

Praktikum Algoritma & Struktur Data

Tugas 3. Queue

Dosen Pengampu

Dr. Tita Karlita S.Kom, M.Kom



Disusun Oleh :

Nama : M. Faza Nur Husain

Nrp : 3121550004

**D3 PJJ AK TEKNIK INFORMATIKA
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA
TAHUN AKADEMIK 2021/2022**

Tugas 3. Queue

Upload implementasi code queue dengan array. tampilkan output

Source Code :

```
// implementasi queue dengan menggunakan array
#include <stdio.h>
#include <stdlib.h>
#define MAX 5 // definisi maksimal ukuran queue

typedef struct{ // struktur data queue dlm bentuk struct
    char item[MAX]; // elemen item karakter maksimal 5
    int front; // front untuk menunjuk elemen awal yg masuk queue
    int rear; // rear untuk menunjuk elemen terakhir yg masuk queue
    int count; // count untuk menjumlah elemen yang ada dalam queue
}queue;

void inisialisasi (queue *q); // fungsi inisialisasi
int kosong (queue *q); // fungsi untuk cek queue kosong atau tidak
int penuh (queue *q); // fungsi untuk cek queue penuh atau tidak
void enqueue(char x, queue *q); // fungsi entry data kedalam queue
void dequeue(queue *q); // fungsi untuk mengirimkan return value berupa karakter
void tampil(queue *q); // fungsi untuk menampilkan

int main(){
    queue antrian;
    char x;
    int i;
    int pilihan;

    inisialisasi(&antrian);
    do{
        puts("MENU QUEUE using ARRAY:");
        puts("1. Mengisi QUEUE (Enqueue)");
        puts("2. Mengambil isi QUEUE (Dequeue)");
        puts("3. Menampilkan isi QUEUE");
        puts("4. Keluar\n");

        fflush(stdin);
        printf("Masukkan Pilihan Anda : ");
        scanf("%d", &pilihan);
        switch(pilihan)
        {
            case 1:
                fflush(stdin);
                printf("Elemen yang dimasukan : ");
                scanf("%c", &x);
                enqueue(x, &antrian);
                break;
            case 2:
                dequeue(&antrian);
                break;
        }
    } while (pilihan != 4);
}
```

```

        case 3:
            printf("Isi dari Queue\n");
            tampil(&antrian);
            break;
        case 4:
            printf("Keluar dari Program ...");
            exit(0);
        default:
            puts("Pilihan salah, mohon ulangi ");
            break;
    }
    puts("");
}while(pilihan != 4);
}

void inisialisasi (queue *q){    // meng set mula dari count, front, rear
    q->count = 0;
    q->front = 0;
    q->rear = 0;
}

int kosong (queue *q){          // mengecek apakah queue kosong atau tidak
    if (q->count == 0)
        return 1;
    else
        return 0;
}

int penuh (queue *q){           // mengecek apakah queue penuh atau tidak
    if (q->count == MAX)
        return 1;
    else
        return 0;
}

void enqueue(char x, queue *q){    // cek apakah enqueue penuh atau tidak
    if(penuh(q))
        printf("Queue Penuh\n");    // jika penuh menampilkan ket Queue Penuh
    else{                            // jika tidak penuh bisa ditaruh
        q->item[q->rear] = x;        // simpan data di index rear
        q->rear = (q->rear + 1) % MAX; // index rear ditambah 1
        (q->count) ++;              // setelah data ditaruh data di increment
count nya
    }
}

void dequeue(queue *q){            // mengambil data
    char temp;

    if(kosong(q))                  // jika kosong tidak bisa diambil
        printf("Queue Kosong\n"); // tampil keterangan queue kosong
    else{
        temp = q->item[q->front];    // temp menampung data dari index front
        q->front = (q->front + 1) % MAX; // index front ditambah 1
        (q->count)--;                // setelah data diambil data di
increment count nya
        printf("Data yang diambil adalah : %c\n", temp);
    }
}

```

```

void tampil(queue *q){
    if(kosong(q))
        printf("Queue Kosong\n");
    else{
        for (int i = q->front; i <= q->rear; i++){
            printf("%c\n", q->item[i]);
        }
    }
}
}

```

```

G:\My Drive\COLLAGE\PENS\SEMESTER 2\Algoritma & Struktur Data\queue\kode.exe
MENU QUEUE using ARRAY:
1. Mengisi QUEUE (Enqueue)
2. Mengambil isi QUEUE (Dequeue)
3. Menampilkan isi QUEUE
4. Keluar

Masukkan Pilihan Anda : 1
Elemen yang dimasukkan : a

MENU QUEUE using ARRAY:
1. Mengisi QUEUE (Enqueue)
2. Mengambil isi QUEUE (Dequeue)
3. Menampilkan isi QUEUE
4. Keluar

Masukkan Pilihan Anda : 1
Elemen yang dimasukkan : b

MENU QUEUE using ARRAY:
1. Mengisi QUEUE (Enqueue)
2. Mengambil isi QUEUE (Dequeue)
3. Menampilkan isi QUEUE
4. Keluar

Masukkan Pilihan Anda : 1
Elemen yang dimasukkan : 3

MENU QUEUE using ARRAY:
1. Mengisi QUEUE (Enqueue)
2. Mengambil isi QUEUE (Dequeue)
3. Menampilkan isi QUEUE
4. Keluar

Masukkan Pilihan Anda : 3
Isi dari Queue
a
b
3

MENU QUEUE using ARRAY:
1. Mengisi QUEUE (Enqueue)
2. Mengambil isi QUEUE (Dequeue)
3. Menampilkan isi QUEUE
4. Keluar

Masukkan Pilihan Anda : 2
Data yang diambil adalah : a

MENU QUEUE using ARRAY:
1. Mengisi QUEUE (Enqueue)
2. Mengambil isi QUEUE (Dequeue)
3. Menampilkan isi QUEUE
4. Keluar

Masukkan Pilihan Anda : 3
Isi dari Queue
b
3

```