

# LAPORAN TUGAS AKHIR SEMESTER 2

## Struktur Data

### QUEUE



Kelompok 2 :

- |                          |          |
|--------------------------|----------|
| 1. Ahmad Rizal Firdaus   | 15220227 |
| 2. Ari Setia Saputra     | 15220569 |
| 3. Fikrian Nur Abdullah  | 15220611 |
| 4. Juli Martinus Zega    | 15220614 |
| 5. Muhammad Amin Fadilah | 15220096 |

ILMU KOMPUTER

Fakultas Teknik & Informatika  
Universitas Bina Sarana Informatika Kramat 98  
2022-2023

## **KATA PENGANTAR**

Puji dan syukur kami panjatkan kepada Tuhan Yang Maha Esa, yang telah melimpahkan rahmat dan hidayah-Nya kepada kami, sehingga kami dapat menyelesaikan tugas proposal bidang studi struktur data dengan baik.

Proposal ini telah kami susun dengan maksimal tanpa ada halangan yang berarti. Proposal ini disusun untuk memenuhi tugas akhir mata kuliah semester 2 yang diampu oleh Bapak Jaffry Arisfiandi Sumarna, ST. MM, selaku dosen mata kuliah Struktur Data. Untuk itu kami menyampaikan banyak terima kasih kepada semua pihak yang telah membantu dalam pembuatan proposal ini.

Terlepas dari semua itu, Kami menyadari sepenuhnya bahwa masih ada kekurangan baik dari segi susunan kalimat maupun tata bahasanya. Oleh karena itu kami menerima segala saran dan kritik dari pembaca agar penulis dapat memperbaiki proposal ini.

Kami berharap semoga proposal tentang struktur data yang baik ini dapat memberikan manfaat maupun inspirasi terhadap pembaca.

Jakarta, 26 Juni 2023

Penulis,

## DAFTAR ISI

DAFTAR ISI.....	3
BAB 1 PENDAHULUAN .....	4
1.1 Latar Belakang .....	4
1.2 Identifikasi Masalah .....	4
1.3 Tujuan.....	4
BAB 2 PEMBAHASAN.....	5
2.1 Landasan Teori.....	5
2.2 Jenis-Jenis Queue .....	5
2.3 Operasi Enqueue dan Dequeue .....	6
2.4 Coding .....	6
2.5 User Interface .....	9
2.6 Fungsi dan Kegunaan .....	14
2.7 Kelebihan dan Kekurangan .....	14
BAB 3 PENUTUP .....	15

# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Program yang kami buat adalah untuk menjalankan program antrian atau queue menggunakan prosedur coding bahasa pemrograman Python.

Program ini dibuat untuk memudahkan pengguna dalam mengelola antrian di berbagai bidang dan tempat seperti perpustakaan, rumah sakit, gudang, dan sebagainya.

### **1.2 Identifikasi Masalah**

- Menggunakan perulangan bersarang (Nested Loop)
- Menggabungkan beberapa method untuk menginisialisasi program
- Menggunakan try dan except untuk menghindari adanya error

### **1.3 Tujuan**

Tujuan disusunnya laporan ini adalah untuk menyelesaikan salah satu tugas mata pembelajaran kuliah khususnya mata pelajaran Struktur Data. Mengenali atau menggunakan Visual Studio Code dengan baik serta program sederhana menggunakan Python.

## **BAB 2**

### **PEMBAHASAN**

#### **2.1 Landasan Teori**

- Queue, struktur data linier yang menerapkan prinsip operasi dimana elemen data yang masuk pertama akan keluar lebih dulu
- Enqueue, Menambahkan elemen ke akhir antrian
- Dequeue, Menghapus elemen dari depan antrian
- IsEmpty, Memeriksa apakah antrian kosong
- IsFull, Memeriksa apakah antrian sudah penuh
- Peek, Mendapatkan nilai bagian depan antrian tanpa menghapusnya
- Initialize, Membuat antrian baru tanpa elemen data (kosong) Method, jenis fungsi khusus yang di definisikan dalam definisi kelas

#### **2.2 Jenis-Jenis Queue**

##### **1. Simple Queue**

Simple queue adalah struktur data queue paling dasar di mana penyisipan item dilakukan di simpul belakang (rear atau tail) dan penghapusan terjadi di simpul depan (front atau head).

##### **2. Circular Queue**

Pada circular queue, simpul terakhir terhubung ke simpul pertama. Queue jenis ini juga dikenal sebagai Ring Buffer karena semua ujungnya terhubung ke ujung yang lain. Penyisipan terjadi di akhir antrian dan penghapusan di depan antrian.

##### **3. Priority Queue**

Priority Queue adalah struktur data queue dimana simpul akan memiliki beberapa prioritas yang telah ditentukan. Simpul dengan prioritas terbesar akan menjadi yang pertama dihapus dari antrian. Sedangkan penyisipan item terjadi sesuai urutan kedatangannya.

##### **4. Double-Ended Queue (Deque)**

Dalam double-ended queue (deque), operasi penyisipan dan penghapusan dapat terjadi di ujung depan dan belakang dari queue.

## 2.3 Operasi Enqueue dan Dequeue

Di bawah ini adalah langkah-langkah untuk enqueue (memasukkan) data ke antrian.

- Periksa apakah antrian sudah penuh atau tidak.
- Jika antrian penuh – cetak kesalahan overflow dan keluar dari program.
- Jika antrian tidak penuh – naikan pointer belakang untuk menunjuk ke ruang kosong berikutnya.
- Tambahkan elemen pada posisi yang ditunjuk oleh pointer belakang.
- Kembalikan status bahwa penambahan telah berhasil

Di bawah ini adalah langkah-langkah untuk dequeue (mengeluarkan) data dari antrian.

- Periksa apakah antrian sudah penuh atau tidak.
- Jika antrian kosong – cetak kesalahan underflow dan keluar dari program.
- Jika antrian tidak kosong – akses elemen data yang ditunjuk oleh pointer depan.
- Geser pointer depan untuk menunjuk ke elemen data berikutnya yang tersedia.
- Kembalikan status bahwa operasi penghapusan telah berhasil

## 2.4 Coding

```
1  try:
2      class Queue:
3          def __init__(self,n=30):
4              self.size = n
5              self.current_size = 0
6              self.data = []
7
8          def isFull(self):
9              if self.current_size == self.size:
10                 return True
11             else:
12                 return False
13         def isEmpty(self):
14             if self.current_size == 0:
15                 return True
16             else:
17                 return False
```

```

20     def enqueue(self,n):
21         if self.isFull():
22             print("Maaf antrian sudah penuh")
23         else:
24             self.data.append(n)
25             self.current_size = len(self.data)
26             print("Antrian dengan nama ",n,"telah berhasil ditambahkan")
27         print("Tekan [Enter] untuk melanjutkan")
28         input()
29         self.menu()
30     def dequeue(self):
31         if self.isEmpty():
32             print("Maaf antrian masih kosong")
33             return None
34         else:
35             datakeluar = self.data.pop(0)
36             self.current_size = len(self.data)
37             print("-----")
38             print("Antrian dengan nama :",datakeluar)
39             print("Mohon untuk mengambil barang")
40             print("-----")
41             print("Antrian setelah ini adalah :",self.data)
42         print("Tekan [Enter] untuk melanjutkan")
43         input()
44         self.menu()

```

```

47     def view(self):
48         if self.isEmpty():
49             print("Maaf antrian masih kosong")
50         else:
51             print("===== DAFTAR ANTRIAN =====")
52             X = 1
53             for i in self.data:
54                 print(" "+str(X)+" ". ,i)
55                 X +=1
56             print("Total Antrian :", len(self.data))
57         print("Tekan [Enter] untuk melanjutkan")
58         input()
59         self.menu()
60
61     def Exit(self):
62         import sys
63         sys.exit()

```

```

65         def menu(self):
66             import os
67             os.system("CLS")
68             print("=====")
69             print("|      PROGRAM ANTRIAN      |")
70             print("|----- Daftar Menu -----|")
71             print("|1. Tambah Antrian          |")
72             print("|2. Panggil Antrian         |")
73             print("|3. Lihat Daftar Antrian    |")
74             print("|4. Exit                    |")
75             print("=====")
76

```

```

77         pil = int(input("Masukan no menu yang ingin dipilih: "))
78
79         if pil == 1:
80             data = input("Masukkan daftar nama antrian : ")
81             self.enqueue(data)
82         elif pil == 2:
83             self.dequeue()
84         elif pil == 3:
85             self.view()
86         elif pil == 4:
87             self.Exit
88         else:
89             print("Maaf pilihan yang anda ketikan tidak ada, silahkan ulangi kembali")
90             print("Tekan [Enter] untuk kembali ke menu")
91             input()
92             self.menu
93
94         q = Queue()
95         q.menu()
96     except ValueError:
97         print("Pilihan hanya berupa angka")
98         print("Silahkan mulai kembali program!")

```



## 2.5 User Interface

### Main Menu

```
=====
|          PROGRAM ANTRIAN          |
|----- Daftar Menu -----|
|1. Tambah Antrian                |
|2. Panggil Antrian                |
|3. Lihat Daftar Antrian          |
|4. Exit                          |
=====
Masukan no menu yang ingin dipilih:
```

### Tambah Antrian

```
=====
|          PROGRAM ANTRIAN          |
|----- Daftar Menu -----|
|1. Tambah Antrian                |
|2. Panggil Antrian                |
|3. Lihat Daftar Antrian          |
|4. Exit                          |
=====
Masukan no menu yang ingin dipilih: 1
Masukkan daftar nama antrian : amin
Antrian dengan nama  amin telah berhasil ditambahkan
Tekan [Enter] untuk melanjutkan
█
```

### Panggil Antrian

```
=====
|   PROGRAM ANTRIAN   |
|----- Daftar Menu -----|
|1. Tambah Antrian    |
|2. Panggil Antrian    |
|3. Lihat Daftar Antrian |
|4. Exit              |
|                     |
=====
Masukan no menu yang ingin dipilih: 2
-----
Antrian dengan nama : amin
Mohon untuk mengambil barang
-----
Antrian setelah ini adalah : []
Tekan [Enter] untuk melanjutkan
█
```

### Kondisi Antrian berjumlah 2

```
=====
|   PROGRAM ANTRIAN   |
|----- Daftar Menu -----|
|1. Tambah Antrian    |
|2. Panggil Antrian    |
|3. Lihat Daftar Antrian |
|4. Exit              |
|                     |
=====
Masukan no menu yang ingin dipilih: 3
===== DAFTAR ANTRIAN =====
1. riz
2. fik
Total Antrian : 2
Tekan [Enter] untuk melanjutkan
█
```

```
=====
|      PROGRAM ANTRIAN      |
|----- Daftar Menu -----|
|1. Tambah Antrian         |
|2. Panggil Antrian        |
|3. Lihat Daftar Antrian   |
|4. Exit                   |
|                           |
=====
Masukan no menu yang ingin dipilih: 2
-----
Antrian dengan nama : riz
Mohon untuk mengambil barang
-----
Antrian setelah ini adalah : ['fik']
Tekan [Enter] untuk melanjutkan
█
```

Exit

```
=====
|      PROGRAM ANTRIAN      |
|----- Daftar Menu -----|
|1. Tambah Antrian         |
|2. Panggil Antrian        |
|3. Lihat Daftar Antrian   |
|4. Exit                   |
|                           |
=====
Masukan no menu yang ingin dipilih: 4
PS C:\Users\fikri> █
```

### Salah Input

```
=====
|      PROGRAM ANTRIAN      |
|----- Daftar Menu -----|
|1. Tambah Antrian         |
|2. Panggil Antrian        |
|3. Lihat Daftar Antrian   |
|4. Exit                   |
=====
```

Masukan no menu yang ingin dipilih: 5

Maaf pilihan yang anda ketikan tidak ada, silahkan ulangi kembali  
Tekan [Enter] untuk kembali ke menu



```
=====
|      PROGRAM ANTRIAN      |
|----- Daftar Menu -----|
|1. Tambah Antrian         |
|2. Panggil Antrian        |
|3. Lihat Daftar Antrian   |
|4. Exit                   |
=====
```

Masukan no menu yang ingin dipilih: y

Pilihan hanya berupa angka

Silahkan mulai kembali program!

PS C:\Users\fikri>

### Antrian Kosong

```
=====
|      PROGRAM ANTRIAN      |
|----- Daftar Menu -----|
|1. Tambah Antrian         |
|2. Panggil Antrian        |
|3. Lihat Daftar Antrian   |
|4. Exit                   |
|                           |
=====

Masukan no menu yang ingin dipilih: 3
Maaf antrian masih kosong
Tekan [Enter] untuk melanjutkan
█
```

### Antrian Penuh

```
=====
|      PROGRAM ANTRIAN      |
|----- Daftar Menu -----|
|1. Tambah Antrian         |
|2. Panggil Antrian        |
|3. Lihat Daftar Antrian   |
|4. Exit                   |
|                           |
=====

Masukan no menu yang ingin dipilih: 1
Masukkan daftar nama antrian : jul
Maaf antrian sudah penuh
Tekan [Enter] untuk melanjutkan
█
```

## 2.6 Fungsi dan Kegunaan

- Queue banyak digunakan untuk menangani lalu lintas (traffic) situs web.
- Membantu untuk mempertahankan playlist yang ada pada aplikasi media player
- Queue digunakan dalam sistem operasi untuk menangani interupsi.
- Membantu dalam melayani permintaan pada satu sumber daya bersama, seperti printer, penjadwalan tugas CPU, dll.
- Digunakan dalam transfer data asinkronus misal pipeline, IO file, dan socket.

## 2.7 Kelebihan dan Kekurangan

Di bawah ini adalah kelebihan queue.

- Data dalam jumlah besar dapat dikelola secara efisien.
- Operasi seperti penyisipan dan penghapusan dapat dilakukan dengan mudah karena mengikuti aturan masuk pertama keluar pertama.
- Berguna ketika layanan tertentu digunakan oleh banyak konsumen.
- Cepat untuk komunikasi antar-proses data.
- Dapat digunakan dalam implementasi struktur data lainnya.

Di bawah ini adalah kekurangan queue.

- Operasi seperti penyisipan dan penghapusan elemen dari tengah cenderung banyak memakan waktu.
- Dalam queue konvensional, elemen baru hanya dapat dimasukkan ketika elemen yang ada dihapus dari antrian.
- Mencari elemen data pada struktur queue membutuhkan time complexity  $O(N)$ .
- Ukuran maksimum antrian harus ditentukan sebelumnya.

## **BAB 3**

### **PENUTUP**

Dari pembahasan di atas dapat disimpulkan bahwa program berbasis Python yang kami buat bisa digunakan dan diterapkan di berbagai tempat dan kondisi karena berupa program antrian.

Kami menambahkan opsi dimana ketika salah input, akan tampil notifikasi salah input. Sehingga memudahkan pengguna untuk memulai kembali program.

Demikian proposal yang kami buat, kami harap proposal ini bisa bermanfaat bagi semuanya dan dapat menambah pengetahuan mengenai bagaimana menjalankan sebuah program antrian dalam Visual Studio Code. Sekian dan terimakasih.