DSA LAB – 5

Name: Etcherla Sai Manoj **Mis. No:** 112015044 **Branch:** CSE

Question 1:

Code:

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

struct employee
{
   int emp_id;
   string emp_name;
   struct employee *next;
};

class program{
   employee *head = NULL, *temp = NULL, *sample = NULL, *head1 = NULL, *head2 = NULL, *temp1 = NULL, *temp2 =
NULL;
   int id, counter, i;
   string name;
   public:
      employee *create();
      void insert();
      void remove();
      void list_display();
      void count();
      void reverse();
      void rev(employee *p);
      void concat();
};

employee *program::create(){
   employee *p = new(struct employee);

   cout << "Enter I.D. : ";
   cin >> id;
   cout << "Enter name : ";
   cin >> name;
   p->emp_id = id;
   p->emp_name = name;
   p->next = NULL;
   return p;
}

void program::insert(){
   employee *p = create();

   if(head == NULL){
      head = p;
   }
   else{
      temp = head;
      while(temp->next != NULL){
         temp = temp->next;
      }
      temp->next = p;
   }
```

```
}

void program::remove(){
   int sid, flag = 0;
   cout << "Enter I.D. of employee whose data you want to delete : ";
   cin >> sid;
   temp = head;

   while(temp->next != NULL){
      if(temp->emp_id == sid){
         sample->next = temp->next;
         delete(temp);
         flag = 1;
      }
      sample = temp;
      temp = temp->next;
   }
   if(flag == 0){
      cout << "Sorry user not found!!! Check again!!!\n";
   }
}
void program::list_display(){
   if(head == NULL){
      cout << "List is empty\n";
   }
   else{
      temp = head;
      cout <<"------------------\n";
      cout << "I.D.No \t Name\n";
      cout <<"------------------\n";
      while(temp->next != NULL){
         cout << temp->emp_id << "\t" << temp->emp_name << "\n";
         temp = temp->next;
      }
      cout << temp->emp_id << "\t" << temp->emp_name << "\n";
   }
}

void program::count(){
   temp = head;
   counter = 0;
   while(temp->next != NULL){
      temp = temp->next;
      counter++;
   }
   counter++;
   cout << "Total number of employees : " << counter << endl;
}

void program::reverse(){
   cout <<"------------------\n";
   cout << "I.D.No \t Name\n";
   cout <<"------------------\n";
   rev(head);
}

void program::rev(employee *temp){
   if(temp == NULL){
      return;
   }
   else{
      rev(temp->next);
```

```
    }
    cout << temp->emp_id << "\t" << temp->emp_name << "\n";
}

void program::concat(){
    int k,j;
    cout<<"Enter no. of members in list1 : ";
    cin>>k;
    head=NULL;
    for(i=0;i<k;i++){
        insert();
        head1=head;
    }
    head=NULL;
    cout<<"Enter no. of members in list2 : ";
    cin>>j;
    for(i=0;i<j;i++){
        insert();
        head2=head;
    }
    head=NULL;
    temp1=head1;

    while(temp1->next!=NULL){
        temp1=temp1->next;
    }

    temp1->next=head2;
    temp2=head1;
    cout << "--------------------\n";
    cout << "I.D.No.\t\tNAME\n";
    cout << "--------------------\n";

    while(temp2->next!=NULL){
        cout<<"\n"<<temp2->emp_id<<"\t\t"<<temp2->emp_name;
        temp2=temp2->next;
    }
    cout<<"\n"<<temp2->emp_id<<"\t\t"<<temp2->emp_name;
    cout << "\n";
}

int main(){
    program p;
    int choice;

    cout << "***********Menu Options***********" << endl;
    cout << "1. COUNT number of employees" << endl;
    cout << "2. DISPLAY the list" << endl;
    cout << "3. INSERT a node into the list" << endl;
    cout << "4. DELETE a node from the list" << endl;
    cout << "5. DISPALY list in reverse order" << endl;
    cout << "6. CONCATENATE two lists" << endl;
    cout << "7. EXIT" << endl;
    cout << "--------------------------------" << endl;

    cout << "Enter your choice : ";
    cin >> choice;
    do{
        switch (choice)
        {
        case 1:
```

```
            p.count();
            break;
        case 2:
            p.list_display();
            break;
        case 3:
            p.insert();
            break;
        case 4:
            p.remove();
            break;
        case 5:
            p.reverse();
            break;
        case 6:
            p.concat();
            break;
        case 7:
            return 0;
            break;
        default:
            cout << "Enter a valid option !!!" << endl;
            break;
        }

        cout << "\nEnter your choice : ";
        cin >> choice;

    }while(choice != 7);


    return 0;
}
```

Input & Output:

Counting number of employees:

```
PS C:\Users\DELL\OneDrive\Desktop\Labs> cd "c:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5\" ; if ($?) { g++ employee.cpp -o employee } ; if ($?) { .\employ
ee }
**********Menu Options**********
1. COUNT number of employees
2. DISPLAY the list
3. INSERT a node into the list
4. DELETE a node from the list
5. DISPALY list in reverse order
6. CONCATENATE two lists
7. EXIT
---------------------------------
Enter your choice : 3
Enter I.D. : 110
Enter name : Tony

Enter your choice : 3
Enter I.D. : 120
Enter name : Steve

Enter your choice : 3
Enter I.D. : 130
Enter name : Bruce

Enter your choice : 3
Enter I.D. : 140
Enter name : Thor

Enter your choice : 1
Total number of employees : 4

Enter your choice : 7
PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5>
```

Displaying the list:

```
PS C:\Users\DELL\OneDrive\Desktop\Labs> cd "c:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5\" ; if ($?) { g++ employee.cpp -o employee } ; if ($?) { .\employ
ee }
**********Menu Options**********
1. COUNT number of employees
2. DISPLAY the list
3. INSERT a node into the list
4. DELETE a node from the list
5. DISPALY list in reverse order
6. CONCATENATE two lists
7. EXIT
---------------------------------
Enter your choice : 3
Enter I.D. : 110
Enter name : Tony

Enter your choice : 3
Enter I.D. : 120
Enter name : Steve

Enter your choice : 3
Enter I.D. : 130
Enter name : Bruce

Enter your choice : 3
Enter I.D. : 140
Enter name : Thor

Enter your choice : 2
-----------------
I.D.No   Name
-----------------
110     Tony
120     Steve
130     Bruce
140     Thor

Enter your choice : 7
PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5>
```

Deleting a node:

```
PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5> cd "c:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5\" ; if ($?) { g++ employee.cpp -o employee } ; if (
$?) { .\employee }
**********Menu Options**********
1. COUNT number of employees
2. DISPLAY the list
3. INSERT a node into the list
4. DELETE a node from the list
5. DISPALY list in reverse order
6. CONCATENATE two lists
7. EXIT
---------------------------------
Enter your choice : 3
Enter I.D. : 110
Enter name : Tony

Enter your choice : 3
Enter I.D. : 120
Enter name : Steve

Enter your choice : 3
Enter I.D. : 130
Enter name : Bruce

Enter your choice : 3
Enter I.D. : 140
Enter name : Thor

Enter your choice : 4
Enter I.D. of employee whose data you want to delete : 120

Enter your choice : 2
-----------------
I.D.No   Name
-----------------
110     Tony
130     Bruce
140     Thor

Enter your choice : 7
PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5>
```

Display list in reverse order:

```
PS C:\Users\DELL\OneDrive\Desktop\Labs> cd "c:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5\" ; if ($?) { g++ employee.cpp -o employee } ; if ($?) { .\employ
ee }
**********Menu Options**********
1. COUNT number of employees
2. DISPLAY the list
3. INSERT a node into the list
4. DELETE a node from the list
5. DISPALY list in reverse order
6. CONCATENATE two lists
7. EXIT
---------------------------------
Enter your choice : 3
Enter I.D. : 110
Enter name : Tony

Enter your choice : 3
Enter I.D. : 120
Enter name : Steve

Enter your choice : 3
Enter I.D. : 130
Enter name : Bruce

Enter your choice : 3
Enter I.D. : 140
Enter name : Thor

Enter your choice : 5
-----------------
I.D.No   Name
-----------------
140     Thor
130     Bruce
120     Steve
110     Tony

Enter your choice : 7
PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5>
```

Concatenation of two lists:

```
PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5> cd "c:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5\" ; if ($?) { g++ employee.cpp -o employee } ; if (
$?) { .\employee }
**********Menu Options**********
1. COUNT number of employees
2. DISPLAY the list
3. INSERT a node into the list
4. DELETE a node from the list
5. DISPALY list in reverse order
6. CONCATENATE two lists
7. EXIT
---------------------------------
Enter your choice : 6
Enter no. of members in list1 : 2
Enter I.D. : 110
Enter name : Tony
Enter I.D. : 120
Enter name : Steve
Enter no. of members in list2 : 3
Enter I.D. : 130
Enter name : Bruce
Enter I.D. : 140
Enter name : Thor
Enter I.D. : 150
Enter name : Natasha
--------------------
I.D.No.         NAME
--------------------

110             Tony
120             Steve
130             Bruce
140             Thor
150             Natasha

Enter your choice : 7
PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5>
```

Question 2:

Code:

```
#include<iostream>
using namespace std;
struct node
{  int x;
   node *next,*prev;
};
class binary
{    node *head=NULL,*temp=NULL, *head1=NULL,*temp1=NULL,
*head2=NULL,*temp2=NULL,*temp3=NULL,*head3=NULL;
     int c,i;
      public:
       node *create();
       void insert();
       void binary1();
       void binary2();
       void add();
       void com();
       void comp();
       void display();
};

node *binary::create()
{   node *p=new(struct node);
     cout<<"Enter binary digit : ";
     cin>>c;
     p->x=c;
     p->next=NULL;
     p->prev=NULL;
     return  p;
}
void binary::insert()
{
    node *p=create();

     if(head==NULL)
     {    head=p;
     }
     else
     {     temp=head;
          while(temp->next!=NULL)
          {    temp=temp->next;   }
             temp->next=p;
             p->prev=temp;

      }

}
void binary:: binary1(){
    int a;
    cout<<"Enter the no. of bits : ";
    cin>>a;
    cout << "---Enter binary number digit wise---\n";
    head=NULL;
    for(i=0;i<a;i++){
       insert();
    }
    head1=head;
```

```
    display();
    head=NULL;
    temp1=head1;
}
void binary:: binary2(){
    int a;
    cout<<"Enter the no. of bits : ";
    cin>>a;
    cout << "---Enter binary number digit wise---\n";
    head=NULL;
    for(i=0;i<a;i++){
        insert();
    }
    head2=head;
    display();
    head=NULL;
}

void binary::add(){
    int carry=0;
    temp1=head1;
    cout << "Addition of two binary numbers : ";
    while(temp1->next!=NULL){
        temp1=temp1->next;
    }
    temp2=head2;
    while(temp2->next!=NULL){
        temp2=temp2->next;
    }
    while(temp1!=NULL)
    {
        node *p=new(struct node);
        p->next=NULL;
        p->prev=NULL;
        if(temp1->x==0&&temp2->x==0){
            p->x=0+carry;  carry=0;
        }
        if(temp1->x==0&&temp2->x==1)
        {
            if(carry==0){
                p->x=1;carry=0;
            }
            else{
                p->x=0;
                carry=1;
            }
        }
        if(temp1->x==1&&temp2->x==0)
        {
            if(carry==0){
                p->x=1;carry=0;
            }
            else{
                p->x=0;  carry=1;
            }
        }
        if(temp1->x==1&&temp2->x==1)
        {
            if(carry==0){
                p->x=0;     carry=1;
            }
            else{
```

```
            p->x=1;  carry=1;
          }
       }
       if(temp3==NULL){
         temp3=p;
       }
       else
       {
          p->next=temp3;
          temp3=p;
       }

       temp1=temp1->prev;
       temp2=temp2->prev;
    }
    node *p=new(struct node);
       p->x=carry;
       p->next=NULL;
       p->prev=NULL;
       if(temp3==NULL)
            {  temp3=p;    }
            else
            {
               p->next=temp3;
               temp3=p;
            }
            head3=temp3;
      temp3=head3;
      cout<<"\n";
      while(temp3->next!=NULL){
         cout<<" "<<temp3->x;;
         temp3=temp3->next;
      }
      cout<<" "<<temp3->x<<"\n";
    }
    void binary::com(){
      while(temp1!=NULL)
      {
         node *p=new(struct node);
         p->next=NULL;
         p->prev=NULL;
         if(temp1->x==0)
            p->x=1;
      else
         p->x=0;

      if(head==NULL)
      {   head=p;
      }
      else
      {     temp=head;
         while(temp->next!=NULL)
         {   temp=temp->next;   }
         temp->next=p;
         p->prev=temp;

      }
      temp1=temp1->next;
    }  cout<<"\n1's compliment of  binary no. : ";
      display();

    int f=0;
```

```
while(temp!=NULL)
{   if(temp->x==1)
    {   temp->x=0;   }
    else
    {   temp->x=1;   f=1;
        break;
    } temp=temp->prev;
}
if(f==0)
{  node *p=new(struct node);
    p->next=NULL;
    p->prev=NULL;
    p->x=1;
    temp=head;
    head=p;
    head->next=temp;
    temp->prev=head;
    }
cout<<"\n2's compliment of  binary no. : ";
    display();
}
void binary::comp()
{

    cout<<"\nFor first binary no.";
        temp1=head1;
        com();
        head=NULL;
    cout<<"\nFor second binary no. ";
        temp1=head2;
        com();
}




void binary::display(){
    temp=head;  cout<<"\n";
    while(temp->next!=NULL){
        cout<<" "<<temp->x;;
        temp=temp->next;
    }
    cout<<" "<<temp->x<<"\n";
}

int main()
{  binary b;
    int choice;
    cout << "\n***********Menu Options***********";
    cout << "\n 1. Insert binary1";
    cout << "\n 2. Insert binary 2";
    cout << "\n 3. Add binary no.s";
    cout << "\n 4. 1's and 2's compliment of binary no.";
    cout << "\n 5. Exit";
    cout << "\n---------------------------------" << endl;
    cout << "\nEnter your choice : ";
    cin>>choice;
    do{
        switch(choice)
        {   case 1:
                b.binary1();
```

```
            break;
        case 2:
            b.binary2();
            break;
        case 3:
            b.add();
            break;
        case 4:
            b.comp();
            break;
        case 5:
            return 0;
            break;
        default:
            cout<<"Enter valid Choice";
            break;
    }
    cout << "\nEnter your choice : ";
    cin>>choice;
}while(choice != 5);
    return 0;
}
```

Input & Output:

Addition:

```
PS C:\Users\DELL\OneDrive\Desktop\Labs> cd "c:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5\" ; if ($?) { g++ binaty.cpp -o binaty } ; if ($?) { .\binaty }

***********Menu Options***********
 1. Insert binary1
 2. Insert binary 2
 4. 1's and 2's compliment of binary no.
 3. Add binary no.s
 5. Exit
--------------------------------

Enter your choice : 1
Enter the no. of bits : 4
---Enter binary number digit wise---
Enter binary digit : 1
Enter binary digit : 0
Enter binary digit : 1
Enter binary digit : 1

 1 0 1 1

Enter your choice : 2
Enter the no. of bits : 4
---Enter binary number digit wise---
Enter binary digit : 1
Enter binary digit : 0
Enter binary digit : 1
Enter binary digit : 0

 1 0 1 0

Enter your choice : 3
Addition of two binary numbers :
 1 0 1 0 1

Enter your choice : 5
PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5>
```

1’s and 2’s Complement:

```
PS C:\Users\DELL\OneDrive\Desktop\Labs> cd "c:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5\" ; if ($?) { g++ binaty.cpp -o binaty } ; if ($?) { .\binaty }

**********Menu Options**********
 1. Insert binary1
 2. Insert binary 2
 3. Add binary no.s
 4. 1's and 2's compliment of binary no.
 5. Exit
---------------------------------

Enter your choice : 1
Enter the no. of bits : 4
---Enter binary number digit wise---
Enter binary digit : 1
Enter binary digit : 0
Enter binary digit : 1
Enter binary digit : 1

 1 0 1 1

Enter your choice : 2
Enter the no. of bits : 4
---Enter binary number digit wise---
Enter binary digit : 1
Enter binary digit : 0
Enter binary digit : 1
Enter binary digit : 0

 1 0 1 0

Enter your choice : 4

For first binary no.
1's compliment of  binary no. :
 0 1 0 0

2's compliment of  binary no. :
 0 1 0 1

 For second binary no.
 1's compliment of  binary no. :
  0 1 0 1

 2's compliment of  binary no. :
```

```
For second binary no.
1's compliment of  binary no. :
 0 1 0 1

2's compliment of  binary no. :
 0 1 1 0

Enter your choice : 5'
PS C:\Users\DELL\OneDrive\Desktop\Labs\DSA LAB\LAB 5>
```