



আন্তর্জাতিক ইসলামী বিশ্ববিদ্যালয় চট্টগ্রাম
الجامعة الإسلامية العالمية شيتاغونغ
International Islamic University Chittagong

Assignment

Assignment No. – 03

Submission date- 31 October, 2021

Course Title- Data Structure (Theory)

Course Code: CSE-2322

Submitted to-

Mohammed Shamsul Alam

Professor, Dept. of CSE, IIUC.

Cell: 01711941680, alam_cse@yahoo.com

Submitted by-

MD. SOROWAR MAHABUB RABBY

Matric ID: **C201032**, Section: **3AM**, Semester: **3rd**

Department of CSE (Computer Science and Engineering), IIUC

Cell: 01834756433, 01521564157, c201032@ugrad.iiuc.ac.bd

Problem No. & Statement	1. Write a program to calculate the Factorial of a number using recursive and non-recursive method.
------------------------------------	------------------------------------------------------------------------------------------------------------

/*	Author: Sorowar Mahabub ID: C201032, Section: 3AM, CSE, IIUC	*/
----	-----------------------------------------------------------------	----

non-recursive method

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

int main()
{
    int i,fact=1,n;
    cin>>n;
    for(i=1; i<=n; i++)
        fact=fact*i;

    cout<<fact<<endl;
    return 0;
}
```

recursive method

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

int factorial(int n)
{
    int fact=1;
    if(n==0)
    {
        return fact;
    }
    else
    {
        return n*factorial(n-1);
    }
}
```

```

/*
                                Author: Sorowar Mahabub
                                ID: C201032, Section: 3AM, CSE, IIUC
                                */

int main()
{
    int n,value;
    cin>>n;
    value=factorial(n);
    cout<<value;
    return 0;
}

```

Problem No. & Statement	2. Write a program to find the nth term F_n of the <i>Fibonacci sequence</i> using <i>recursive</i> and <i>non-recursive</i> method.
------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------

```

/*
                                Author: Sorowar Mahabub
                                ID: C201032, Section: 3AM, CSE, IIUC
                                */

```

non-recursive method

```

#include<iostream>
using namespace std;

int main()
{
    int N,f,s,t,i;
    cin>>N;
    f=0;
    s=1;
    for(i=0; i<N; i++)
    {
        cout<<f<<" ";
        t=f+s;
        f=s;
        s=t;
    }
    return 0;
}

```

```

/*
                                Author: Sorowar Mahabub
                                ID: C201032, Section: 3AM, CSE, IIUC
                                */

```

recursive method

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

int fibbonacci(int n)
{
    if(n == 0)
        return 0;

    else if(n == 1)
        return 1;

    else
        return (fibbonacci(n-1) + fibbonacci(n-2));
}

int main()
{
    int n,i;
    cin>>n;
    for(i = 0; i<n; i++)
        cout << fibbonacci(i) << " ";

    return 0;
}
```

Problem No. & Statement	3. Write a program to move n disks for Tower of Hanoi problem.
------------------------------------	------------------------------------------------------------------------------------

```
/*
                                Author: Sorowar Mahabub
                                ID: C201032, Section: 3AM, CSE, IIUC
                                */

#include<iostream>
using namespace std;

void Tower(int n,char Beg, char Aux,char End)
{
    if(n==1)
    {
        cout<<"Move Disk "<<n<<" from "<<Beg<<" to
"<<End<<endl;
        return;
    }
}
```

```

Tower(n-1,Beg,End,Aux);
cout<<"Move Disk "<<n<<" from "<<Beg<<" to "<<End<<endl;
Tower(n-1,Aux,Beg,End);
}

/*
                                Author: Sorowar Mahabub
                                ID: C201032, Section: 3AM, CSE, IIUC                                */

int main()
{
    int n;
    cout<<"Enter no. of disks: ";
    cin>>n;
    Tower(n,'A','B','C');

    return 0;
}

```

Problem No. & Statement	4. Write a program to find the value from Ackerman function.
------------------------------------	---------------------------------------------------------------------

```

/*
                                Author: Sorowar Mahabub
                                ID: C201032, Section: 3AM, CSE, IIUC                                */

#include <iostream>
using namespace std;

int ackf(int m, int n)
{
    if (m == 0)
    {
        return n + 1;
    }
    else if((m!= 0) && (n == 0))
    {
        return ackf(m - 1, 1);
    }
    else if((m != 0) && (n!=0))
    {
        return ackf(m - 1, ackf(m, n - 1));
    }
}

/*
                                Author: Sorowar Mahabub
                                ID: C201032, Section: 3AM, CSE, IIUC                                */

```

```

int main()
{
    int m,n,A;
    cin>>m>>n;
    A = ackf(m,n);
    cout << A << endl;
    return 0;
}

```

Problem No. & Statement	5. Write a program to show the insert and delete operations of a circular queue.
------------------------------------	-----------------------------------------------------------------------------------------

```

/*
                                Author: Sorowar Mahabub
                                ID: C201032, Section: 3AM, CSE, IIUC
                                                                */

#include <iostream>
using namespace std;

int Queue[3];

int Front = -1, rear = -1, n=3;
void Insert(int a)
{
    if ((Front == 0 && rear == n-1) || (Front == rear+1))
    {
        cout<<"Queue Overflow "<<endl;
        return;
    }
    if (Front == -1)
    {
        Front = 0;
        rear = 0;
    }
    else
    {
        if (rear == n - 1)
        {
            rear = 0;
        }
        else
        {
            rear = rear + 1;
        }
    }
}

```

```
    Queue[rear] = a ;
}

void Delete()
{
    if (Front == -1)
    {
        cout<<"Queue Underflow"<<endl;
        return ;
    }
    cout<<"Element deleted from queue is : "<<Queue[Front]<<endl;
    if (Front == rear)
    {
        Front = -1;
        rear = -1;
    }
    else
    {
        if (Front == n - 1)
        {
            Front = 0;
        }
        else
        {
            Front = Front + 1;
        }
    }
}

void display()
{
    int f = Front, r = rear;
    if (Front == -1)
    {
        cout<<"Queue is empty"<<endl;
        return;
    }
    cout<<"Queue elements are : "<<endl;
    if (f <= r)
    {
        while (f <= r)
        {
            cout<<Queue[f]<<" ";
            f++;
        }
    }
    else
    {

```

```
        while (f <= n - 1)
        {
            cout<<Queue[f]<<" ";
            f++;
        }
        f = 0;
        while (f <= r)
        {
            cout<<Queue[f]<<" ";
            f++;
        }
    }
    cout<<endl;
}
int main()
{
    int ch, a;
    cout<<"1) Insert"<<endl;
    cout<<"2) Delete"<<endl;
    cout<<"3) Display"<<endl;
    cout<<"4) Exit"<<endl;
    do
    {
        cout<<"Enter choice : "<<endl;
        cin>>ch;
        switch(ch)
        {
            case 1:
                cout<<"Input for insertion: "<<endl;
                cin>>a;
                Insert(a);
                break;
            case 2:
                Delete();
                break;
            case 3:
                display();
                break;
            case 4:
                cout<<"Exit\n";
                break;
            default:
                cout<<"Incorrect!\n";
        }
    }
    while(ch != 4);
    return 0;
}
```


**Problem No. &
Statement**

6. Write a program to show the insert and delete operations of a priority queue using linked-list.

```
/*                                     Author: Sorowar Mahabub                                     */
                                     ID: C201032, Section: 3AM, CSE, IIUC                                     */

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

#define NULL 0

struct node
{
    int priority;
    int info;
    node *link;
};

node *Front = NULL;

void display();

void Insert(int item,int priority)
{
    node *temp, *q;

    temp = new node();
    temp->info = item;
    temp->priority = priority;
}
```

```
/*                                     Author: Sorowar Mahabub                                     */
                                     ID: C201032, Section: 3AM, CSE, IIUC                                     */
```

```

    if( Front == NULL || priority < Front->priority )
    {
        temp->link = Front;
        Front = temp;
    }

    else
    {
        q = Front;

        while( q->link != NULL && q->link->priority <= priority )
        {
            q=q->link;
        }
        temp->link = q->link;
        q->link = temp;
    }
    display();
}

```

```

void Delete()
{
    node *temp;
    if(Front == NULL)
        cout<<"Queue Underflow\n";

    else
    {
        temp = Front;
        cout<<"Deleted item is "<<temp->info<<endl;
        Front = Front->link;
    }
}

```

```

void display()
{
    node *ptr;
    ptr = Front;
    if(Front == NULL)
        cout<<"\nQueue is empty\n";

    else
    {
        cout<<"\nQueue Elements  :";
        while(ptr != NULL)
        {
            cout<<ptr->info<<" ("<<ptr->priority<<") ";

```

```

        ptr = ptr->link;
    }
}

int main()
{
    int choice,item,priority;
    do
    {
        cout<<"\n1.Insert\n2.Delete\n3.Display\n4.Quit\n";
        printf("Enter your choice : ");
        scanf("%d", &choice);

        switch(choice)
        {
            case 1:
                cout<<"Input the item value  : ";
                cin>>item;
                cout<<"Enter its priority : ";
                cin>>priority;
                Insert(item,priority);
                break;

            case 2:
                Delete();
                break;

            case 3:
                display();
                break;

            case 4:
                break;

            default :
                cout<<"Wrong choice\n";

        }

    }while(choice!=4);
    return 0;
}

```

/*

*/

Problem No. & Statement	7. Write a program to show the insert and delete operations of a priority queue using array..
------------------------------------	------------------------------------------------------------------------------------------------------

```

/*                                     Author: Sorowar Mahabub
                                     ID: C201032, Section: 3AM, CSE, IIUC                                     */

#include<bits/stdc++.h>
using namespace std;
#define N 10
int A[N+1][N+1], Front[N+1], Rear [N+1];

void display();

//-----Insert
void QueueInsert()
{
    int p, ITEM;
    cout<<"Enter the priority Num: ";
    cin>>p;

    if((Front[p] == 1 && Rear[p] == N ) || (Front[p] ==
Rear[p]+1))
    {
        cout<<"Overflow"<<endl;
        return;
    }

    cout<<"Enter the element to insert in Queue["<<p<<"] :
";
    cin>>ITEM;

    if(Front[p] == 0)
    {
        Front[p]=1;
        Rear[p]=1;
    }
    else if(Rear[p]==N)
    {
        Rear[p]=1;
    }
    else
    {
        Rear[p]=Rear[p]+1;
    }
    A[p][Rear[p]]=ITEM;

```

```
        display();
    }
    //-----Delete
    void QueueDelete()
    {
        int p;
        for(int i=1; i<=N; i++)
        {
            if(Front[i]==0)
                continue;
            else
            {
                p=i;
                break;
            }
        }

        if(Front[p]==0)
        {
            cout<<"Underflow"<<endl;
            return;
        }

        cout<<"\nDeleted Item : "<<A[p][Front[p]]<<endl;

        if(Front[p] == Rear[p])
        {
            Front[p]=0;
            Rear[p]=0;
        }
        else if(Front[p] == N)
        {
            Front[p]=1;
        }
        else
            Front[p] = Front[p]+1;

        display();
    }
    //-----Display
    void display()
    {
        int f,r;
        for(int i=1; i<=N; i++)
        {
            if(Front[i]!=0)
```

```

        {
            f=Front[i],r=Rear[i];
            if (f == 0)
            {
                cout<<"Queue["<<i<<" is empty"<<endl;
                return;
            }
            if(f<=r)
            {
                cout<<"\nElements in Queue of Priority
"<<i<<" are: ";

                while(f<=r)
                {
                    cout<<A[i][f]<<" ";
                    f++;
                }
            }
            else
            {
                cout<<"\nElements in Queue of Priority
"<<i<<" are: ";
                while(f<=N)
                {
                    cout<<A[i][f]<<" ";
                    f++;
                }
                f=1;
                while(f<=r)
                {
                    cout<<A[i][f]<<" ";
                    f++;
                }
            }
        }
    }
    return;
}

int main()
{
    int choice;

    do
    {
        cout<<"\n1) Insert\n2) Delete\n0) Exit :
"<<endl<<"Enter your choice: ";
        cin>>choice;
    }

```

```

        switch(choice)
        {
            case 1:
                QueueInsert();
                break;
            case 2:
                QueueDelete();
                break;
            case 0:
                printf("End of operation\n");
                break;
        }
    }
    while(choice!=0);
    return 0;
}

```

Problem No. & Statement	8. Write a program to create a Linked List of n elements and then display the list.
------------------------------------	---------------------------------------------------------------------------------------------------------

```

/*
                                Author: Sorowar Mahabub
                                ID: C201032, Section: 3AM, CSE, IIUC
                                                                */

#include<stdio.h>
#include<stdlib.h>
#define NULL 0
struct linked_list
{
    int info;
    struct linked_list *link;
};

typedef struct linked_list node;
int main()
{
    int n,i,item;
    node *start,*ptr;
    start=(node*)malloc(sizeof(node));
    ptr=start;
    printf("How many elements: ");
    scanf("%d",&n);
    printf("Enter the number: ");
    for(i=1; i<=n; i++)

```

```

{
    scanf("%d",&ptr->info);
    if(i!=n)
    {
        ptr->link=(node*)malloc(sizeof(node));
        ptr=ptr->link;
    }
}
ptr->link=NULL;
printf("\nElements in the Link list are: \n");
ptr=start;
while(ptr!=NULL)
{
    printf("%d\n",ptr->info);
    ptr= ptr->link;
}
}

```

```

/*
                                Author: Sorowar Mahabub
                                ID: C201032, Section: 3AM, CSE, IIUC
                                                                */

```

Problem No. & Statement	9. Write a program to create a Linked List of n elements and then search an element from the list.
------------------------------------	------------------------------------------------------------------------------------------------------------------------

```

/*
                                Author: Sorowar Mahabub
                                ID: C201032, Section: 3AM, CSE, IIUC
                                                                */

```

```

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

struct linked_list
{
    int num;
    struct linked_list *next;
};

typedef struct linked_list node;

```



```

int main()
{
    int n,i,item;
    node *start, *ptr;

    start = (node *) malloc(sizeof(node));
    ptr=start;

    printf("How many elements: ");
    scanf("%d",&n);

    for(i=1; i<=n; i++)
    {
        printf("input a number: ");
        scanf("%d",&ptr->num);
        if(i!=n)
        {
            ptr->next=(node *)malloc(sizeof(node));
            ptr=ptr->next;
        }
    }

    ptr->next=NULL;
    int count= 0;
    printf("\nElements in the Link list are: \n");
    ptr=start;
    while(ptr!=NULL)
    {

        printf("%d ",ptr->num);
        ptr= ptr->next;
    }

    ptr= start;

    cout << endl << "Enter The Searching Item : ";
    cin >> item;

    int loc= 0;
    while(ptr!=NULL)
    {

        count++;

        if(item==ptr->num)
        {
            loc= count;

```

```

        break;
    }

    ptr= ptr->next;
}

if(loc==0)
    cout << "Item is not found here!" << endl;
else
    cout << loc << " is the position of the searching
Item " << item << endl;

return 0;
}

```

Problem No. & Statement	10. Write a program to create a Linked List of n elements and then insert an element to the list.
------------------------------------	-----------------------------------------------------------------------------------------------------------------------

```

/*                                     Author: Sorowar Mahabub
                                     ID: C201032, Section: 3AM, CSE, IIUC                                     */

#include<bits/stdc++.h>
using namespace std;
#define NULL 0
struct Node
{
    int Info;
    struct Node *Link;
};

Node *Start, *Curr, *Prev, *Loc, *New;

//-----Function Prototypes
void Display ();
void FindLoc(int Item);
void InsertLoc(int Item);

//-----Create List
void CREATE()
{
    int item,num,N,i;
    Node *Location;

```

```

    cout<<"How Many Numbers :";
    cin>>N;
    i=1;

    cout<<"Enter the elements: ";

    while(i<=N)
    {
        cin>>num;

        FindLoc(num);
        InsertLoc(num);
        i++;

    }

    Display();
}
//-----Find Location
void FindLoc(int Item)
{
    if (Start == NULL)
    {
        Loc = NULL;
        return ;
    }
    if(Item < Start->Info)
    {
        Loc = NULL;
        return ;
    }
    Prev = Start;
    Curr = Start->Link;

    while(Curr!=NULL)
    {
        if(Item < Curr->Info)
        {
            Loc = Prev;
            return ;
        }
        Prev = Curr;
        Curr = Curr->Link;
    }
    Loc = Prev;

    return ;
}

```

```

}

/*                                     Author: Sorowar Mahabub
                                     ID: C201032, Section: 3AM, CSE, IIUC                                     */

//-----Insert
void InsertLoc(int Item)
{
    New = new Node();
    New->Info = Item;

    if(Loc == NULL)
    {
        New->Link= Start;
        Start = New;
    }
    else
    {
        New->Link = Loc->Link;
        Loc->Link = New;
    }
}

//-----Display
void Display ()
{
    Node *ptr;
    cout<<"\nElements in the Link list are(sorted): \n";
    ptr=Start;
    while(ptr!=NULL)
    {
        cout<<ptr->Info<<" ";
        ptr= ptr->Link;
    }
    cout<<endl;
}

int main()
{
    int item;

    CREATE();

    cout<<"\n\nEnter a number to Insert: ";
    cin>>item;

    FindLoc(item);
    InsertLoc(item);
}

```

```

    Display();

    return 0;
}

```

```

/*
                                Author: Sorowar Mahabub
                                ID: C201032, Section: 3AM, CSE, IIUC
                                                                */

```

Problem No. & Statement	11. Write a program to create a Linked List of n elements and then delete an element from the list.
------------------------------------	-------------------------------------------------------------------------------------------------------------------------

```

/*
                                Author: Sorowar Mahabub
                                ID: C201032, Section: 3AM, CSE, IIUC
                                                                */

#include<bits/stdc++.h>
using namespace std;
#define NULL 0
struct Node
{
    int Info;
    struct Node *Link;
};
Node *Start, *Prev, *Curr, *Loc, *LocPrev;

void Display ();

//-----Create
void CREATE()
{
    int N;
    Node *ptr;
    Start = new Node();
    ptr=Start;

    cout<<"How many elements: ";
    cin>>N;

    for(int i=1; i<=N; i++)
    {
        printf("input a number: ");
        cin>>ptr->Info;
        if(i!=N)
        {

```

```

        ptr->Link= new Node();
        ptr=ptr->Link;
    }
}
ptr->Link=NULL;

Display();
}
//-----Find Location
void FindLoc(int Item)
{
    if (Start == NULL)
    {
        Loc=NULL;
        LocPrev= NULL;
        return;
    }
    if(Start->Info == Item)
    {
        Loc=Start;
        LocPrev=NULL;
        return;
    }
    Prev = Start;
    Curr = Start->Link;

    while(Curr!=NULL)
    {
        if(Curr->Info == Item)
        {
            Loc=Curr;
            LocPrev=Prev;
            return;
        }
        Prev = Curr;
        Curr = Curr->Link;
    }

    Loc=NULL;
}

//-----Delete
void Delete()
{
    if(Loc==NULL)
    {
        cout<<"Item is not in the List"<<endl;
        return;
    }
}

```

```

    }
    if (LocPrev==NULL)
    {
        Start=Start->Link;
    }
    else
    {
        LocPrev->Link=Loc->Link;
    }
    return;
}

/*                                     Author: Sorowar Mahabub                                     */
/*                                     ID: C201032, Section: 3AM, CSE, IIUC                                     */

//-----Display
void Display ()
{
    Node *ptr;
    cout<<"\nElements in the Link list are: \n";
    ptr=Start;
    while(ptr!=NULL)
    {
        cout<<ptr->Info<<" ";
        ptr= ptr->Link;
    }
    cout<<endl;
}

int main()
{
    int item;

    CREATE();

    cout<<"Enter a number to Delete: ";
    cin>>item;

    FindLoc(item);
    Delete();
    Display();

    return 0;
}

/*                                     Author: Sorowar Mahabub                                     */
/*                                     ID: C201032, Section: 3AM, CSE, IIUC                                     */

```

Problem No. & Statement	12. Write a program to create a Circular Header Linked List of n elements and then display the list.
------------------------------------	-------------------------------------------------------------------------------------------------------------

```

/*                                     Author: Sorowar Mahabub
                                     ID: C201032, Section: 3AM, CSE, IIUC                                     */

#include<bits/stdc++.h>
using namespace std;
#define NULL 0
struct node
{
    int data;
    node *next;
};

node *head = NULL;
node *tail = NULL;
node *newNode;

void Insert(int data)
{
    newNode = new node();
    if(head == NULL)
    {
        head = newNode;
        tail = newNode;
        newNode->next = head;
    }
    else
    {
        tail->next = newNode;
        tail = newNode;
        tail->next = head;
    }
    newNode->data = data;
}

void display()
{
    node *curr = head;
    if(head == NULL)
    {
        cout<<"List is empty"<<endl;
    }
}

```



```

else
{
    cout<<"Elements of the Circular linked list: \n";
    do
    {
        cout<<curr->data<<" ";
        curr = curr->next;
    }
    while(curr != head);
    cout<<endl;
}
}

int main()
{
    int item,n;
    cout<<"How Many Elements: ";
    cin>>n;

    cout<<"Enter the elements: "<<endl;
    for(int i=1; i<=n; i++)
    {
        cin>>item;
        Insert(item);
    }
    display();
    return 0;
}

/*                                     Author: Sorowar Mahabub
                                     ID: C201032, Section: 3AM, CSE, IIUC                                     */

```

**Problem No. &
Statement**

13. Write a program to create a Two way Linked List of n elements and then display the list.

```

/*                                     Author: Sorowar Mahabub
                                     ID: C201032, Section: 3AM, CSE, IIUC                                     */

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

struct Node
{
    int data;

```

```

    struct Node *prev;
    struct Node *next;
};
Node* head = NULL;

void Insert(int newdata)
{
    Node* newnode = new Node();
    newnode->data = newdata;
    newnode->prev = NULL;
    newnode->next = head;
    if(head != NULL)
        head->prev = newnode ;
    head = newnode;
}

void display()
{
    struct Node* ptr;
    ptr = head;
    while(ptr != NULL)
    {
        cout<< ptr->data <<" ";
        ptr = ptr->next;
    }
}

int main()
{
    int item,N;
    cout<<"How many Elements: ";
    cin>>N;
    cout<<"Enter elements: "<<endl;
    for (int i=1; i<=N; i++)
    {
        cin>>item;
        Insert(item);
    }
    cout<<"The doubly linked list is: ";
    display();
    return 0;
}

```

/*

*/

Problem No. & Statement	14. Write a program to find the 100!.
<pre> /* Author: Sorowar Mahabub ID: C201032, Section: 3AM, CSE, IIUC */ num = int(input("Enter any number :")) def cal_factorial(num): factorial = 1 if num == 0 or num == 1: return 1 for i in range(1, num+1): factorial = factorial * i return factorial output = cal_factorial(num) print('Factorial of number ', num , ' is : ', output) </pre>	

Problem No. & Statement	15. Write a program to determine the value of the n th Fibonacci number F_n where $F_n = F_{n-1} + F_{n-2}$ and $F_1 = F_2 = 1$ and $n \leq 500$.
<pre> /* Author: Sorowar Mahabub ID: C201032, Section: 3AM, CSE, IIUC */ num = int(input("Enter any number :")) n1, n2 = 0, 1 sum , i=0,0 if num <= 0: print("Please enter number greater than 0") else: while(i<=num): print(sum, end=" ") n1 = n2 n2 = sum sum = n1 + n2 i+=1 </pre>	

Submitted by-

MD. SOROWAR MAHABUB RABBY

Matric ID: **C201032**, Section: **3AM**

Department of CSE (Computer Science and Engineering)