MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY BHOPAL

DEPARTMENT OF CSE

Name: Yashwant Patidar

Scholar Number: 191112243

Section: CSE 2

3rd SEM BTech

Subject: Principles of Programming Languages Lab - CSE 219

Lab Assignment 2

Question 1: WAP to create, initialize, assign and access a pointer variable.

main.cpp

```cpp
#include <iostream>

using namespace std;

int main() {
    int a = 10;
    int *ptr; //create or declaration
    ptr = &a; //initialisation
    //accessing address at pointer
    cout<<"The address stored at pointer: "<<ptr;
    //accessing address OF pointer
    cout<<"\nThe address of the pointer variable: "<<&ptr;
    //accessing value at the address
    cout<<"\nThe value at the address stored in the pointer: "<<*ptr;
    return 0;
}
```

```
The address stored at pointer: 0x7ffed5596b64
The address of the pointer variable: 0x7ffed5596b68
The value at the address stored in the pointer: 10

...Program finished with exit code 0
Press ENTER to exit console.
```

Question 2: WAP to swap two numbers using pointers.

main.cpp

```cpp
#include <iostream>

using namespace std;

int main() {
    int a = 10;
    int b = 20;
    int *p = &a;
    int *q = &b;
    cout<<"Before swap:\ta = "<<a<<", b = "<<b;
    *p ^= *q ^= *p ^= *q;
    cout<<"\nAfter swap:\ta = "<<a<<", b = "<<b;
}
```

```
Before swap:    a = 10, b = 20
After swap:     a = 20, b = 10

...Program finished with exit code 0
Press ENTER to exit console.
```

Question 3: WAP to count all vowels and consonants in a string using a pointer.

main.cpp

```
#include <iostream>
#include <string>

using namespace std;

int main() {
  string s;
  cout<<"Enter the string to count vowels and consonants: ";
  getline(cin,s);
  char *p;
  p = &s[0];
  int v=0,c=0;
  while(*p!='\0'){
    if((*p>='a' && *p<='z') || (*p>='A' && *p<='Z'))
    {
      int isLowerVowel = (*p=='a' || *p=='e' || *p=='i' || *p=='o' || *p=='u');
      int isUpperVowel = (*p=='A' || *p=='E' || *p=='I' || *p=='O' || *p=='U');
      if (isLowerVowel || isUpperVowel)
        v++;
      else
        c++;
    }
  p++;
  }
  cout<<"The string contains:\n";
  cout<<v<<" vowels and "<<c<<" consonants.";
}
```

input

```
Enter the string to count vowels and consonants: binod binod binod binod
The string contains:
8 vowels and 12 consonants.

...Program finished with exit code 0
Press ENTER to exit console.
```

Question 4: WAP to read array elements and print with addresses.

main.cpp

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

int main()
{
  int size, *ptr;
  cout<<"Enter size of array: ";
  cin>>size;
  int arr[size];
  ptr = arr;
  cout << "Enter The Elements Of An array: ";
  for (int i = 0; i < size; i++)
    cin>>arr[i];

  cout<<"Printing elements and their addresses using pointers:\n";
  for (int i = 0; i < size; i++)
    cout<<"Address of element "<<*ptr<<" is "<<ptr++<<endl;
}
```

```
Enter size of array: 5
Enter The Elements Of An array: 5 2 4 10 7
Printing elements and their addresses using pointers:
Address of element 5 is 0x7ffef1392810
Address of element 2 is 0x7ffef1392814
Address of element 4 is 0x7ffef1392818
Address of element 10 is 0x7ffef139281c
Address of element 7 is 0x7ffef1392820


...Program finished with exit code 0
Press ENTER to exit console.
```

MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY BHOPAL

DEPARTMENT OF CSE

Name: Yashwant Patidar

Scholar Number: 191112243

Section: CSE 2

3rd SEM BTech

Subject: Principles of Programming Languages Lab - CSE 219

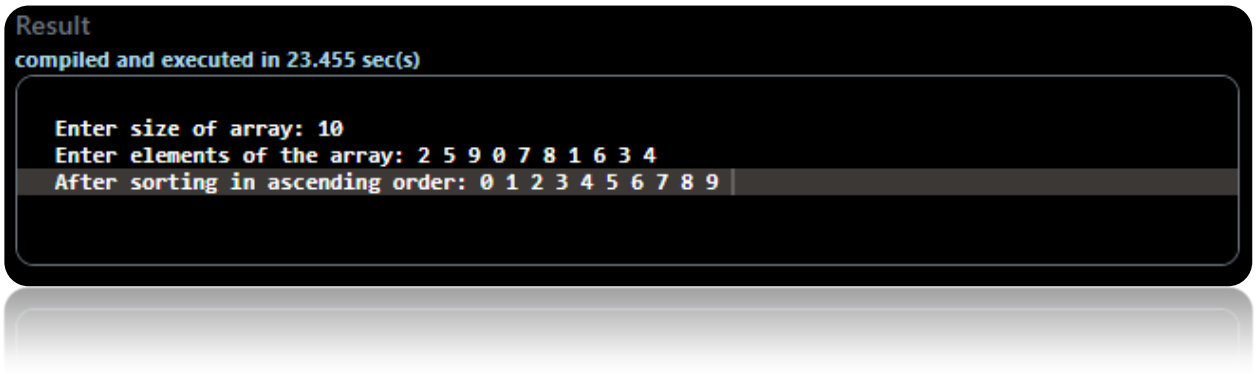
Lab Assignment 3

Question 1: WAP to sort an array using pointer.

Program code:

```
#include <iostream>

using namespace std;

int main() {
    int *ptr,n;
    cout<<"Enter size of array: ";
    cin>>n;
    cout<<"Enter elements of the array: ";
    for(int i=0;i<n;i++)
        cin>>*(ptr+i);

    for(int i=n-1;i>0;i--)
        for(int j=i-1;j>=0;j--)
            if((*(ptr+i))<(*(ptr+j))){
                int temp=*(ptr+i);
                *(ptr+i)=*(ptr+j);
                *(ptr+j)=temp;
            }

    cout<<"After sorting in ascending order: ";
    for(int i=0;i<n;i++)
        cout<<*ptr++<<" ";
    return 0;
}
```

Output:

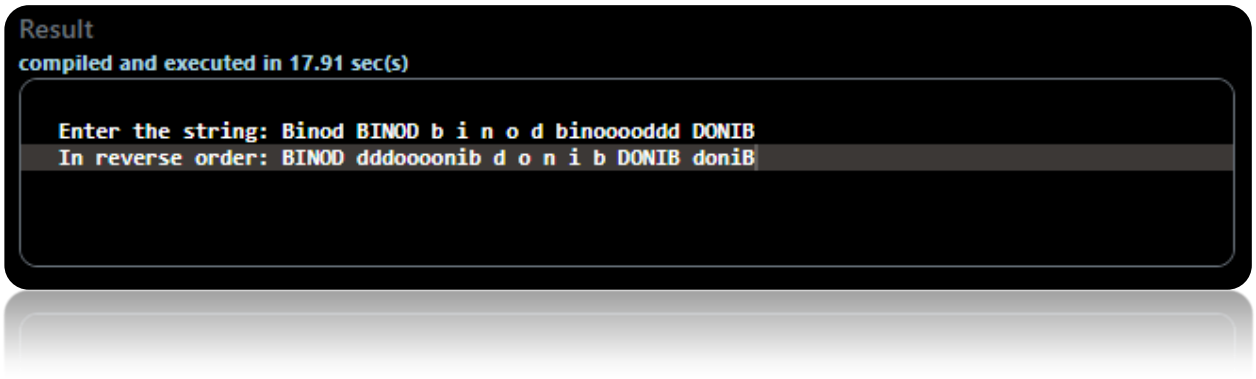

Result
compiled and executed in 23.455 sec(s)
Enter size of array: 10
Enter elements of the array: 2 5 9 0 7 8 1 6 3 4
After sorting in ascending order: 0 1 2 3 4 5 6 7 8 9

Question 2: WAP to print a string in reverse using a pointer.

Program Code:

```
#include <iostream>

using namespace std;

void reverse(char *p){
    if(*p=='\0') return;
    reverse(p+1);
    cout<<*p;
}

int main() {
    string s;
    cout<<"Enter the string: ";
    getline(cin,s);

    cout<<"In reverse order: ";
    char *p = &s[0];
    reverse(p);
}
```

Output:

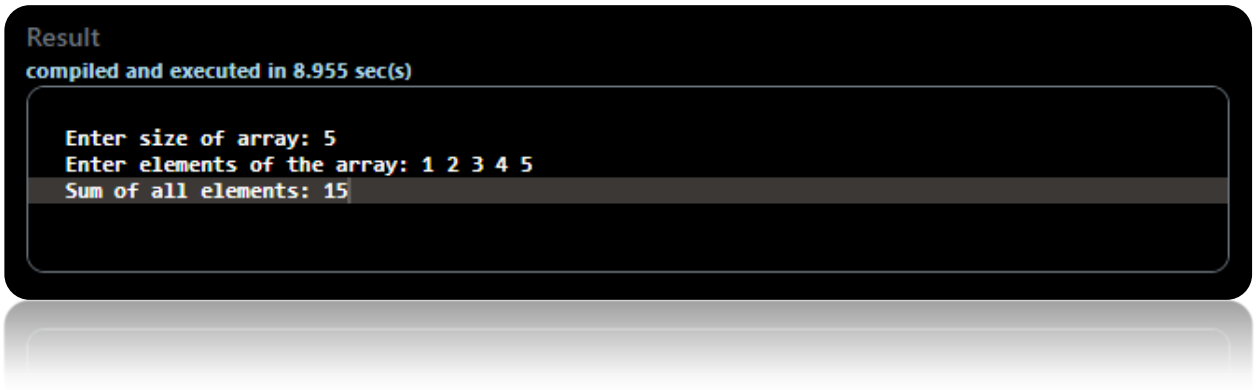


Question 3: WAP to compute the sum of all elements in an array using pointers.

Problem Code:

```
#include <iostream>

using namespace std;

int main() {
    int *ptr,n;
    cout<<"Enter size of array: ";
    cin>>n;
    cout<<"Enter elements of the array: ";
    for(int i=0;i<n;i++)
        cin>>*(ptr+i);

    for(int i=1;i<n;i++)
        *ptr+=*(ptr+i);

    cout<<"Sum of all elements: "<<*ptr;
}
```

Output:



MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY BHOPAL

DEPARTMENT OF CSE

Name: Yashwant Patidar

Scholar Number: 191112243

Section: CSE 2

3rd SEM BTech

Subject: Principles of Programming Languages Lab - CSE 219

Lab Assignment 2

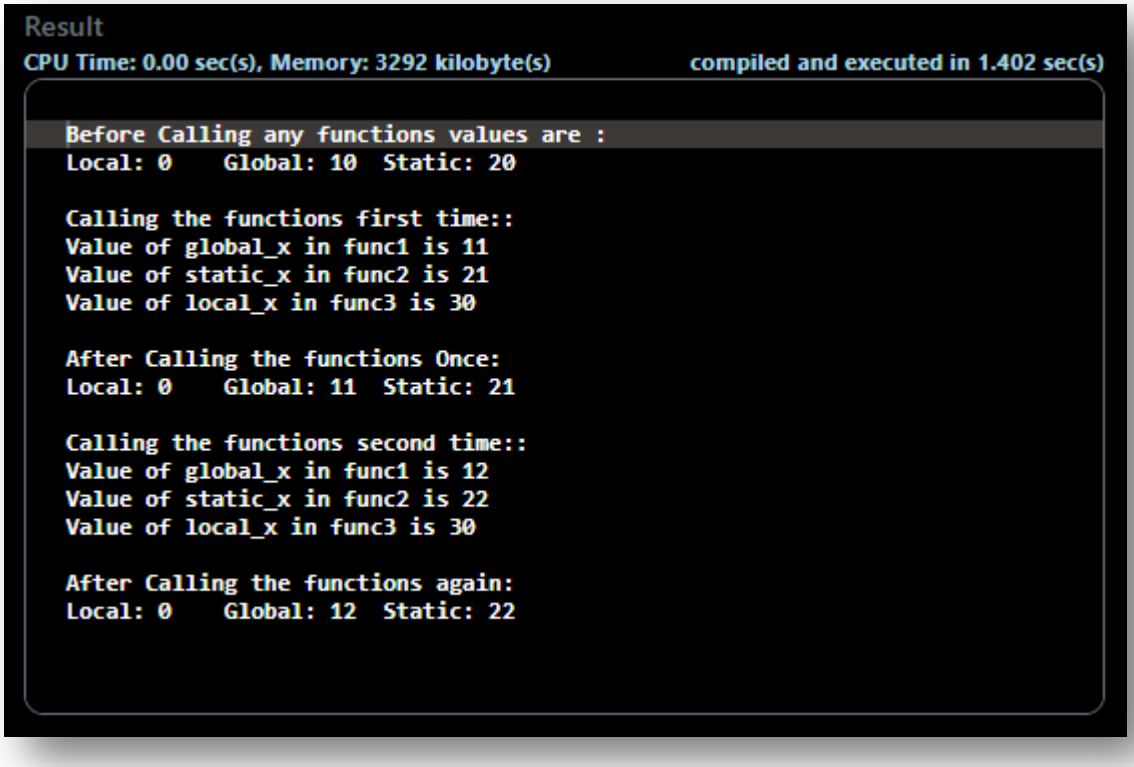
Question 1: WAP to use static, local, global variable in different functions and call that functions twice, then analyze the results and print final value of each variable.

Program code:

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

int global_x=10; //global variable
static int static_x=20;//static variable

void func1(){
  global_x++;
  cout<<"Value of global_x in func1 is "<<global_x<<endl;
}

void func2(){
  static_x++;
  cout<<"Value of static_x in func2 is "<<static_x<<endl;
}

void func3(){
  int local_x=30;
  cout<<"Value of local_x in func3 is "<<local_x<<endl;
}

int main(){
  int local_x=0;//local variable
  cout<<"Before Calling any functions values are :"<<endl;
  cout<<"Local: "<<local_x<<"\tGlobal: "<<global_x<<"\tStatic: "<<static_x<<endl;
```

```
    cout<<"\nCalling the functions first time::"<<endl;
    //calling functions once
    func1();
    func2();
    func3();

    cout<<"\nAfter Calling the functions Once: "<<endl;
    cout<<"Local: "<<local_x<<"\tGlobal: "<<global_x<<"\tStatic: "<<static_x<<endl;

    cout<<"\nCalling the functions second time::"<<endl;
    //calling functions again
    func1();
    func2();
    func3();

    cout<<"\nAfter Calling the functions again: "<<endl;
    cout<<"Local: "<<local_x<<"\tGlobal: "<<global_x<<"\tStatic: "<<static_x<<endl;
}
```

Output:

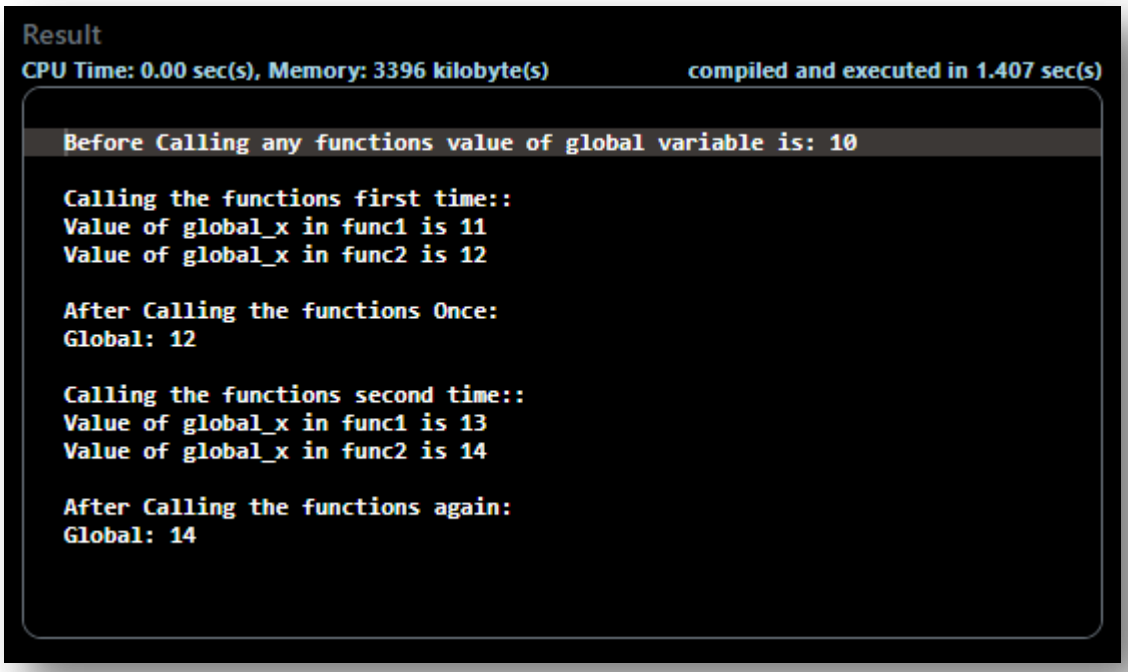


Question 2: WAP to define global variable which is accessed by `main()` and two other functions and then print the values of global variable in each function call.

Program Code:

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

int global_x=10; //global variable

void func1(){
  global_x++;
  cout<<"Value of global_x in func1 is "<<global_x<<endl;
}

void func2(){
  global_x++;
  cout<<"Value of global_x in func2 is "<<global_x<<endl;
}

int main(){

  cout<<"Before Calling any functions value of global variable is: "<<global_x<<e
ndl;

  cout<<"\nCalling the functions first time::"<<endl;
  //calling functions once
  func1();
  func2();

  cout<<"\nAfter Calling the functions Once: "<<endl;
  cout<<"Global: "<<global_x<<endl;

  cout<<"\nCalling the functions second time::"<<endl;
  //calling functions again
  func1();
  func2();

  cout<<"\nAfter Calling the functions again: "<<endl;
  cout<<"Global: "<<global_x<<endl;
}
```

Output:

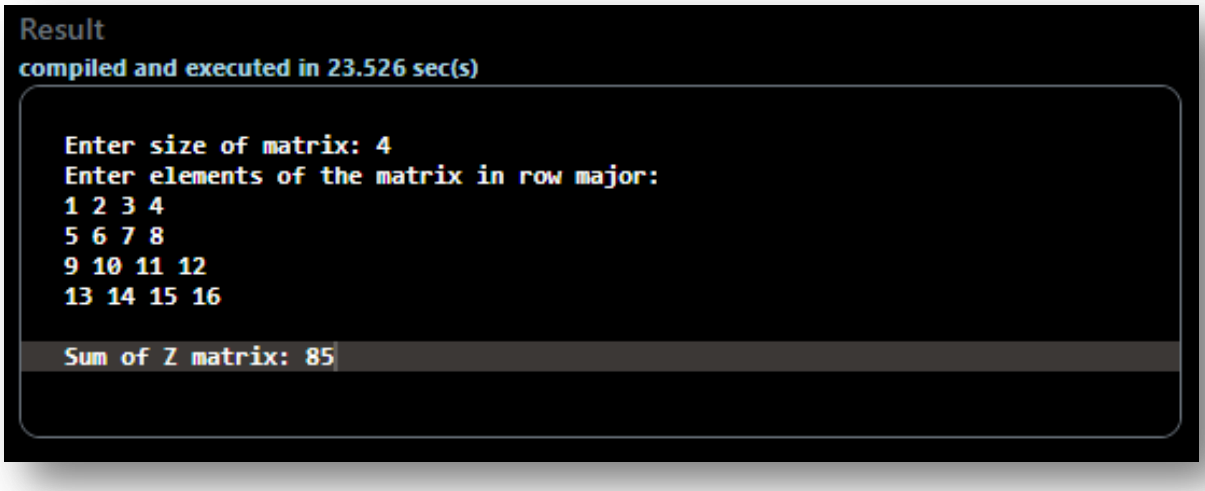

Result
CPU Time: 0.00 sec(s), Memory: 3396 kilobyte(s)
compiled and executed in 1.407 sec(s)
Before Calling any functions value of global variable is: 10
Calling the functions first time::
Value of global_x in func1 is 11
Value of global_x in func2 is 12
After Calling the functions Once:
Global: 12
Calling the functions second time::
Value of global_x in func1 is 13
Value of global_x in func2 is 14
After Calling the functions again:
Global: 14

Question 3: WAP to compute the sum of all elements in a Z-matrix using pointers.

Problem Code:

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

int main() {
    int n;
    cout<<"Enter size of matrix: ";
    cin>>n;
    int a[n][n],sum=0;
    cout<<"Enter elements of the matrix in row major: \n";
    for(int i=0;i<n;i++)
      for(int j=0;j<n;j++)
        cin>>a[i][j];
    //top row and bottom row sum
    for(int i=0;i<n;i++){
        sum+=a[0][i]+a[n-1][i]+a[i][n-1-i];
    }
    //subtracting the double counted elements at corners of the diagonal
    sum-=(a[0][n-1]+a[n-1][0]);
    cout<<"\nSum of Z matrix: "<<sum;
}
```

Output:

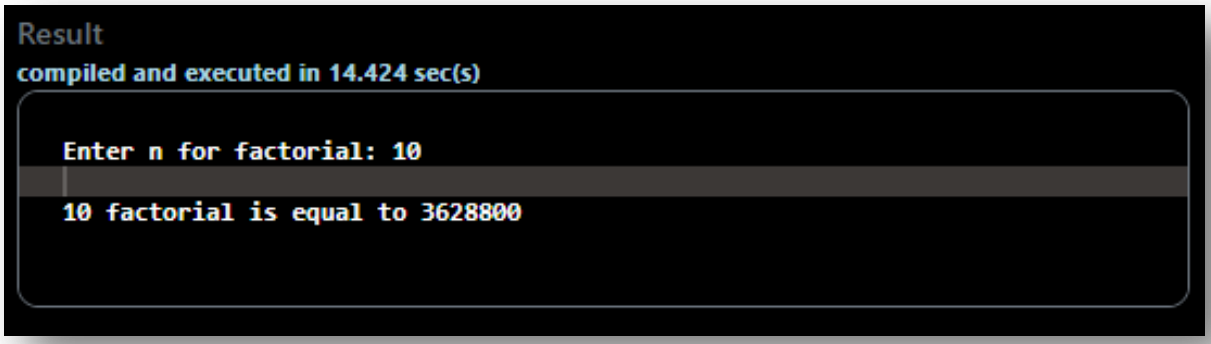


Question 4: WAP to calculate the factorial of n using recursion.

Problem Code:

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

long long fact(long long n){
  if(n<=2) return n;
  return n*fact(n-1);
}

int main() {
    long long n;
    cout<<"Enter n for factorial: ";
    cin>>n;
    cout<<"\n"<<n<<" factorial is equal to "<<fact(n);
}
```

Output:



MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY BHOPAL

DEPARTMENT OF CSE

Name: Yashwant Patidar

Scholar Number: 191112243

Section: CSE 2

3rd SEM BTech

Subject: Principles of Programming Languages Lab - CSE 219

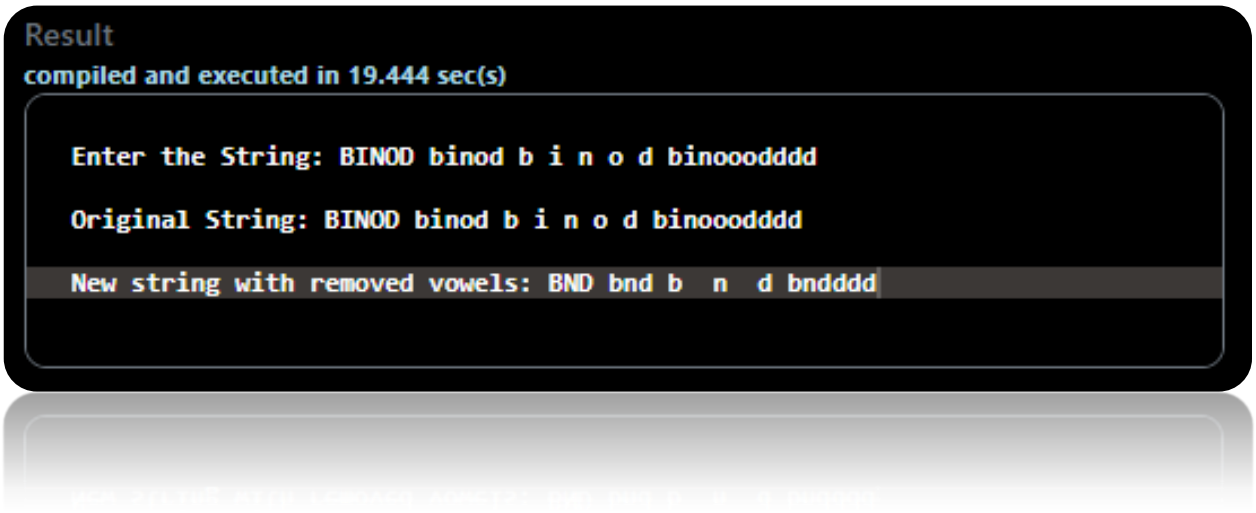
Lab Assignment

Question 1: WAP to delete all vowels from the string using call by value.

Program code:

```
#include <iostream>
#include <bits/stdc++.h>

using namespace std;

string replaceVowels(string s){
    regex reg("[aeiouAEIOU]");
    return regex_replace(s,reg,"");
}

int main() {
    string s1,s2;
    cout<<"Enter the String: ";

    getline(cin,s1);
    s2 = replaceVowels(s1);

    cout<<"\nOriginal String: "<<s1<<endl;

    cout<<"\nNew string with removed vowels: "<<s2;
}
```

Output:

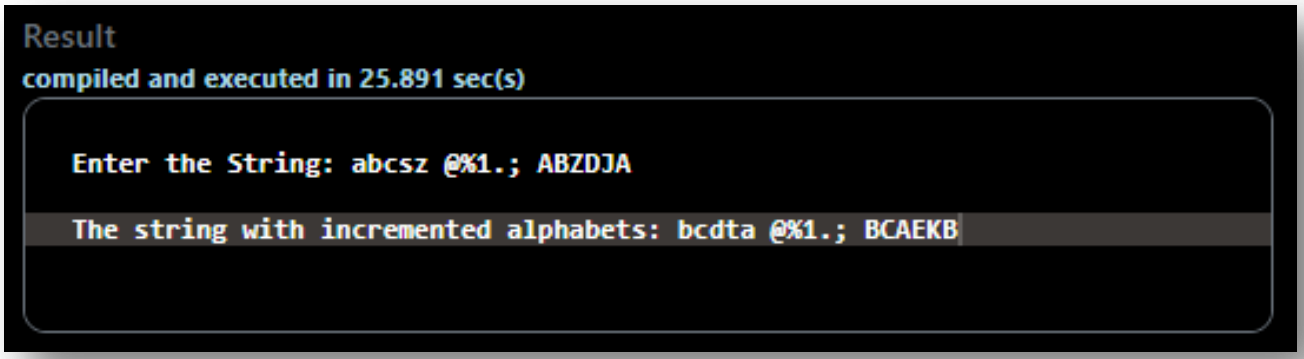

Result
compiled and executed in 19.444 sec(s)
Enter the String: BINOD binod b i n o d binooodddd
Original String: BINOD binod b i n o d binooodddd
New string with removed vowels: BND bnd b n d bnddd

Question 2: WAP to increment the alphabets of a string by one alphabet using call by reference.

Program Code:

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

void nextAlpha(char *ptr){
    while(*ptr!='\0'){
        if(isalpha(*ptr))
            if(*ptr == 'z')     *ptr = 'a';
            else if(*ptr =='Z') *ptr = 'A';
            else                *ptr +=1;
        ptr++;
    }
}

int main() {
    string s;
    cout<<"Enter the String: ";
    getline(cin,s);
    nextAlpha(&s[0]);
    cout<<"\nThe string with incremented alphabets: "<<s;
}
```

Output:

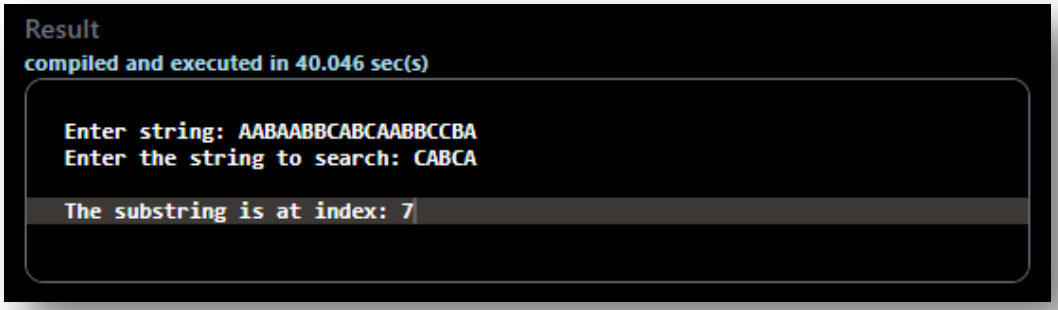


Question 3: WAP to check string 1 is a substring of string 2 or not.

Problem Code:

```
#include <bits/stdc++.h>
using namespace std;
int isSubstr(string s, string sub)
{
    Inti,j,l1 = s.length();
    int l2 = sub.length();
    for (i = 0; i <= l1 - l2; i++) {
        for (j = 0; j < l2; j++)
            if (s[i + j] != sub[j])
                break;
        if (j == l2)
            return i;
    }
    return -1;
}

int main()
{
    string s,sub;
    cout<<"Enter string: ";
    getline(cin,s);
    cout<<"Enter the string to search: ";
    getline(cin,sub);

    int index = isSubstr(s, sub);
    if (index == -1)    cout << "\nThere is no such substring in the string";
    else                cout << "\nThe substring is at index: " << index;
    return 0;
}
```

Output:



MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY BHOPAL

DEPARTMENT OF CSE

Name: Yashwant Patidar

Scholar Number: 191112243

Section: CSE 2

3rd SEM BTech

Subject: Principles of Programming Languages Lab - CSE 219

Lab Assignment

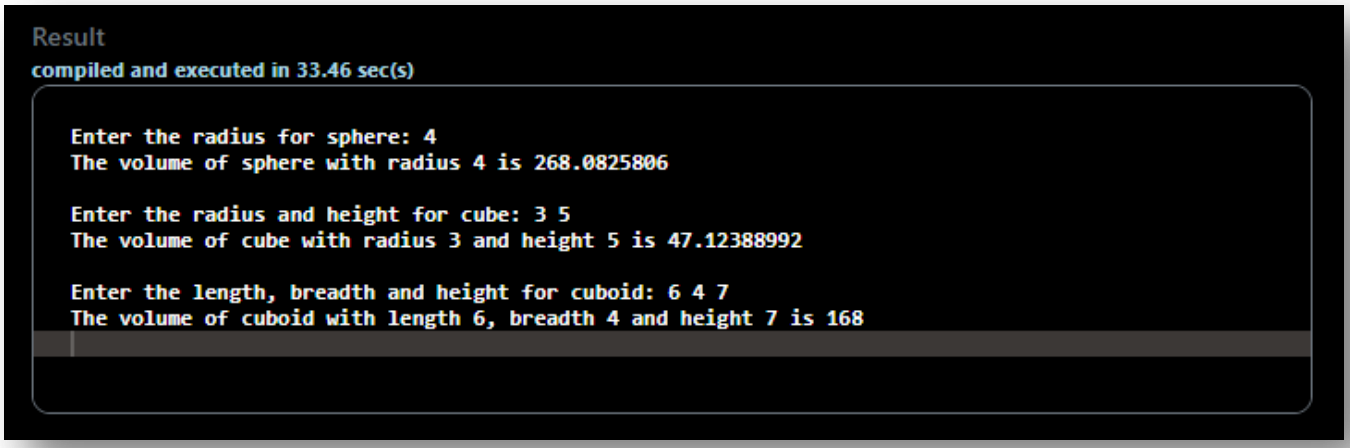
Question 1: WAP to calculate the volume of cube, sphere and cone using function overloading.

Program code:

```
#include <iostream>
#include <math.h>
using namespace std;
#define _USE_MATH_DEFINES

float volume(float r){
    return 4*M_PI*r*r*r/3;
}
float volume(float r, float h){
    return M_PI*r*r*h/3;
}
float volume(float a, float b, float c){
    return a*b*c;
}

int main() {
    cout.precision(10);
    float x,y,z;

    cout<<"Enter the radius for sphere: ";
    cin>>x;
    cout<<"The volume of sphere with radius "<<x<<" is "<<volume(x)<<endl;
```

```
    cout<<"\nEnter the radius and height for cube: ";
    cin>>x>>y;
    cout<<"The volume of cube with radius "<<x<<" and height "<<y<<" is "<<volume
(x,y)<<endl;

    cout<<"\nEnter the length, breadth and height for cuboid: ";
    cin>>x>>y>>z;
    cout<<"The volume of cuboid with length "<<x<<", breadth "<<y<<" and height "
<<z<<" is "<<volume(x,y,z)<<endl;
}
```

Output:

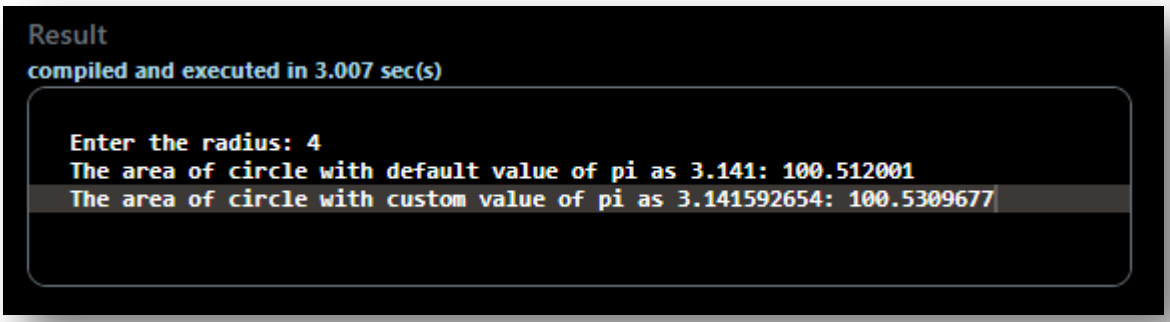


Question 2: WAP to compute the area of circle, use a default value of pi as 3.141 in case pi is omitted in the function call.

Program Code:

```
#include <iostream>
#include <math.h>
using namespace std;

#define _USE_MATH_DEFINES

float area(float r, float pi=3.141){
    return 2*pi*r*r;
}

int main() {
    cout.precision(10);
    float r;
    cout<<"Enter the radius: ";
    cin>>r;
    cout<<"The area of circle with default value of pi as 3.141: "<<area(r)<<endl
;
    cout<<"The area of circle with custom value of pi as "<<M_PI<<": "<<area(r,M_
PI);
}
```

Output:

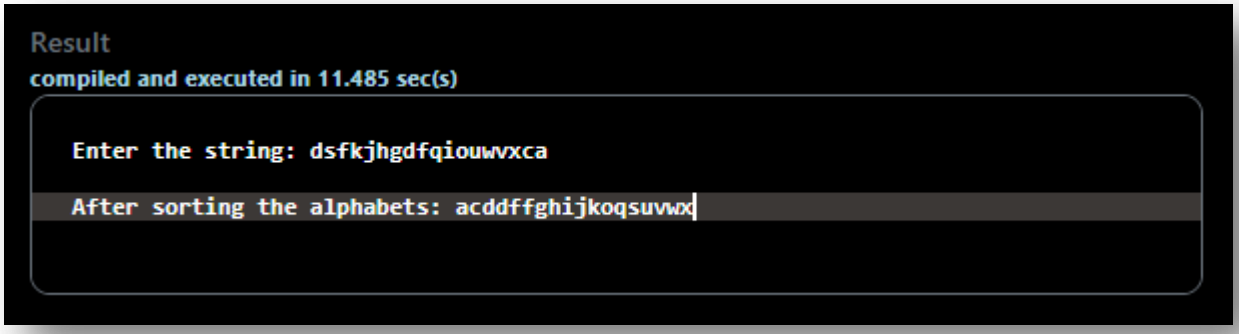


Question 3: WAP to sort the alphabets of given string using call by reference.

Problem Code:

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

void sort(string &s){
    int i,j;
    char temp;

    for(i = 0; i < s.length(); i++){
        temp = s[i];
        j=i-1;
        while(j>=0 && s[j] > temp){
            s[j+1] = s[j];
            j--;
        }
        s[j+1] = temp;
    }
}

int main() {
    string str;
    cout<<"Enter the string: ";
    cin>>str;
    sort(str);
    cout<<"\nAfter sorting the alphabets: "<<str;
}
```

Output:

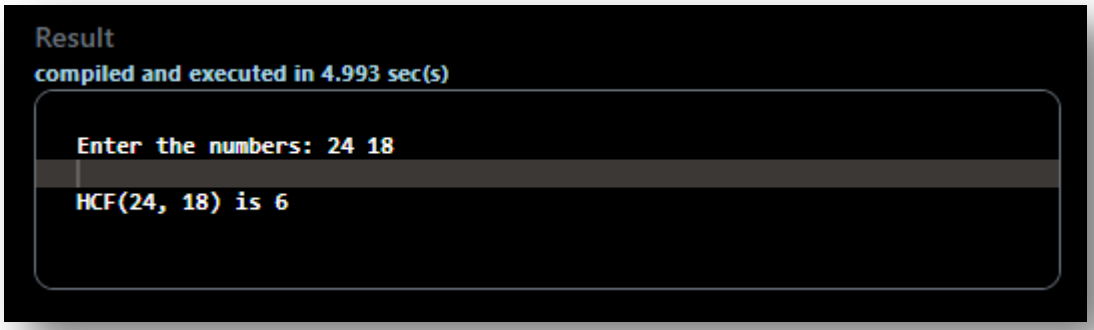


Question 4: WAP to calculate HCF(M,N) where M>N using "Euclid's division method" using recursion.

Problem Code:

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

int hcf(int m, int n){
    if(m==0)    return n;
    return hcf(n%m,m);
}

int main() {
    cout<<"Enter the numbers: ";
    int m,n;
    cin>>m>>n;
    cout<<"\nHCF("<<m<<", "<<n<<") is "<<hcf(m,n);
}
```

Output:



MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY BHOPAL

DEPARTMENT OF CSE

Name: Yashwant Patidar

Scholar Number: 191112243

Section: CSE 2

3rd SEM BTech

Subject: Principles of Programming Languages Lab - CSE 219

Lab Assignment

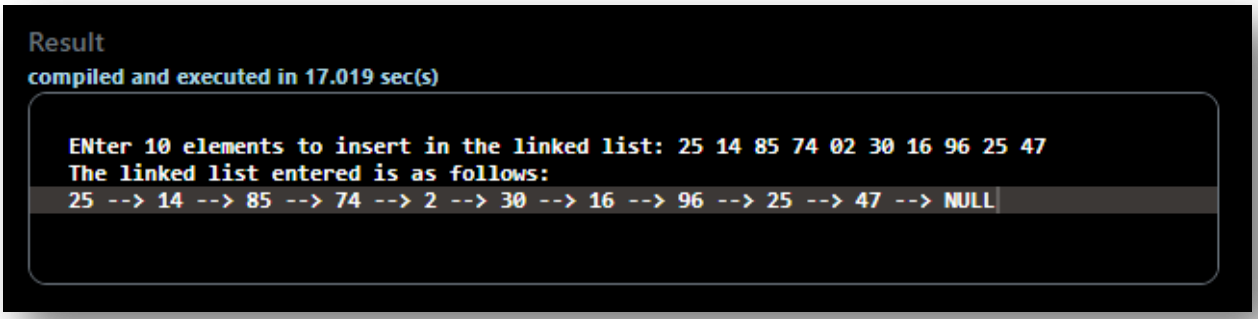
Question 1: WAP to insert 10 elements in a linked list.

Program code:

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

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

struct Node* input(){
    struct Node *header = new Node;
    cout<<"ENter 10 elements to insert in the linked list: ";
    cin>>header->data;
    struct Node *ptr = header;
    for(int i=0;i<9;i++){
        ptr->next = new Node;
        ptr = ptr->next;
        cin>>ptr->data;
    }
    return header;
}

void display(struct Node *ptr){
    if(ptr){
        cout<<ptr->data<<" --> ";
```

```
        display(ptr->next);
    } else
        cout<<"NULL";
}

int main() {
    struct Node *header;
    header = input();
    display(header);
}
```

Output:



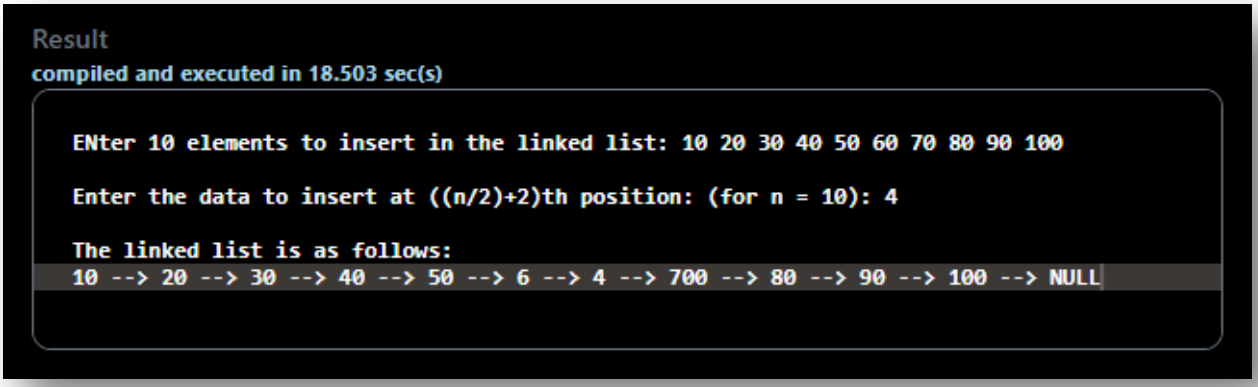
Question 2: WAP to add a node to ((n/2)+2)th position in an existing linked list.

Program Code:

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

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

struct Node* input(){
    struct Node *header = new Node;
    cout<<"ENter 10 elements to insert in the linked list: ";
    cin>>header->data;
    struct Node *ptr = header;
    for(int i=0;i<9;i++){
        ptr->next = new Node;
        ptr = ptr->next;
        cin>>ptr->data;
    }
    return header;
}

void insert(struct Node *ptr, int data){
    struct Node *q, *New = new Node;
    int i=1,n=10;
    for(; i < (n / 2 + 2) - 1; i++)
        ptr = ptr->next;
    q = ptr->next;
    ptr->next = New;
    New->data = data;
    New->next = q;
}

void display(struct Node *ptr){
    if(ptr){
        cout<<ptr->data<<" --> ";
        display(ptr->next);
    } else
        cout<<"NULL";
}

int main() {
    int data;
```

```
    struct Node *header;
    header = input();
    cout<<"\nEnter the data to insert at ((n/2)+2)th position: (for n = 10): ";
    cin>>data;
    insert(header, data);
    cout<<"\nThe linked list is as follows:\n";
    display(header);
}
```

Output:



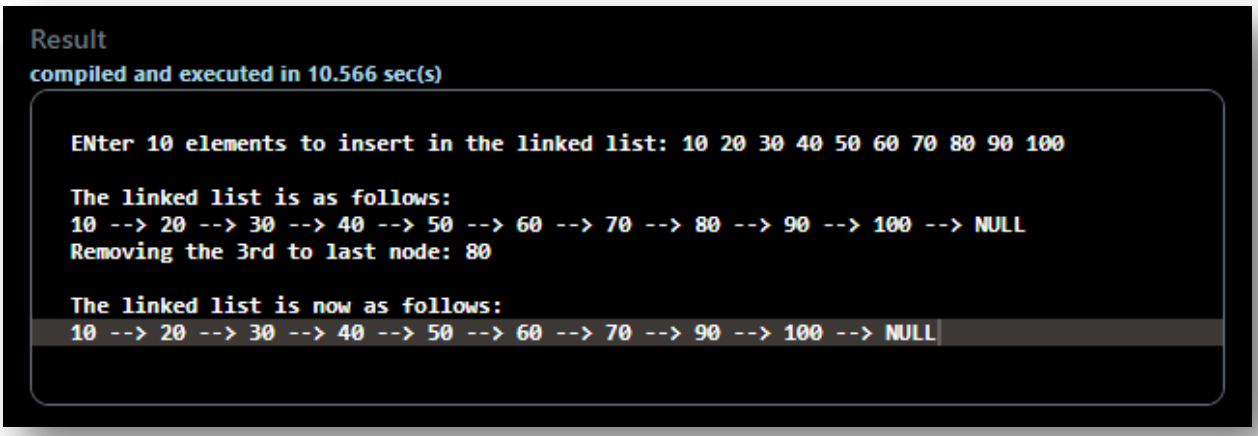
Question 3: WAP to remove a node 3rd to the last from an existing linked list.

Problem Code:

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

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

struct Node* input(){
    struct Node *header = new Node;
    cout<<"ENter 10 elements to insert in the linked list: ";
    cin>>header->data;
    struct Node *ptr = header;
    for(int i=0;i<9;i++){
        ptr->next = new Node;
        ptr = ptr->next;
        cin>>ptr->data;
    }
    return header;
}

int remove3rdLast(struct Node *ptr){
    struct Node *q = ptr;
    ptr = ptr->next->next->next->next;
    if(ptr == NULL){
        cout<<"No such element.";
        return -1;
    }
    while(ptr){
        ptr = ptr->next;
        q = q->next;
    }
    ptr = q->next;
    q->next = ptr->next;
    int data = ptr->data;
    free(ptr);
    return data;
}

void display(struct Node *ptr){
    if(ptr){
        cout<<ptr->data<<" --> ";
```

```
        display(ptr->next);
    } else
        cout<<"NULL";
}

int main() {
    int data;
    struct Node *header;
    header = input();
    cout<<"\nThe linked list is as follows:\n";
    display(header);
    cout<<"\nRemoving the 3rd to last node: ";
    data = remove3rdLast(header);
    cout<<data<<endl;
    cout<<"\nThe linked list is now as follows:\n";
    display(header);
}
```

Output:



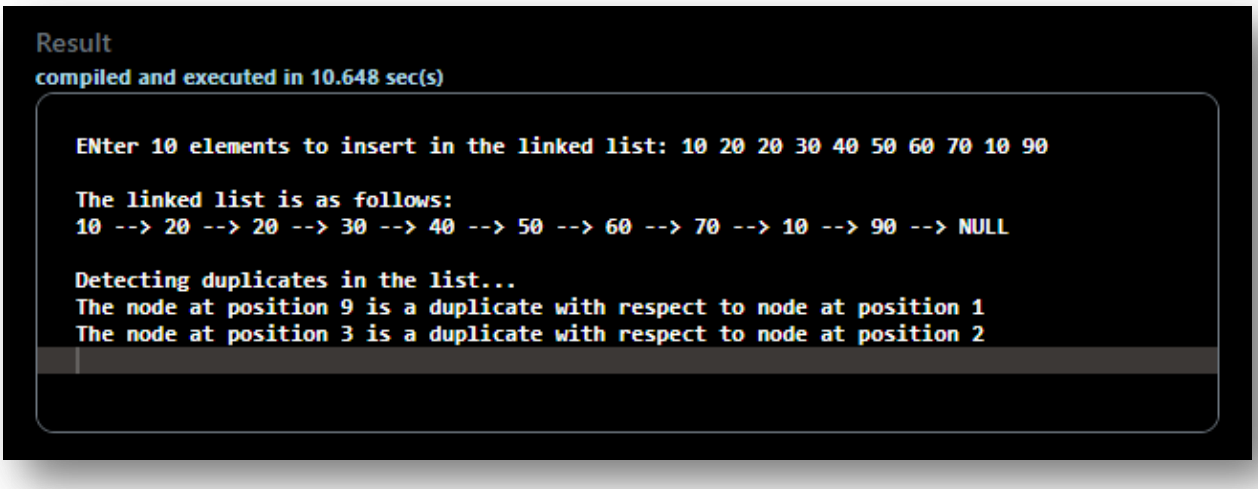
Question 4: WAP to detect duplicate node(s) if exists in a linked list.

Problem Code:

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

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

struct Node* input(){
    struct Node *header = new Node;
    cout<<"ENter 10 elements to insert in the linked list: ";
    cin>>header->data;
    struct Node *ptr = header;
    for(int i=0;i<9;i++){
        ptr->next = new Node;
        ptr = ptr->next;
        cin>>ptr->data;
    }
    return header;
}

void detectDuplicates(struct Node *ptr){
    int i =1,j, flag=0;
    struct Node *q;
    for(; ptr; i++ ){
        for(j=i+1, q=ptr->next; q; j++ ){
            if(ptr->data == q->data){
                cout<<"The node at position "<<j<<" is a duplicate with respect t
o node at position "<<i<<endl;
                flag = 1;
            }
            q = q->next;
        }
        ptr = ptr->next;
    }

    if(!flag)
        cout<<"No duplicates found."<<endl;

}
```

```
void display(struct Node *ptr){
    if(ptr){
        cout<<ptr->data<<" --> ";
        display(ptr->next);
    } else
        cout<<"NULL";
}

int main() {
    int data;
    struct Node *header;
    header = input();
    cout<<"\nThe linked list is as follows:\n";
    display(header);
    cout<<"\n\nDetecting duplicates in the list..."<<endl;
    detectDuplicates(header);
}
```

Output:



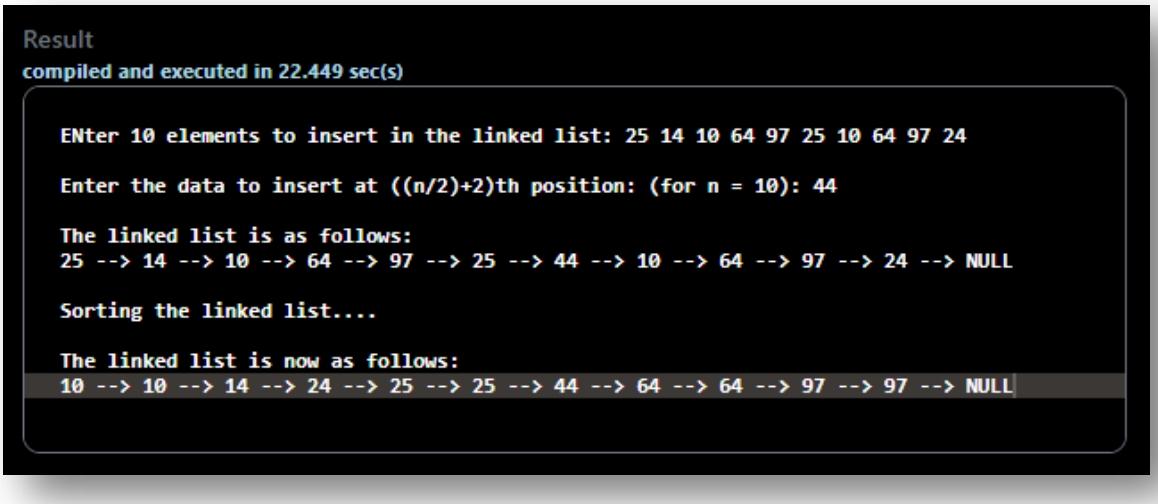
Question 5: WAP to sort the linked list after implementation of question 2.

Problem Code:

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

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

struct Node* input(){
    struct Node *header = new Node;
    cout<<"ENter 10 elements to insert in the linked list: ";
    cin>>header->data;
    struct Node *ptr = header;
    for(int i=0;i<9;i++){
        ptr->next = new Node;
        ptr = ptr->next;
        cin>>ptr->data;
    }
    return header;
}

void insert(struct Node *ptr, int data){
    struct Node *q, *New = new Node;
    int i=1,n=10;
    for(; i < (n / 2 + 2) - 1; i++)
        ptr = ptr->next;
    q = ptr->next;
    ptr->next = New;
    New->data = data;
    New->next = q;
}

void sort(struct Node *ptr){
    struct Node *q;
    while(ptr){
        q = ptr->next;
        while(q){
            if(ptr->data > q->data){
                int temp = q->data;
                q->data = ptr->data;
                ptr->data = temp;
            }
```

```
            q = q->next;
        }
        ptr = ptr->next;
    }
}

void display(struct Node *ptr){
    if(ptr){
        cout<<ptr->data<<" --> ";
        display(ptr->next);
    } else
        cout<<"NULL";
}

int main() {
    int data;
    struct Node *header;
    header = input();
    cout<<"\nEnter the data to insert at ((n/2)+2)th position: (for n = 10): ";
    cin>>data;
    insert(header, data);
    cout<<"\nThe linked list is as follows:\n";
    display(header);

    cout<<"\n\nSorting the linked list...."<<endl;
    sort(header);
    cout<<"\nThe linked list is now as follows: "<<endl;
    display(header);
}
```

Output:



MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY BHOPAL

DEPARTMENT OF CSE

Name: Yashwant Patidar

Scholar Number: 191112243

Section: CSE 2

3rd SEM BTech

Subject: Principles of Programming Languages Lab - CSE 219

Lab Assignment

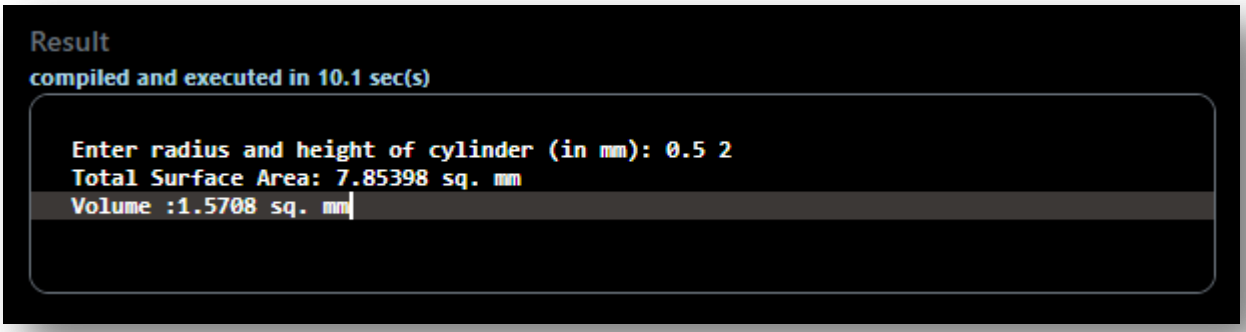
Question 1: WAP to print the total surface area and volume of a cylinder by creating a class named "cylinder" with a function to print the area and volume.

Program code:

```
#include <iostream>
#include <math.h>
using namespace std;
#define _USE_MATH_DEFINES

class Cylinder{
    float r;
    float h;
    public:
    Cylinder(float radius, float height){
        r = radius;
        h = height;
    }
    float surfaceArea(){
        return 2*M_PI*r*(r+h);
    }
    float volume(){
        return M_PI*r*r*h;
    }
};

int main() {
    float r,h;
```

```
    cout<<"Enter radius and height of cylinder (in mm): ";
    cin>>r>>h;
    Cylinder c(r,h);
    cout<<"Total Surface Area: "<<c.surfaceArea()<<" sq. mm"<<endl;
    cout<<"Volume :"<<c.volume()<<" sq. mm";
}
```

Output:

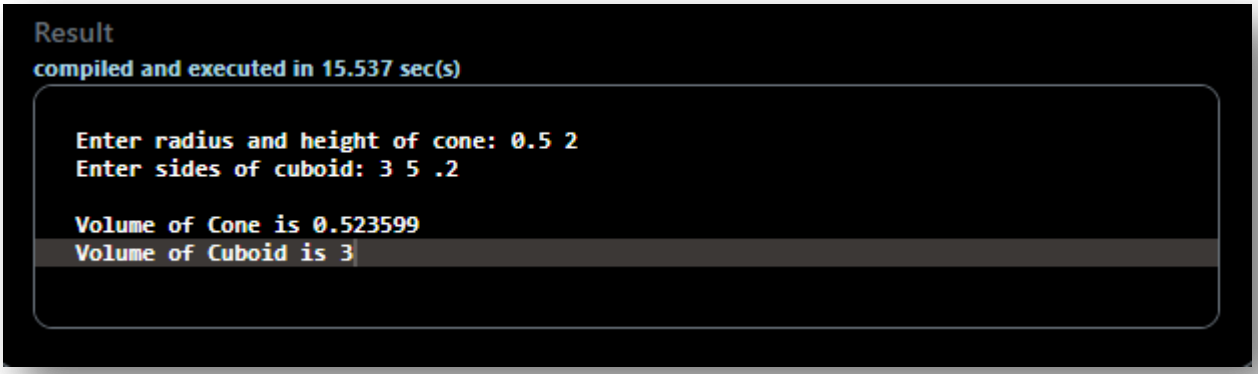


Question 2: WAP to print the volume of cones and cuboids by creating a class named "volume". Height, radius and sides are passed as parameter to its constructor.

Program Code:

```
#include <iostream>
#include <math.h>
using namespace std;
#define _USE_MATH_DEFINES

class Volume{
    float r,h;
    float a,b,c;
    public:
    Volume(float radius, float height){
        r = radius;
        h = height;
    }
    Volume(float x,float y, float z){
        a=x;
        b=y;
        c=z;
    }
    float cone(){
        return M_PI*r*r*h/3;
    }
    float cuboid(){
        return a*b*c;
    }
};

int main() {
    float a,b,c,r,h;
    cout<<"Enter radius and height of cone: ";
    cin>>r>>h;
    cout<<"Enter sides of cuboid: ";
    cin>>a>>b>>c;
    Volume Cone(r,h);
    Volume Cuboid(a,b,c);
    cout<<"\nVolume of Cone is "<<Cone.cone()<<endl;
    cout<<"Volume of Cuboid is "<<Cuboid.cuboid();
}
```

Output:

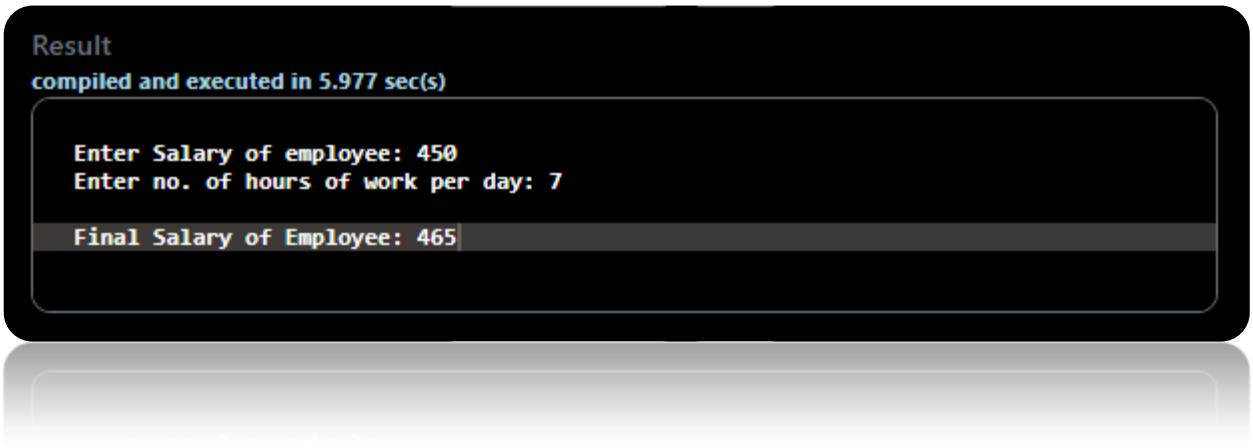

Result
compiled and executed in 15.537 sec(s)
Enter radius and height of cone: 0.5 2
Enter sides of cuboid: 3 5 .2
Volume of Cone is 0.523599
Volume of Cuboid is 3

Question 3: WAP by creating an 'Employee' class having the following functions and print final salary: 1- '`AddInfo()`' which takes the salary, number of hours of work per day of employees as parameters. 2 – '`AddSal()`' which adds $10 to the salary of the employee if it is less than $500. 3 – '`AddWork()`' which adds $5 to the salary of the employee if the number of hours of work per day is more than 6 hours.

Problem Code:

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

class Employee{
    public:
    float sal;
    int hours;
    void AddInfo(float salary, int h){
        sal = salary;
        hours = h;
    }
    void AddSal(){
        if(sal < 500) sal+=10;
    }
    void AddWork(){
        if(hours > 6) sal+=5;
    }
};

int main() {
    float sal;
    int h;
    cout<<"Enter Salary of employee: ";
    cin>>sal;
    cout<<"Enter no. of hours of work per day: ";
    cin>>h;
    Employee E;
    E.AddInfo(sal,h);
    E.AddSal();
    E.AddWork();
    cout<<"\nFinal Salary of Employee: "<<E.sal;
}
```

Output:

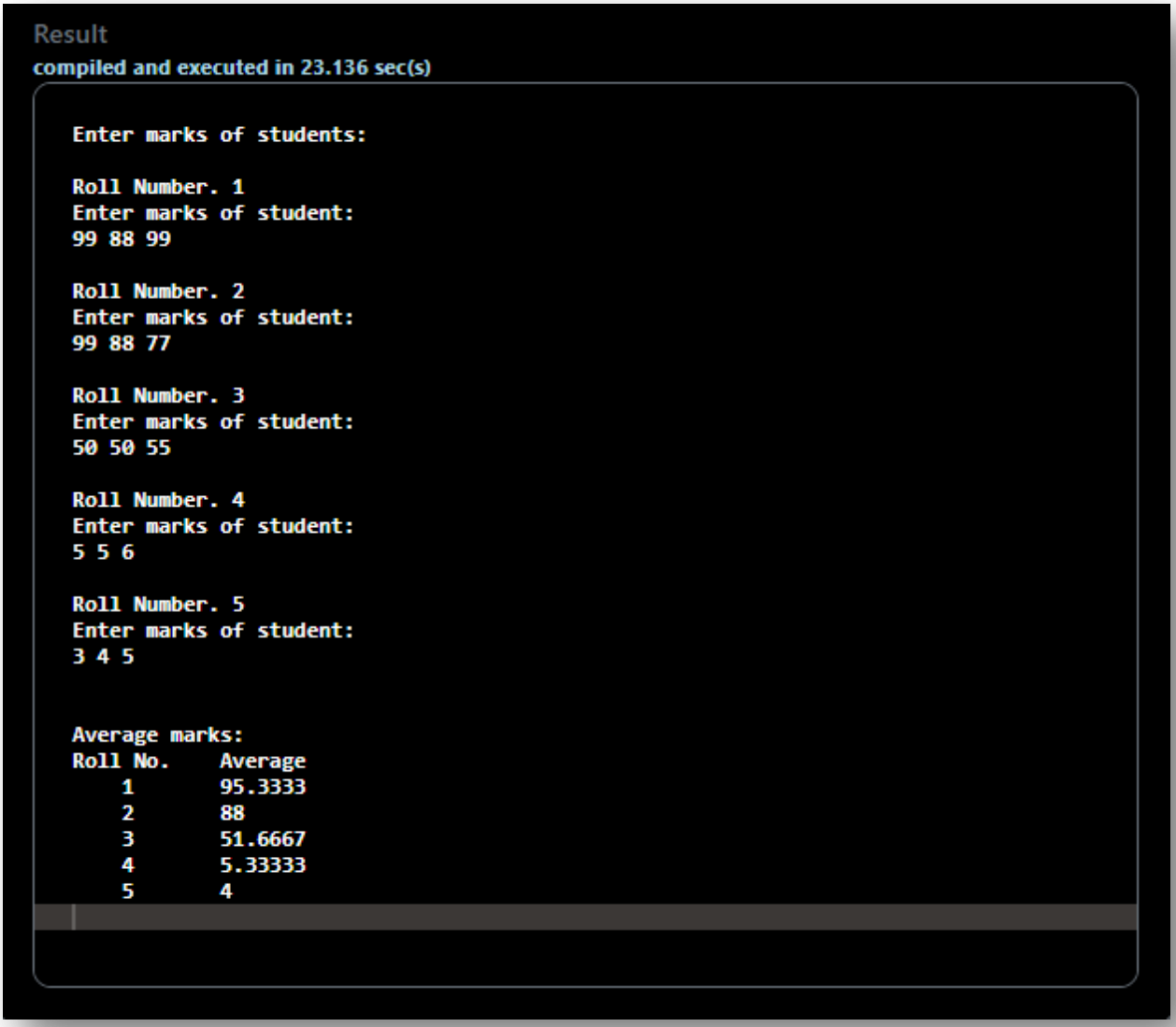

Result
compiled and executed in 5.977 sec(s)
Enter Salary of employee: 450
Enter no. of hours of work per day: 7
Final Salary of Employee: 465

Question 4: WAP to print the roll number and average marks of 5 students in three subjects (each out of 100). The marks are entered by the user and the roll numbers are automatically assigned.

Problem Code:

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

class Student{
public:
    int roll;
    int s1,s2,s3;
    float average;

    Student() {} //dummy constructor

    //parameterized constructor
    Student(int r,int sub1, int sub2, int sub3){
        roll=r;
        s1=sub1;
        s2=sub2;
        s3=sub3;
        average = (float)(s1+s2+s3)/3;
    }
};

int main() {
    int s1,s2,s3;
    Student* S = new Student[5];
    cout<<"Enter marks of students: "<<endl;
    for(int i=1;i<=5;i++){
        cout<<"\nRoll Number. "<<i<<"\nEnter marks of student: "<<endl;
        cin>>s1>>s2>>s3;
        S[i-1] = Student(i,s1,s2,s3);
    }
    cout<<"\n\nAverage marks:"<<endl;
    cout<<"Roll No.\tAverage"<<endl;
    for(int i=0;i<5;i++){
        cout<<"\t"<<S[i].roll<<"\t\t"<<S[i].average<<endl;
    }
}
```

Output:


Result
compiled and executed in 23.136 sec(s)
Enter marks of students:
Roll Number. 1
Enter marks of student:
99 88 99
Roll Number. 2
Enter marks of student:
99 88 77
Roll Number. 3
Enter marks of student:
50 50 55
Roll Number. 4
Enter marks of student:
5 5 6
Roll Number. 5
Enter marks of student:
3 4 5
Average marks:
Roll No. Average
1 95.3333
2 88
3 51.6667
4 5.33333
5 4

MAULANA AZAD NATIONAL INSTITUTE OF TECHNOLOGY BHOPAL

DEPARTMENT OF CSE

Name: Yashwant Patidar

Scholar Number: 191112243

Section: CSE 2

3rd SEM BTech

Subject: Principles of Programming Languages Lab - CSE 219

Lab Assignment

Question 1: WAP to create two classes named A and B, create another class named C which inherits both class A and B. Now, create a function in each of these classes which prints “A-class”, “B-class” and “both-class A and B” respectively. Now create an object for each class. Call the function of each of its parent by the object of class C.

Program code:

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

class A{
public:
    void print(){ cout<<"A-class"<<endl;}
};

class B{
public:
    void print(){ cout<<"B-class"<<endl;}
};

class C: public A, public B{
public:
    void print(){ cout<<"both-class:A and B"<<endl;}
};

int main(){
    A a;
    B b;
    C c;
```

```
        a.print();
        b.print();
        c.print();

        c.A::print();
        c.B::print();
        c.print();
    }
```

Output:

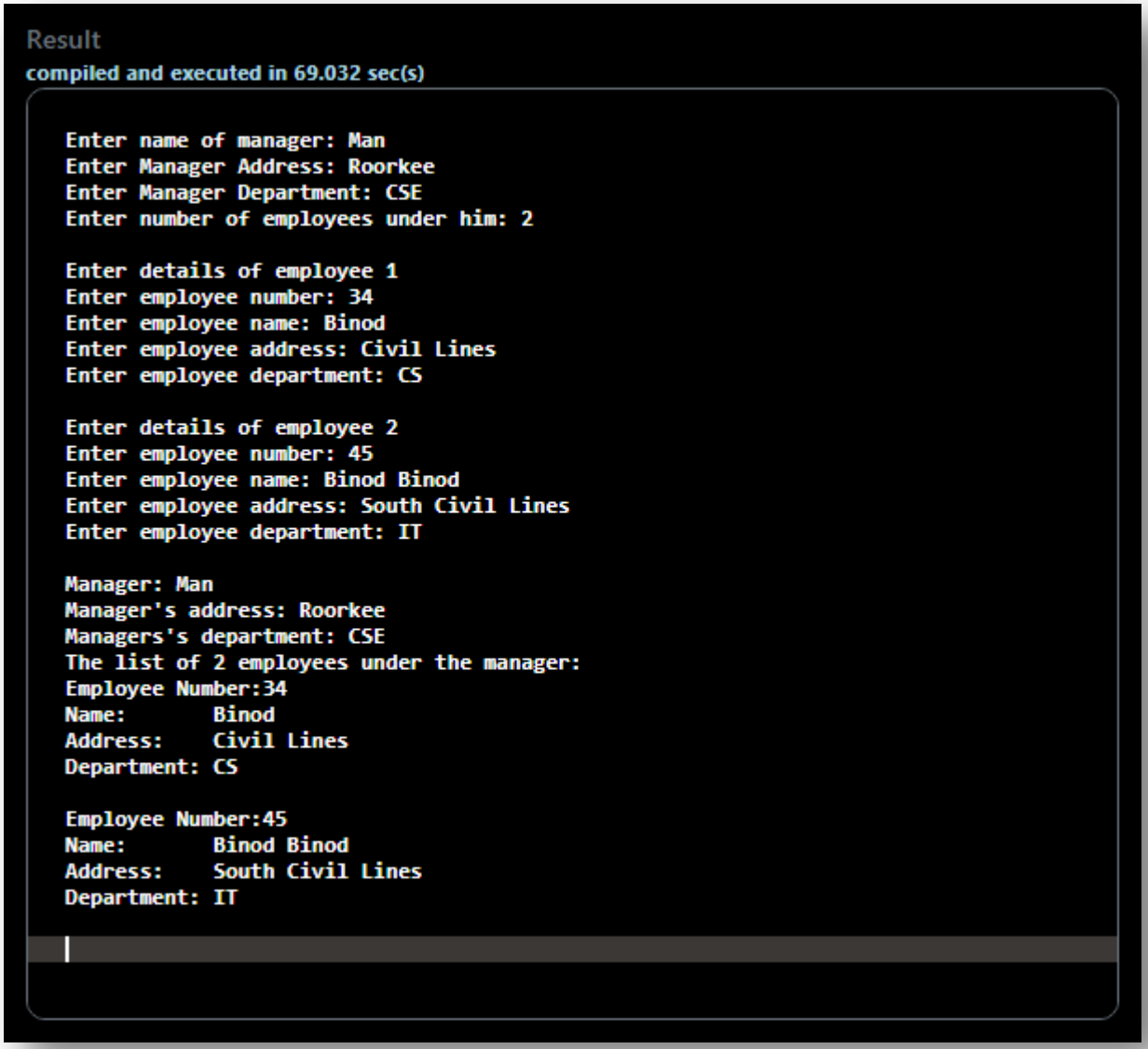
Question 2: WAP to read and display information about employees and managers. Emp is a class that contains emp_no, name, address and department. Manager class contains all information of the Emp class and a list of employees working under a manager.

Program Code:

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

class Emp{
protected:
    int emp_no;
    string name, address, dept;
public:
    void getInfo(){
        cout<<"Enter employee number: ";
        cin>>emp_no;
        cout<<"Enter employee name: ";
        getline(cin.ignore(), name);
        cout<<"Enter employee address: ";
        getline(cin, address);
        cout<<"Enter employee department: ";
        getline(cin, dept);
    }
    void display(){
        cout<<"Employee Number:"<<emp_no<<endl;
        cout<<"Name:\t\t"<<name<<endl;
        cout<<"Address:\t"<<address<<endl;
        cout<<"Department:\t"<<dept<<endl;
    }
};

class Manager:public Emp{
    int size;
    Emp* Employees;
public:
    void getInfo(){
        cout<<"Enter name of manager: ";
        getline(cin, name);
        cout<<"Enter Manager Address: ";
        getline(cin, address);
        cout<<"Enter Manager Department: ";
        getline(cin, dept);
```

```
        cout<<"Enter number of employees under him: ";
        cin>>size;
        Employees = new Emp[size];
        for(int i=0;i<size;i++){
            cout<<"\nEnter details of employee "<<i+1<<endl;
            Employees[i].getInfo();
        }
    }
    void display(){
        cout<<"\nManager: "<<name<<endl;
        cout<<"Manager's address: "<<address<<endl;
        cout<<"Managers's department: "<<dept<<endl;
        cout<<"The list of "<<size<<" employees under the manager:"<<endl;
        for(int i=0;i<size;i++){
            Employees[i].display();
            cout<<endl;
        }
    }
};
int main(){
    Manager M;
    M.getInfo();
    M.display();
}
```

Output:


Result
compiled and executed in 69.032 sec(s)
Enter name of manager: Man
Enter Manager Address: Roorkee
Enter Manager Department: CSE
Enter number of employees under him: 2
Enter details of employee 1
Enter employee number: 34
Enter employee name: Binod
Enter employee address: Civil Lines
Enter employee department: CS
Enter details of employee 2
Enter employee number: 45
Enter employee name: Binod Binod
Enter employee address: South Civil Lines
Enter employee department: IT
Manager: Man
Manager's address: Roorkee
Managers's department: CSE
The list of 2 employees under the manager:
Employee Number:34
Name: Binod
Address: Civil Lines
Department: CS
Employee Number:45
Name: Binod Binod
Address: South Civil Lines
Department: IT

Question 3: WAP to print the factorial of a number given by user by creating a class Factorial. If no number is passed by the user while creating an object of factorial class, then the number should be 0, using constructor overloading.

Problem Code:

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

class Factorial{
    int num;
    int fact(int n){
        return n <= 1 ? 1 : fact(n-1)*n;
    }
public:
    Factorial(int n){num=n;}
    Factorial(){num=0;}
    int calc(){return fact(num);}
};
int main(){
    Factorial F1, F2(6);
    cout<<"Factorial by not passing any argument: "<<F1.calc()<<endl;
    cout<<"Factorial by passing 6: "<<F2.calc()<<endl;
}
```

Output: