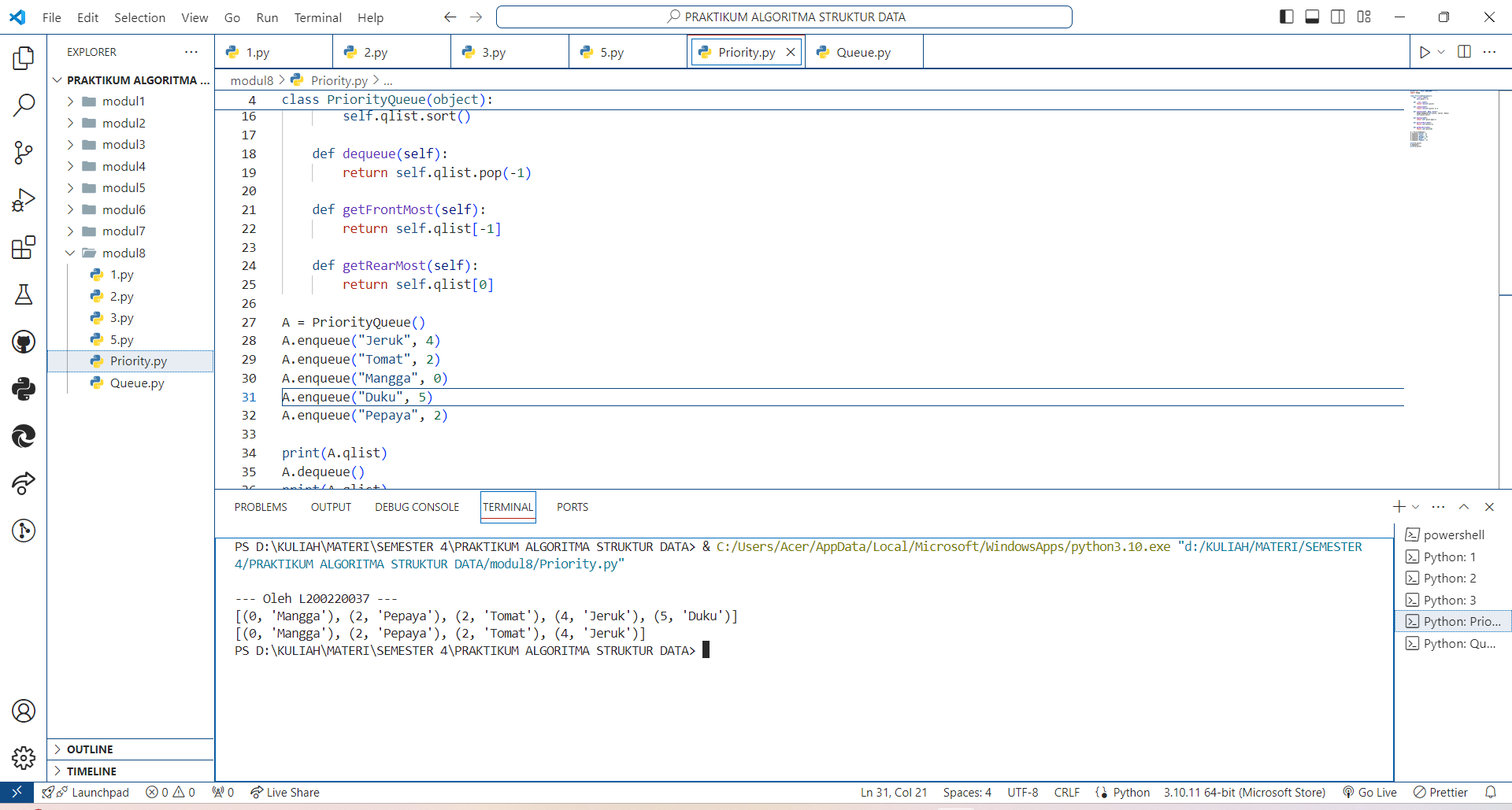
## Tulis perintahnya di sini

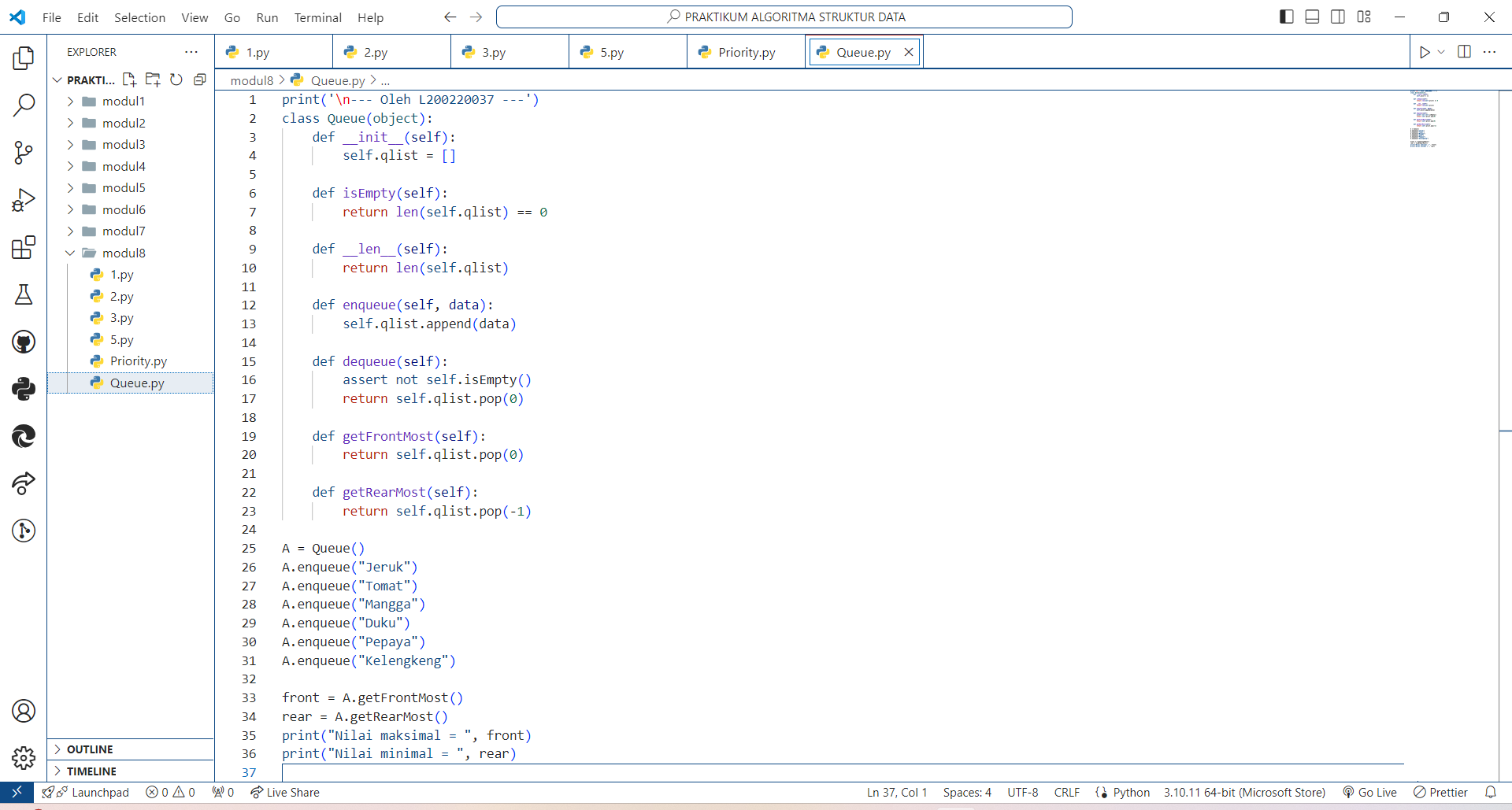
• Metode untuk mengetahui item yang paling belakang tanpa menghapusnya

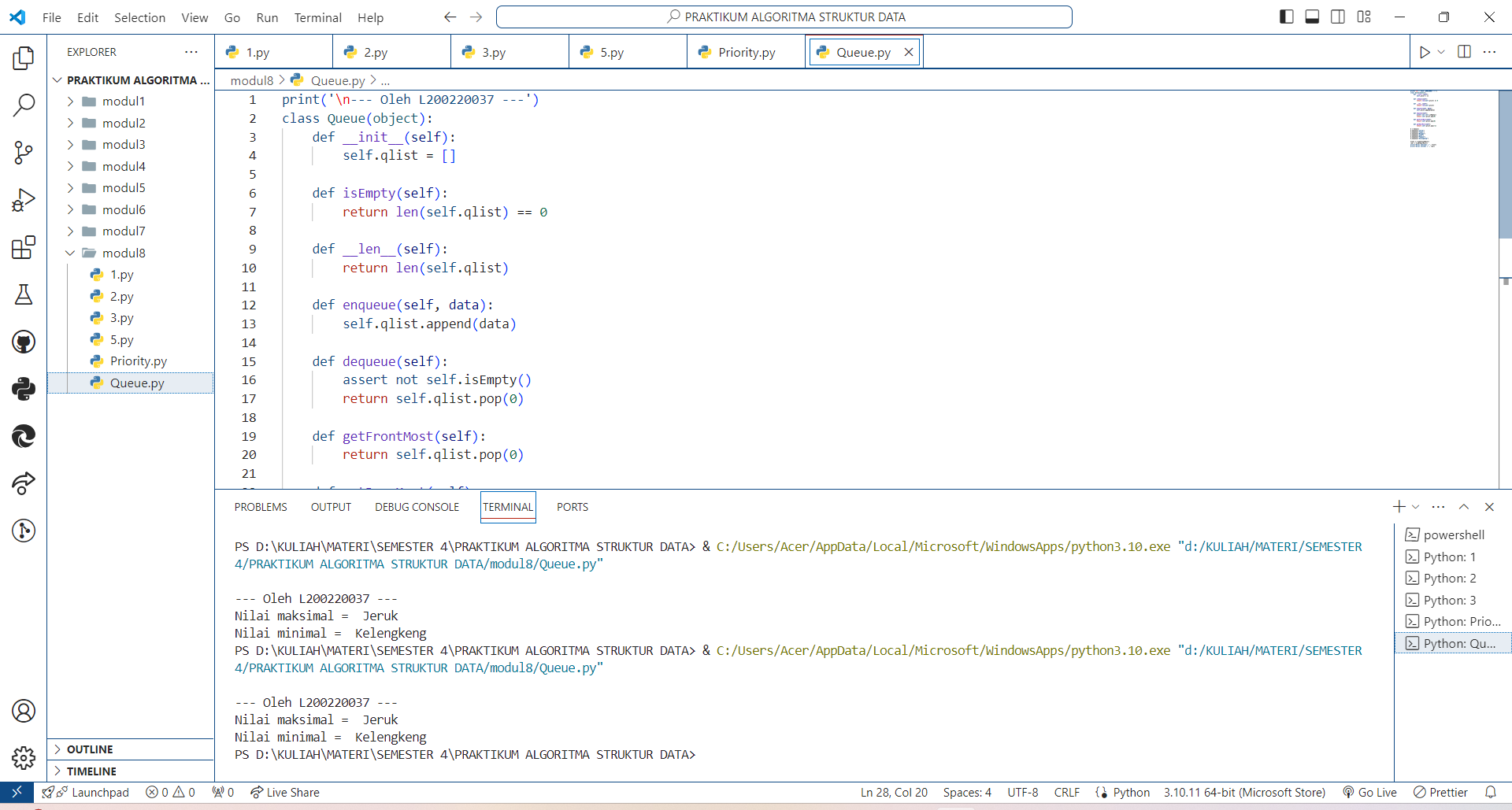
def getRearMost(self) :

## Tulis perintahnya di sini









1. Pada class Priority Queue diatas, metode dequeue() belum diimplementasikan. Tulislah metode dequeue() ini dengan memperhatikan syarat-syarat seperti yang telah dicantumkan dihalaman81.

