Jahangirnagar University

Institute of Information Technology

2nd Year 1st Semester B.Sc. (Honors) Final Examination-2020

Course No. # ICT - 2102
Course Title# Data Structure Lab

Examination Roll No. #	**192340**
Registration No. #	**20193650283**
Academic Session #	**2018 - 2019**

Total no of written pages in the script #

Exam Date: 6, September, 2021

Instructions:

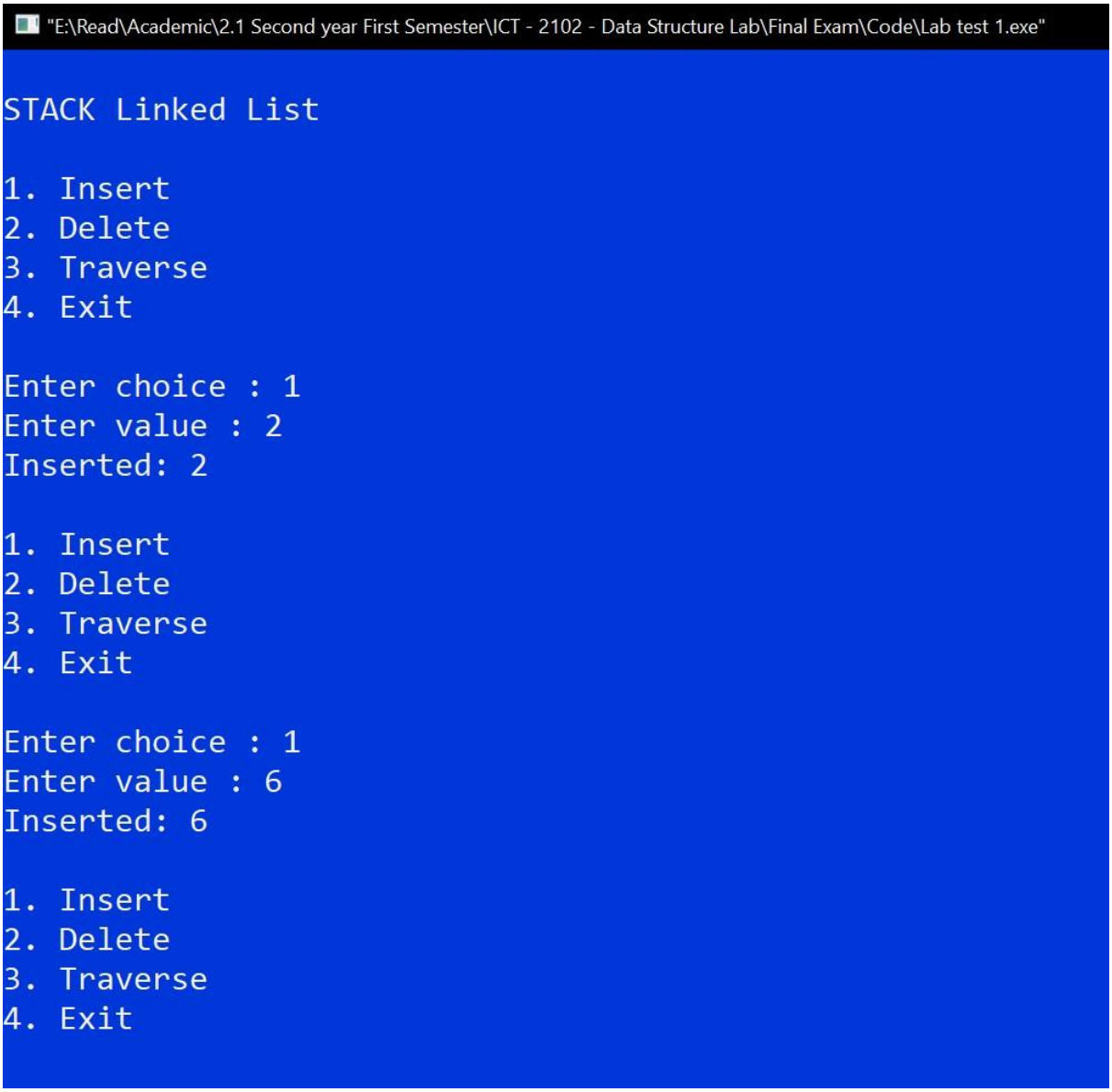
1. Examinee must write his/her exam roll no. and page no. at the top of every page of the script.
2. Do not write your name or any identification mark anywhere of the script.
3. Total time for exam is 45 minutes. You will get 15 additional minutes for submission.
4. Delay in submission is not acceptable.
5. You have to submit your exam script in PDF format.
6. The examinee must submit the examination script **through online (Google classroom/email/google form etc.)** as prescribed by the examiner.
7. You must use **your EXAM ID** only for naming your submitted file.
8. After completing the exam, you must write the total number of pages used for the exam in the top sheet.

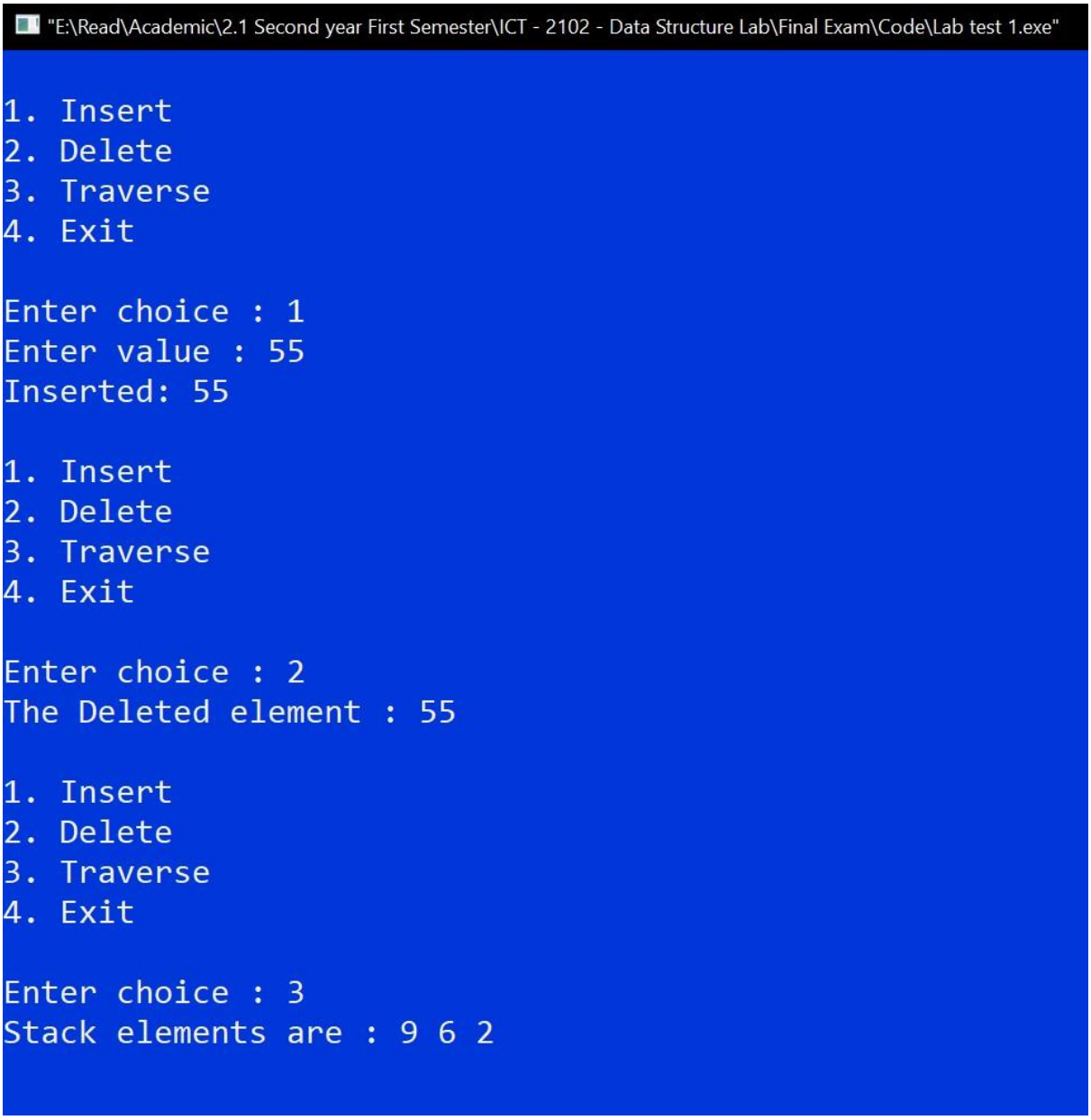
Answer to the question no 2

```
/// Md. Shakil Hossain
/// Exam Roll - 192340
/// Class Roll - 2023
/// Final Lab Exam
/// Question 2


#include<bits/stdc++.h>
using namespace std;

struct Node
{
    int data;
    struct Node *next;
};
struct Node* top = NULL;


void insert(int val)
{
    struct Node* newnode = (struct Node*) malloc(sizeof(struct Node));
    newnode->data = val;
    newnode->next = top;
    top = newnode;
}


void deleted()
{
    if(top==NULL)
        cout<<"Stack Underflow"<<endl;
    else
    {
        cout<<"The Deleted element : "<< top->data <<endl;
        top = top->next;
    }
}


void traverse()
{
    struct Node* p;
    if(top==NULL)
        cout<<"stack is empty."<<endl;
    else
    {
        p = top;
        cout<<"Stack elements are : ";
        while (p != NULL)
        {
            cout<< p->data <<" ";
            p = p->next;
```

```
        }
    }
    cout<<endl;
}


int main()
{
    int a, x;
    cout << "\n";
    cout << "STACK Linked List\n";
    while(1)
    {
        cout<<"\n1. Insert"<<endl;
        cout<<"2. Delete"<<endl;
        cout<<"3. Traverse"<<endl;
        cout<<"4. Exit"<<endl;
        cout<<"\nEnter choice : ";
        cin>>a;
        switch(a)
        {
        case 1:
        {
            cout<<"Enter value : ";
            cin>>x;
            insert(x);
            cout << "Inserted: " << x << endl ;
            break;
        }
        case 2:
        {
            deleted();
            break;
        }
        case 3:
        {
            traverse();
            break;
        }
        case 4:
        {
            cout<<"Thank you"<<endl;
            exit(0);
        }
        default:
        {
            cout<<"Invalid Choice"<<endl;
            break;
        }
        }
    }
    return 0;
}
```

Output :


"E:\Read\Academic\2.1 Second year First Semester\ICT - 2102 - Data Structure Lab\Final Exam\Code\Lab test 1.exe"
STACK Linked List
1. Insert
2. Delete
3. Traverse
4. Exit
Enter choice : 1
Enter value : 2
Inserted: 2
1. Insert
2. Delete
3. Traverse
4. Exit
Enter choice : 1
Enter value : 6
Inserted: 6
1. Insert
2. Delete
3. Traverse
4. Exit


"E:\Read\Academic\2.1 Second year First Semester\ICT - 2102 - Data Structure Lab\Final Exam\Code\Lab test 1.exe"
1. Insert
2. Delete
3. Traverse
4. Exit
Enter choice : 1
Enter value : 55
Inserted: 55
1. Insert
2. Delete
3. Traverse
4. Exit
Enter choice : 2
The Deleted element : 55
1. Insert
2. Delete
3. Traverse
4. Exit
Enter choice : 3
Stack elements are : 9 6 2

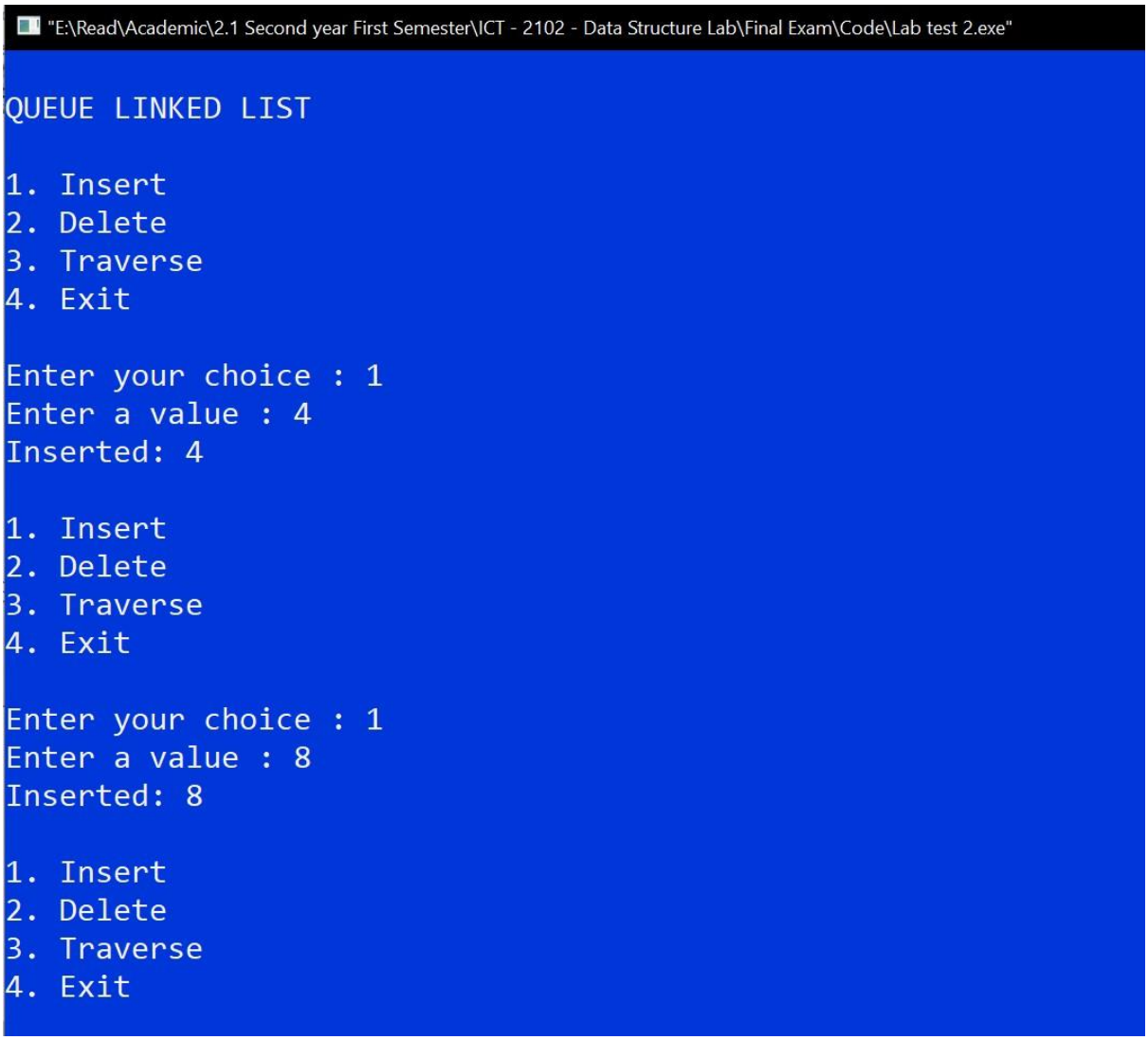
Answer to the question no 3

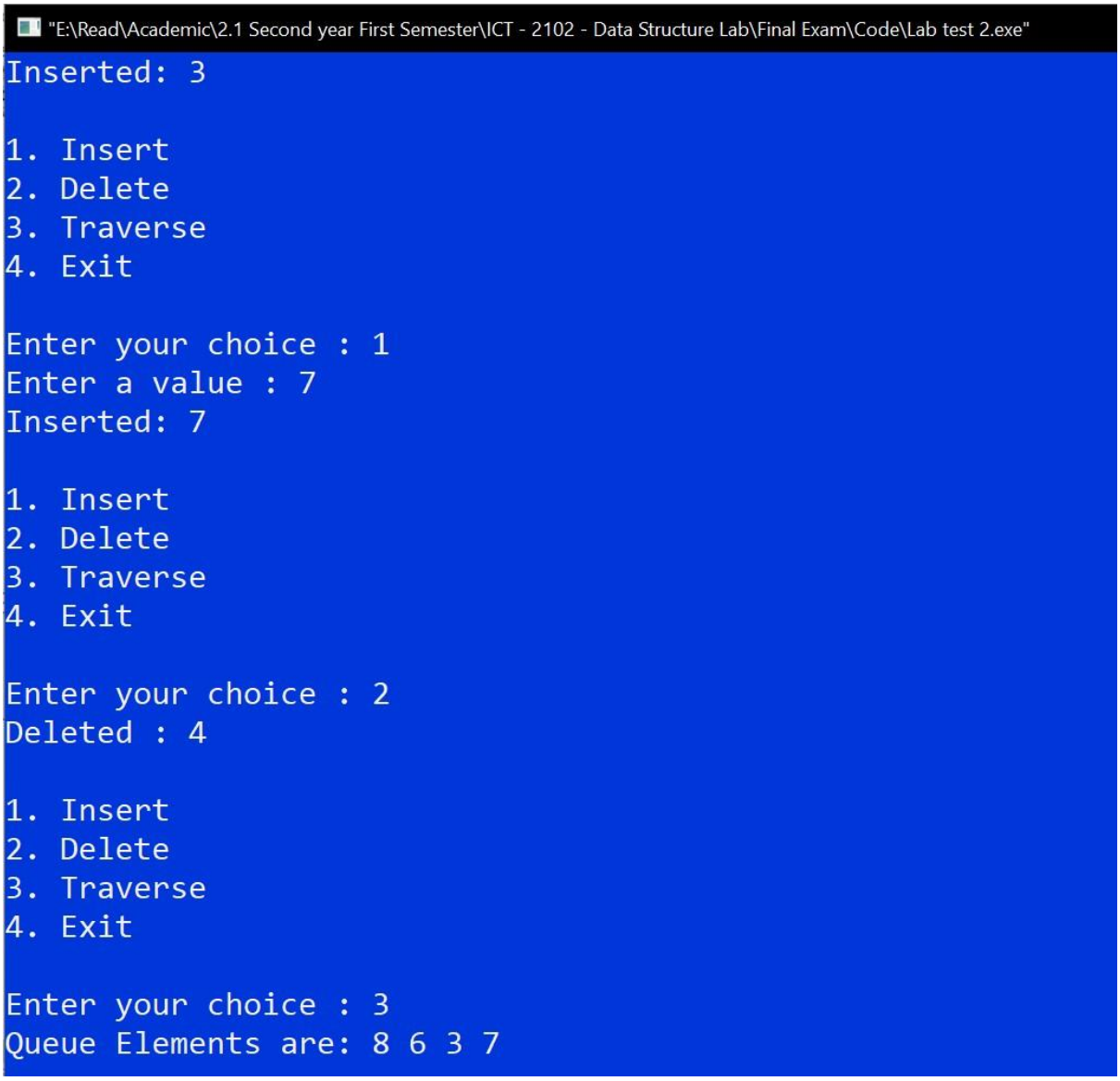
```
/// Md. Shakil Hossain
/// Exam Roll - 192340
/// Class Roll - 2023
/// Final Lab Exam
/// Question 3


#include<bits/stdc++.h>
using namespace std;

struct node
{
   int data;
   struct node *next;
};

struct node* front = NULL;
struct node* rear = NULL;
struct node* temp;


void Insert()
{
   int v;
   cout<<"Enter a value : ";
   cin>>v;
   cout << "Inserted: " << v << endl;
   if (rear == NULL)
   {
      rear = (struct node *)malloc(sizeof(struct node));
      rear->next = NULL;
      rear->data = v;
      front = rear;
   }
   else
   {
      temp=(struct node *)malloc(sizeof(struct node));
      rear->next = temp;
      temp->data = v;
      temp->next = NULL;
      rear = temp;
   }
}


void Delete()
{
   temp = front;
   if (front == NULL)
   {
      cout<<"Underflow"<<endl;
      return;
```

```
    }
    else if (temp->next != NULL)
    {
        temp = temp->next;
        cout<<"Deleted : "<<front->data<<endl;
        free(front);
        front = temp;
    }
    else
    {
        cout<<"Deleted : "<<front->data<<endl;
        free(front);
        front = NULL;
        rear = NULL;
    }
}


void traverse()
{
    temp = front;
    if ((front == NULL) && (rear == NULL))
    {
        cout<<"Queue is empty"<<endl;
        return;
    }
    cout<<"Queue Elements are: ";
    while (temp != NULL)
    {
        cout<<temp->data<<" ";
        temp = temp->next;
    }
    cout<<endl;
}


int main()
{
    int c;
    cout << "\nQUEUE LINKED LIST\n";
    while(1)
    {
        cout<<"\n1. Insert"<<endl;
        cout<<"2. Delete"<<endl;
        cout<<"3. Traverse"<<endl;
        cout<<"4. Exit"<<endl;
        cout<<endl<<"Enter your choice : ";
        cin>>c;

        switch (c)
        {
        case 1:
            Insert();
            break;
```

```
        case 2:
            Delete();
            break;
        case 3:
            traverse();
            break;
        case 4:
            cout<<"Exit"<<endl;
            exit(0);
        default:
            cout<<"Invalid choice"<<endl;
        }
    }
    return 0;
}
```

Output :


"E:\Read\Academic\2.1 Second year First Semester\ICT - 2102 - Data Structure Lab\Final Exam\Code\Lab test 2.exe"
QUEUE LINKED LIST
1. Insert
2. Delete
3. Traverse
4. Exit
Enter your choice : 1
Enter a value : 4
Inserted: 4
1. Insert
2. Delete
3. Traverse
4. Exit
Enter your choice : 1
Enter a value : 8
Inserted: 8
1. Insert
2. Delete
3. Traverse
4. Exit


"E:\Read\Academic\2.1 Second year First Semester\ICT - 2102 - Data Structure Lab\Final Exam\Code\Lab test 2.exe"
Inserted: 3
1. Insert
2. Delete
3. Traverse
4. Exit
Enter your choice : 1
Enter a value : 7
Inserted: 7
1. Insert
2. Delete
3. Traverse
4. Exit
Enter your choice : 2
Deleted : 4
1. Insert
2. Delete
3. Traverse
4. Exit
Enter your choice : 3
Queue Elements are: 8 6 3 7

Answer to the question no 4

```
/// Md. Shakil Hossain
/// Exam Roll - 192340
/// Class Roll - 2023
/// Final Lab Exam
/// Question 4


#include<bits/stdc++.h>
using namespace std;

struct bst
{
   int data;
   bst* left;
   bst* right;
};


bst* newnode(int data)
{
   bst* newnode = new bst();
   newnode->data=data;
   newnode->left=NULL;
   newnode->right=NULL;
   return newnode;
}


bst* Insert(bst* root, int data )
{
   if(root==NULL)
   {
      root= newnode(data);
   }
   else if(data<= root->data)
   {
      root->left=Insert(root->left, data);
   }
   else if( data > root->data)
   {
      root->right=Insert(root->right,data);
   }

   return root;
}


bool Search(bst* root, int data )
{
   if(root==NULL)
   {
      return false;
```

```
    }
    else if( root->data==data)
    {
        return true;
    }
    else if(data<=root->data)
    {
        return Search(root->left,data);
    }
    else if(data>root->data)
    {
        return Search(root->right,data);
    }
}


void porder(bst *temp)
{

    if(temp==NULL)
    {
        return;
    }
    cout<<temp->data<<" ";
    porder(temp->left);
    porder(temp->right);
    return;
}


int main()
{
    bst* root;
    root = NULL;
    int n,i,x;
    cout<<"Enter size of the BST :  ";
    cin>>n;
    cout<<"Enter data : ";
    for(i=1; i<=n; i++)
    {
        cin>>x;
        root= Insert(root, x);
    }

    cout<<"\nBST Pre-order"<<endl;
    porder(root);

    int num;
    cout<<"\nEnter data you search: ";
    cin>>num;
    if(Search(root,num)==true)
    {
        cout<<"Number is found"<<endl;
    }
```

```
    else
    {
        cout<<"Number is not found"<<endl;
    }
    return 0;
}
```

Output :