

# **QUIZ 2**

## **STRUKTUR DATA**



**Disusun Oleh:**

**Prames Ray Lopian - 140810210059**

**PROGRAM STUDI S-1 TEKNIK INFORMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS PADJADJARAN  
JATINANGOR**

**2022**

## 1. Soal 1:

```
#include <iostream>
using namespace std;

const int maxElemen = 255;

struct Stack
{
    char isi[maxElemen];
    int TOP;
};

Stack S;

void createStack (Stack& S)
{
    S.TOP = -1;
}

void push (Stack& S, char elemenBaru)
{
    if (S.TOP == maxElemen-1)
    {
        cout <<"Stack Overflow" << endl;
    }
    else
    {
        S.TOP += 1;
        S.isi[S.TOP] = elemenBaru;
    }
}

void pop(Stack& S, char& elemenHsl)
{
    if (S.TOP < 0)
    {
        cout << "Stack Underflow " << endl;
    }
    else
    {
        elemenHsl= S.isi[S.TOP];
        S.TOP -= 1;
    }
}
```

```

}

void traversal(Stack& S)
{
    int idxBantu = S.TOP;

    while (idxBantu >= 0)
    {
        cout << "___" << endl
              << S.isi[idxBantu] << endl;
        idxBantu -= 1;
    }
}

void swap(Stack& S, int idx)
{
    char temp;
    temp = S.isi[idx];
    S.isi[idx] = S.isi[idx + 1];
    S.isi[idx + 1] = temp;
}

main()
{
    Stack myTumpukan;
    char hasil, temp;

    cout << "PROGRAM SWAP MENGGUNAKAN STACK ARRAY" << endl;

    createStack(myTumpukan);

    push(myTumpukan, 'A');
    push(myTumpukan, 'B');

    cout << "\nSebelum SWAP:" << endl;
    traversal(myTumpukan);

    cout << "\nSetelah SWAP:" << endl;
    swap(myTumpukan, 0);
    traversal(myTumpukan);
}

```

## PROGRAM SWAP MENGGUNAKAN STACK ARRAY

Sebelum SWAP:

--  
B

--  
A

Setelah SWAP:

--  
A

--  
B

## 2. Soal 2:

```
#include <iostream>
using namespace std;

struct ElemenQueue
{
    char info;
    ElemenQueue* next;
};

typedef ElemenQueue* pointer;
typedef pointer List;

struct Queue
{
    List Head;
    List Tail;
};

Queue Q;

void createQueue(Queue& Q)
{
    Q.Head = NULL;
    Q.Tail = NULL;
}

void createElement(pointer &pBaru)
{
    pBaru = new ElemenQueue;
    cout << "input Data    : "; cin >> pBaru->info;
```

```

    pBaru->info = pBaru->info;
    pBaru->next = NULL;
}

void insertQueue(Queue& Q, pointer pBaru)
{
    if (Q.Head==NULL && Q.Tail==NULL)
    {
        Q.Head = pBaru;
        Q.Tail = pBaru;
    }
    else
    {
        pointer temp = Q.Head;
        Q.Head = pBaru;
        Q.Head->next = temp;
    }
}

void deleteQueue(Queue& Q, pointer& pHapus)
{
    cout << "Delete Queue" << endl;
    if (Q.Head == NULL && Q.Tail == NULL)
    {
        pHapus = NULL;
        cout << "Antrian kosong " << endl;
    }
    else if (Q.Head == Q.Tail)
    {
        pHapus = Q.Head;
        Q.Head = NULL;
        Q.Tail = NULL;
    }
    else
    {
        pointer preTail = Q.Head;
        while (preTail->next != Q.Tail)
        {
            preTail = preTail->next;
        }

        Q.Tail = preTail;
        pHapus = Q.Tail;
        pHapus->next = NULL;
    }
}

```

```

}

void traversal(Queue Q)
{
    if (Q.Head == NULL)
    {
        cout << "\nAntrian kosong!" << endl;
    }
    else
    {
        pointer pBantu = Q.Head;
        cout << endl;

        do
        {
            cout << pBantu->info << "|\t";
            pBantu = pBantu -> next;
        }
        while (pBantu != NULL);
    }
}

char menu()
{
    int opsi;
    cout << "===== " << endl
         << "          MENU PROGRAM QUEUE          " << endl
         << "===== " << endl
         << "1. Insert Queue " << endl
         << "2. Delete Queue " << endl
         << "3. Print Queue " << endl << endl
         << "Pilihan\t: "; cin >> opsi; cin.ignore();

    return opsi;
}

int main()
{
    Queue Q;
    pointer pBaru, pHapus;
    int opsi;
    int found = 0;
    bool program = true;
    createQueue(Q);

```

```

while (program)
{
    int pil = menu();

    switch (pil)
    {
        case 1:

            createElement(pBaru);
            insertQueue(Q, pBaru);
            traversal(Q);
            break;

        case 2:
            deleteQueue(Q, pHapus);
            traversal(Q);
            break;

        case 3:
            traversal(Q);
            break;

        default:
            cout << "\nPilihan Tidak Tersedia." << endl;
            break;
    }

    cout << "\nIngin terus menggunakan program?" << endl
        << "1.YA" << endl
        << "2.TIDAK" << endl
        << "Pilihan\t: "; cin >> opsi; cin.ignore();

    if (opsi == 1)
    {
        program = true;
    }
    else if (opsi == 2)
    {
        program = false;
        cout << "\nTerima kasih!" << endl;
    }
    else
    {
        program = false;
    }
}

```

```
        cout << "\nPilihan Tidak Tersedia" << endl;
    }
}
```



```
=====
                        MENU PROGRAM QUEUE
=====
1. Insert Queue
2. Delete Queue
3. Print Queue

Pilihan : 1
input Data      : A

A|
Ingin terus menggunakan program?
1.YA
2.TIDAK
Pilihan : 1
=====
                        MENU PROGRAM QUEUE
=====
1. Insert Queue
2. Delete Queue
3. Print Queue

Pilihan : 1
input Data      : B

B|      A|
Ingin terus menggunakan program?
1.YA
2.TIDAK
Pilihan : 1
=====
                        MENU PROGRAM QUEUE
=====
1. Insert Queue
2. Delete Queue
3. Print Queue

Pilihan : 1
input Data      : C

C|      B|      A|
Ingin terus menggunakan program?
1.YA
2.TIDAK
Pilihan : 2

Terima kasih!
```

### 3. Soal 3:

```
#include <iostream>
using namespace std;

struct ElemenQueue
{
    char info;
    int prior;
    ElemenQueue* next;
    ElemenQueue* prev;
};

typedef ElemenQueue* pointer;
typedef pointer List;

struct Queue
{
    List Head;
    List Tail;
};

Queue Q;

void createQueue(Queue& Q)
{
    Q.Head = NULL;
    Q.Tail = NULL;
}

void createElement(pointer &pBaru)
{
    pBaru = new ElemenQueue;
    cout << "input Data      : "; cin >> pBaru->info;
    cout << "input Priority : "; cin >> pBaru->prior;
    pBaru->info = pBaru->info;
    pBaru->prior = pBaru->prior;
    pBaru->next = NULL;
    pBaru->prev = NULL;
}

void enqueue(Queue& Q, pointer pBaru)
```

```

{
    if (Q.Head==NULL && Q.Tail==NULL)
    {
        Q.Head = pBaru;
        Q.Tail = pBaru;
        return;
    }

    pointer temp = Q.Head;
    pointer tempPrev = NULL;

    while (temp->next != NULL && pBaru->prior >= temp->prior)
    {
        tempPrev = temp;
        temp = temp->next;
    }

    if (temp == Q.Head && pBaru->prior < temp->prior)
    {
        pBaru->next = Q.Head;
        Q.Head = pBaru;
    }
    else if (temp == Q.Tail && pBaru->prior > temp->prior)
    {
        Q.Tail->next = pBaru;
        Q.Tail = pBaru;
    }
    else
    {
        tempPrev->next = pBaru;
        pBaru->next = temp;
    }
}

char menu()
{
    int opsi;
    cout << "===== " << endl
         << "          MENU PROGRAM QUEUE          " << endl
         << "===== " << endl
         << "1. Insert Queue          " << endl
         << "2. Print Queue          " << endl << endl
         << "Pilihan\t: "; cin >> opsi; cin.ignore();

    return opsi;
}

```

```

}

void traversal(Queue Q)
{
    if (Q.Head == NULL)
    {
        cout << "\nAntrian kosong!" << endl;
    }
    else
    {
        pointer pBantu = Q.Head;
        do
        {
            cout << pBantu->info << " | " << pBantu->prior << '\n';
            pBantu = pBantu->next;
        }
        while (pBantu != NULL);
    }
}

int main()
{
    Queue Q;
    pointer pBaru;
    int opsi;
    int found = 0;
    bool program = true;
    createQueue(Q);

    while (program)
    {
        int pil = menu();

        switch (pil)
        {
            case 1:
                createElement(pBaru);
                enqueue(Q, pBaru);
                traversal(Q);
                break;

            case 2:
                traversal(Q);

```

```

        break;

    default:
        cout << "\nPilihan Tidak Tersedia." << endl;
        break;
    }

    cout << "\nIngin terus menggunakan program?" << endl
        << "1.YA" << endl
        << "2.TIDAK" << endl
        << "Pilihan\t: "; cin >> opsi; cin.ignore();

    if (opsi == 1)
    {
        program = true;
    }
    else if (opsi == 2)
    {
        program = false;
        cout << "\nTerima kasih!" << endl;
    }
    else
    {
        program = false;
        cout << "\nPilihan Tidak Tersedia" << endl;
    }
}
}

```

```
=====
                        MENU PROGRAM QUEUE
=====
1. Insert Queue
2. Print Queue

Pilihan : 1
input Data      : A
input Priority   : 10
A | 10

Ingin terus menggunakan program?
1.YA
2.TIDAK
Pilihan : 1
=====
                        MENU PROGRAM QUEUE
=====
1. Insert Queue
2. Print Queue

Pilihan : 1
input Data      : B
input Priority   : 8
B | 8
A | 10

Ingin terus menggunakan program?
1.YA
2.TIDAK
Pilihan : 1
=====
                        MENU PROGRAM QUEUE
=====
1. Insert Queue
2. Print Queue

Pilihan : 1
input Data      : A
input Priority   : 9
B | 8
A | 9
A | 10

Ingin terus menggunakan program?
1.YA
2.TIDAK
Pilihan : 2

Terima kasih!
```

