

MODUL 7: Antrian

1.1. Deskripsi Singkat

Queue atau antrian adalah suatu struktur data berbentuk list atau daftar yang proses pengelolaan datanya menggunakan konsep FIFO (First In First Out) atau LIFO (Last In Last Out), sehingga proses menambah data dan penghapusannya berbeda dengan tumpukan, menggunakan dua pintu. Menambahkan data ke rear atau tail dan menghapus data mulai dari head atau front. Antrian dapat diimplementasikan dengan array atau linked list.

1.2. Tujuan Praktikum

- 1) Mahasiswa mampu mengimplementasikan berbagai persoalan antrian dengan menggunakan bahasa C
- 2) Mahasiswa dapat melakukan operasi penyisipan (Enqueue) maupun penghapusan (Dequeue) item dari antrian dengan menggunakan bahasa C

1.3. Material Praktikum

Kegiatan pada modul ini memerlukan material berupa software editor dan compiler (atau IDE) untuk bahasa pemrograman C.

1.4. Kegiatan Praktikum

Pada praktikum kali ini, kita akan membuat sebuah program menggunakan queue untuk generate 1 sampai N bilangan biner.

Contoh:

N = 10

Angka Binary: 1 10 11 100 101 110 111 1000 1001 1010 (lihat pola perubahan angka-nya)

Dalam proses pembuatannya, ada 4 tahap, yaitu:

1. Persiapan,
2. Membuat fungsi generate bilangan biner,
3. Membuat fungsi enqueue,
4. Membuat fungsi dequeue,
5. Finalisasi.

A. Persiapan

Dalam tahapan ini kita harus mempersiapkan struktur data yang dibutuhkan serta fungsi lain yang akan digunakan dalam proses berikutnya. Ketik potongan program di bawah ini pada IDE Anda, lalu simpan dengan nama **praktikum7a.c**.

```
#define MAX 50
#define MAX_CHARS 16
char QUEUE[MAX_CHARS][MAX], rear, front;
```

Sampai dengan potongan program di atas, kita sudah menyiapkan struktur data berupa array yang akan digunakan dalam proses pembuatan antrian selanjutnya. Kali ini kita deklarasikan array antriannya bertipe **char**.

B. Pembuatan Fungsi Generate Bilangan Biner

```
void generateBinaryNumbers(int N)
{
    char bin[] = "1";
    enqueue(bin);

    for(int i=0; i < N; i++)
    {
        char temp[MAX_CHARS] = "";
        char temp2[MAX_CHARS] = "";

        strcpy(temp, QUEUE[front]);
        strcpy(temp2, QUEUE[front]);

        printf("%s ", temp);
        dequeue();

        strcat(temp, "0");
        enqueue(temp);

        strcat(temp2, "1");
        enqueue(temp2);
    }
}
```

Dapatkah Anda memahami algoritme pada fungsi `generateBinaryNumbers` di atas? Coba jelaskan cara kerja fungsi tersebut!

C. Pembuatan Fungsi Enqueue

Fungsi enqueue adalah fungsi utama dari program ini yang berguna untuk menyimpan rangkaian bilangan biner yang digenerate. Ada beberapa kondisi yang perlu diperhatikan berkaitan dengan penempatan antrian baru.

```
void enqueue(char item[])
{
    if(rear == MAX - 1){
        printf("Antrian penuh \n");
        return;
    }

    if(front == -1)
        front = 0;

    rear++;
    //karena antrian bertipe char, maka enqueue tidak bisa
    //dilakukan dengan cara QUEUE[rear]=item
    strcpy(QUEUE[rear], item);
}
```

D. Pembuatan Fungsi Dequeue

Untuk membuat fungsi dequeue, ada dua kondisi yang harus diperhatikan. Kondisi pertama yaitu saat antrian masih kosong dan selain itu. Jika masih kosong maka tidak bisa menghapus, jika tidak baru menghapus antrian dari depan.

```
void dequeue()
{
    if(rear == -1){
        printf("Antrian kosong \n");
        return;
    }

    if(front == rear)
        front = rear = -1;
    else{
        for(int i = 0; i < rear; i++) {
```

```

        strcpy(QUEUE[i], QUEUE[i + 1]);
    }
    rear--;
    front = 0;
}
}

```

E. Finalisasi

Langkah terakhir adalah mengatur fungsi-fungsi tersebut sehingga user bisa menginputkan jumlah bilangan biner yang ingin digenerate dan antrian bisa menyimpan hasil yang sesuai.

```

int main()
{
    rear = -1;
    front = -1;

    int n;
    printf("masukkan sebuah bilangan: ");
    scanf("%d", &n);
    generateBinaryNumbers(n);

    return 0;
}

```

1.5. Penugasan

Buatlah sebuah program antrian untuk pasien di sebuah klinik. Menggunakan program tersebut, pasien dapat:

1. Mengambil nomor antrian berobat
2. Melihat antrian yang tersisa
3. Melihat perkiraan waktu tunggu pasien pada saat mengantri untuk berobat (Perkiraan waktu pemeriksaan per pasien 15 menit)
4. Keluar dari program

Simpan program dengan nama **praktikum7b_kelas_nim.c (jangan dizip/rar)**. Unggah file tersebut di Google Classroom sesuai dengan batas waktu yang telah ditetapkan. Keterlambatan pengumpulan dikenakan sanksi pemotongan nilai sebesar 10 poin per jam.