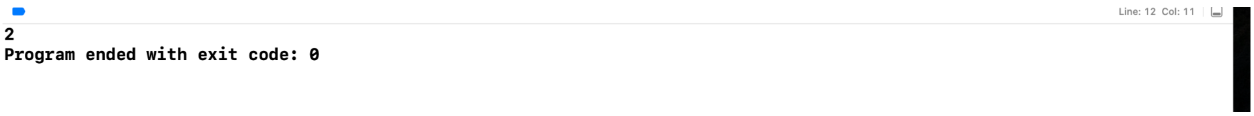
Assignment-1

Question 1

Create class Point(X, Y) and find distance between two points using friend function?

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

class Point{
    int x;
    int y;
public:
    void set_x(int X);
    void set_y(int Y);
    friend float distance(Point a,Point b);

};

void Point::set_x(int X)
{
    x=X;
}
```

```
void Point::set_y(int Y)
{
    y=Y;
}

float distance(Point a,Point b)
{
    float distance=0;

    int  q= (a.x - b.x);
    int  r= (a.y - b.y);

    distance= sqrt(q*q + r*r);
    return distance;
}



int main() {

    Point c,d;
    c.set_x(2);
    c.set_y(3);
    d.set_x(2);
    d.set_y(5);

    cout<<distance(c, d)<<endl;


    return 0;
}
```



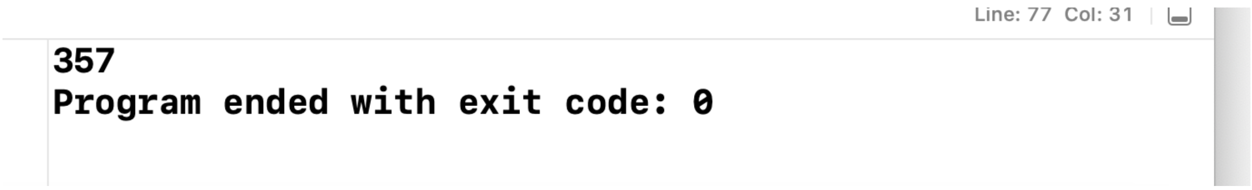
```
2
Program ended with exit code: 0
```

Question 2

Create Class Point(X, Y, Z) and find mid point by passing object and returning object from function, function should not be member of class.

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

class Point{
    float x;
    float y;
    float z;
public:
    void set_x(float X);
    void set_y(float Y);
    void set_z(float Z);
    float get_x(float X);
    float get_y(float Y);
    float get_z(float Z);

    friend Point midPoint(Point a,Point b);

};

void Point::set_x(float X)
{
    x=X;
}
void Point::set_y(float Y)
{
    y=Y;
}
void Point::set_z(float Z)
{
    z=Z;
}
float Point::get_x(float X)
{
    return X;
}
float Point::get_y(float Y)
{
    return Y;
}
float Point::get_z(float Z)
{
    return Z;
}

Point midPoint(Point a,Point b)
{
    Point tmp;
    float mp_x=(a.x+b.x)/2;
    float mp_y=(a.y+b.y)/2;
    float mp_z=(a.z+b.z)/2;
    tmp.set_x(mp_x);
    tmp.set_y(mp_y);
    tmp.set_z(mp_z);
    return tmp;
}
```

```
int main() {

    Point c,d,j;
    c.set_x(2);
    c.set_y(4);
    c.set_z(6);
    d.set_x(4);
    d.set_y(6);
    d.set_z(8);

   j= midPoint(c, d);

    cout<<j.get_x()<<endl;


    return 0;
}
```

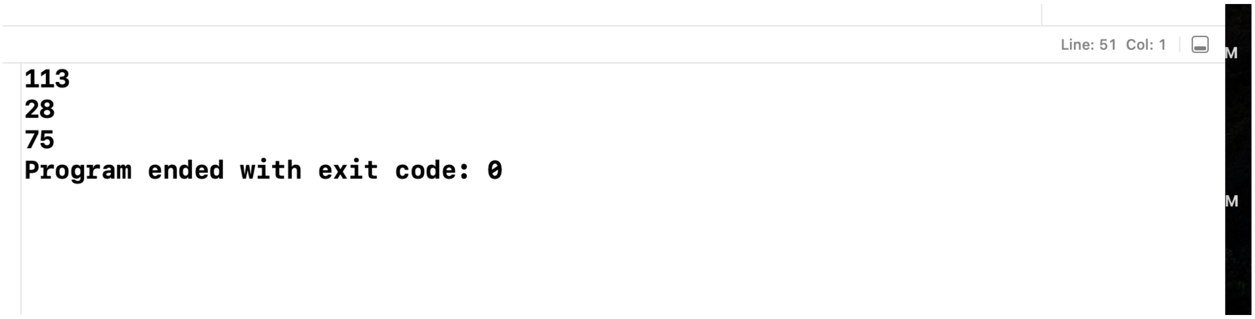


```
357
Program ended with exit code: 0
```

Question 3

create class Cylinder(r,h) and find volume, base area and surface area using three different member functions? Radius and height of object should be initialized using parameterized constructor and explicit call.

```
#include <iostream>
#include <math.h>

using namespace std;

class Cylinder{
    int r;
    int h;
public:

    Cylinder(int x,int y)
    {
        r=x;
        h=y;
    }

    int volume(int r,int h);
    int base_area(int r,int h);
    int surface_area(int r,int h);
};

int Cylinder :: volume(int r, int h)
{
    return 3.14*r*r*h;
}

int Cylinder :: base_area(int r, int h)
{
    return 3.14*r*r;
}

int Cylinder :: surface_area(int r, int h)
{
    return 2*3.14*r*h;
}



int main() {
    Cylinder c = Cylinder(3, 4);
    cout<<c.volume(3, 4)<<endl;
    cout<<c.base_area(3, 4)<<endl;
    cout<<c.surface_area(3, 4)<<endl;
    return 0;
}
```



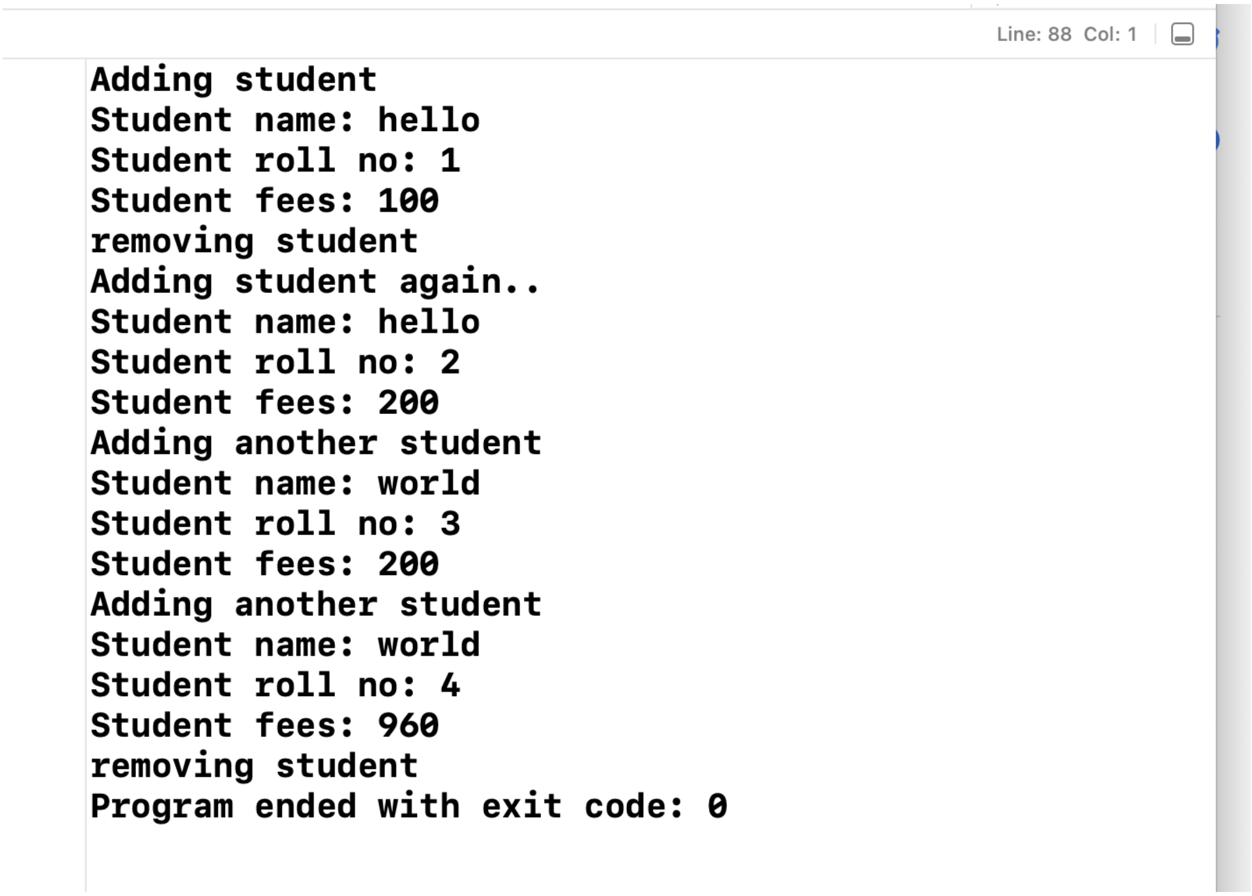
```
113
28
75
Program ended with exit code: 0
```

Question 5

create class student(Rollno, fee, Name) and create object array, add students and delete students, Rollno should be auto increment?

```
#include <iostream>

using namespace std;

static int roll = 1;

class student {

    private:
        int fee;
        char *name;
    public:
        int rollno;
        void init(int fee, char name[]);
        void print(void);

};

void student::init(int fee, char name[]) {

    student::rollno = roll;
    student::fee = fee;
    student::name = name;

}

void student::print(void) {

    cout << "Student name: " << student::name << endl;
    cout << "Student roll no: " << student::rollno << endl;
    cout << "Student fees: " << student::fee << endl;

}
```

```
void addStudent(int *l, student students[], int fee, char name[]) {

    students[*l].init(fee, name);
    roll++;
    *l = *l + 1;

}

void removeStudent(int *l, student students[], int rollno) {

    int p = -1;
    for (int i = 0; i < *l; i++) {
        if (students[i].rollno == rollno) {
            p = i;
            break;
        }
    }
    if (p != -1) {
        for (int i = p; i < *l - 1; i++) {
            students[i] = students[i+1];
        }
    }
    *l = *l - 1;

}

int main(void) {

    int max = 20;
    student students[max];

    int l = 0;
    char *x = "hello";
    char *y = "world";

    cout << "Adding student" << endl;
    addStudent(&l, students, 100, x);
    students[0].print();
    cout << "removing student" << endl;
    removeStudent(&l, students, 1);
    cout << "Adding student again.." << endl;
    addStudent(&l, students, 200, x);
    students[0].print();
    cout << "Adding another student" << endl;
    addStudent(&l, students, 200, y);
    students[1].print();
    cout << "Adding another student" << endl;
    addStudent(&l, students, 960, y);
    students[2].print();
    cout << "removing student" << endl;
    removeStudent(&l, students, 3);
}
```

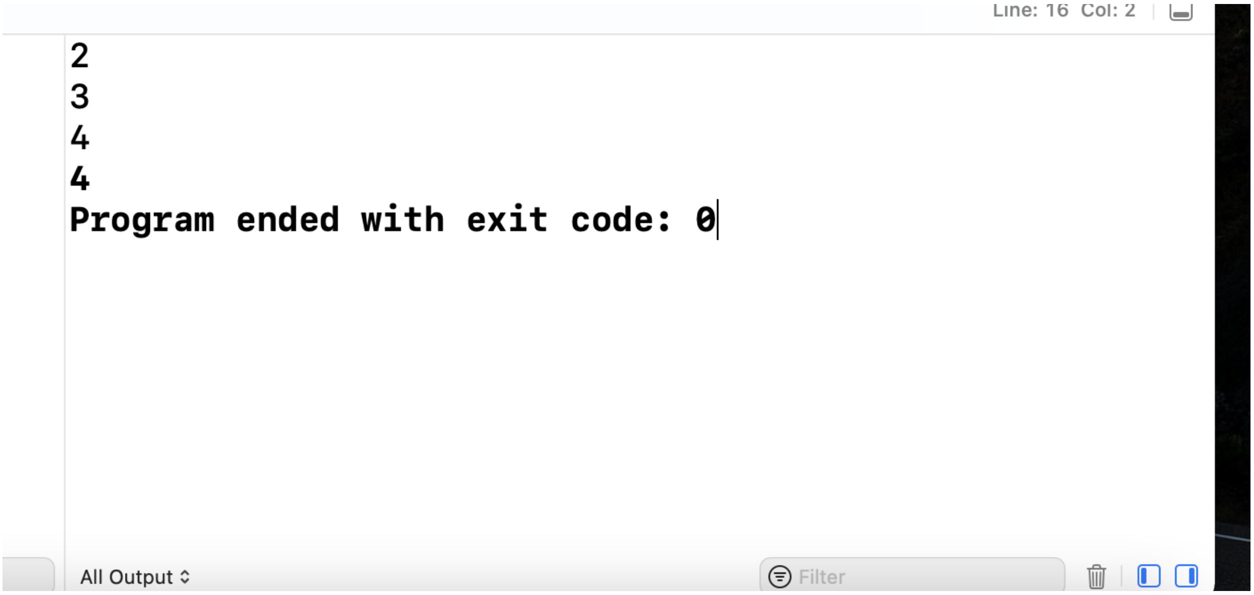


```
Adding student
Student name: hello
Student roll no: 1
Student fees: 100
removing student
Adding student again..
Student name: hello
Student roll no: 2
Student fees: 200
Adding another student
Student name: world
Student roll no: 3
Student fees: 200
Adding another student
Student name: world
Student roll no: 4
Student fees: 960
removing student
Program ended with exit code: 0
```

EXPERIMENT-1

Question-1
WAP to find out greatest number among three numbers.

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


int main() {

    int a,b,c; cin >> a >> b >> c;

    int x;
    x = (a>b)?((a>c)?a:c):((b>c)?b:c);

    cout << x << endl;

    return 0;
}
```



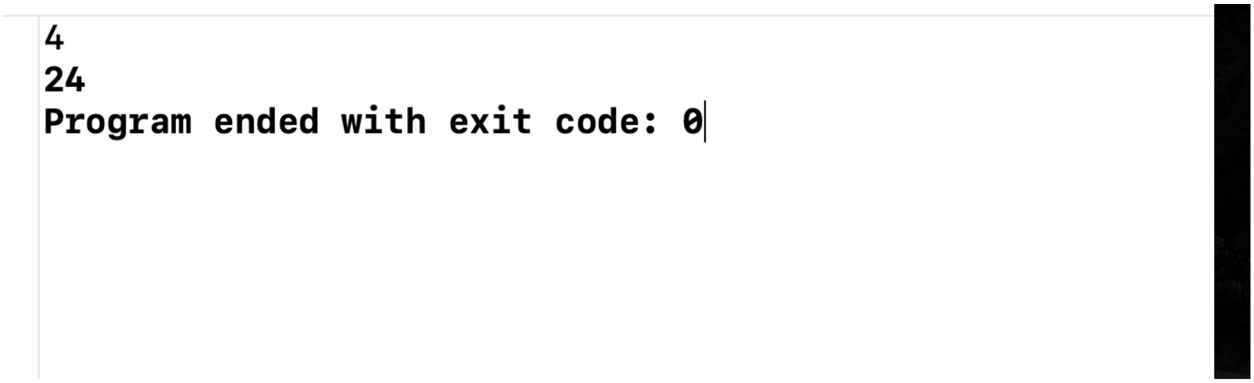
```
2
3
4
4
Program ended with exit code: 0
```

Question 2

WAP to calculate the factorial of a given number.

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

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

int main() {

    int n; cin >> n;
    cout<<fact(n)<<endl;
    return 0;
}
```

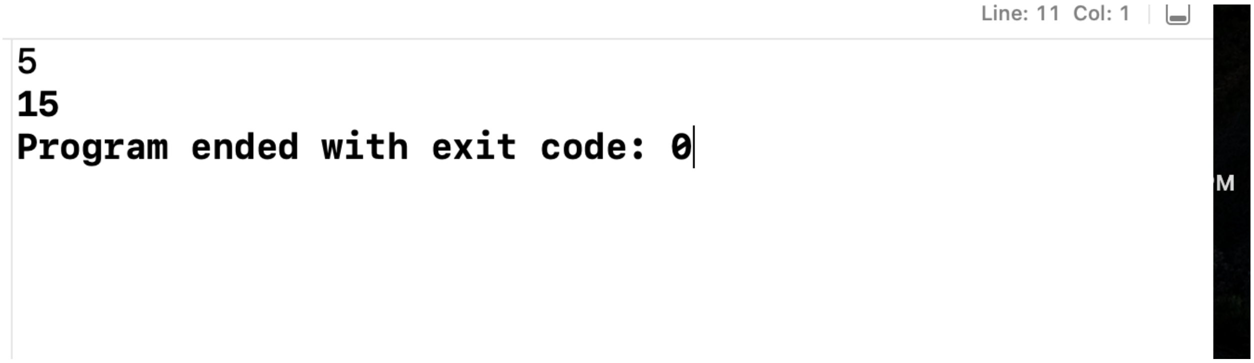


```
4
24
Program ended with exit code: 0
```

Question 3

WAP to find out the sum of first n integer numbers

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

int main() {
    int n ; cin >> n;

    cout << (n)*(n+1)/2 << endl;
    return 0;
}
```

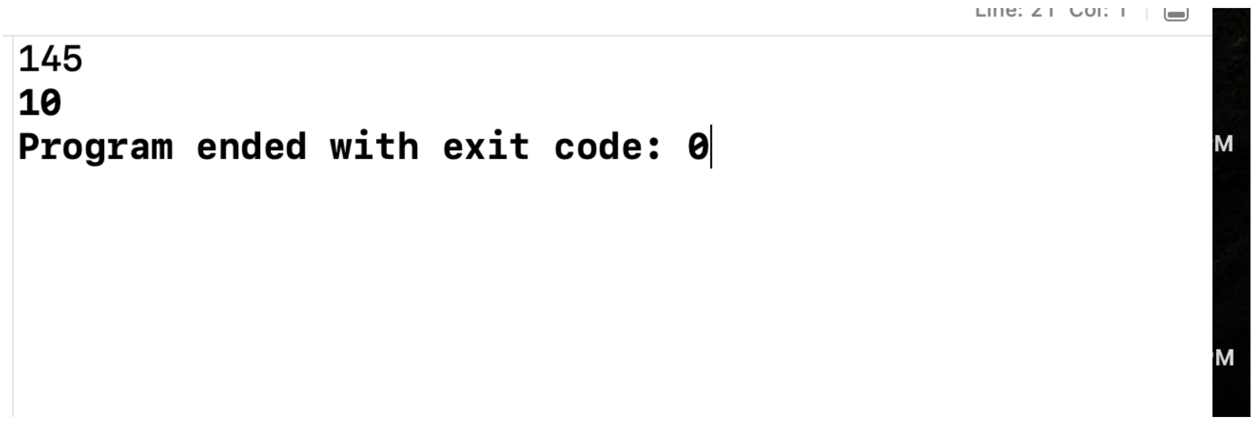


```
5
15
Program ended with exit code: 0
```

Question 4

WAP to find out the sum of digits of a number.

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


int main() {
    int n ; cin >> n;

    int sum=0;

    while(n>0){
        sum+= n%10;
        n /=10;

    }

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

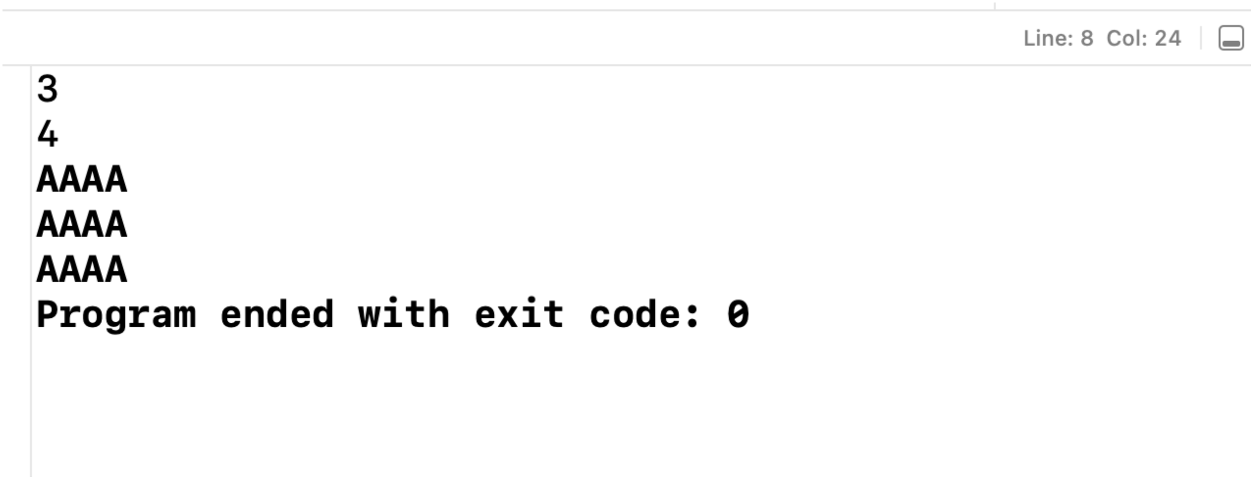


```
145
10
Program ended with exit code: 0
```

Question 5

WAP to print ‘A’ in a matrix with n and m order

```
#include <iostream>
using namespace std;
int main() {
    int n,m; cin>>n >> m;

    for(int i=0;i<n;i++){
        for(int j=0;j<m;j++){
            printf("%c",'A');
        }
        printf("\n");
    }
    return 0;
}
```



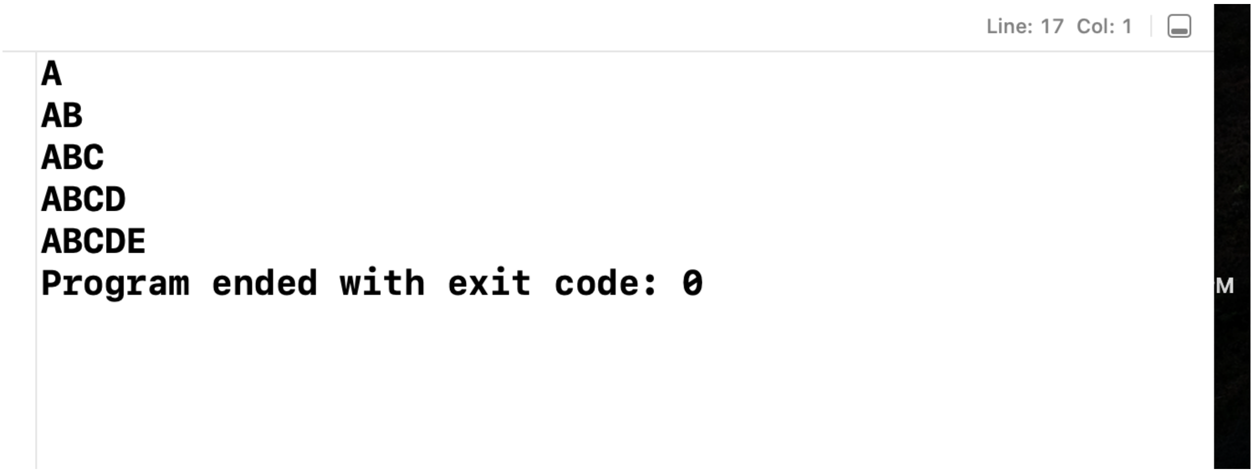
```
3
4
AAAA
AAAA
AAAA
Program ended with exit code: 0
```

Question 6

WAP to print the following pattern

A
A B
A B C
A B C D
A B C D E

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


int main() {

    for(int i=0;i<5;i++){
        for(int j=0;j<=i;j++){
            printf("%c",65+j);
        }
        printf("\n");
    }

    return 0;
}
```



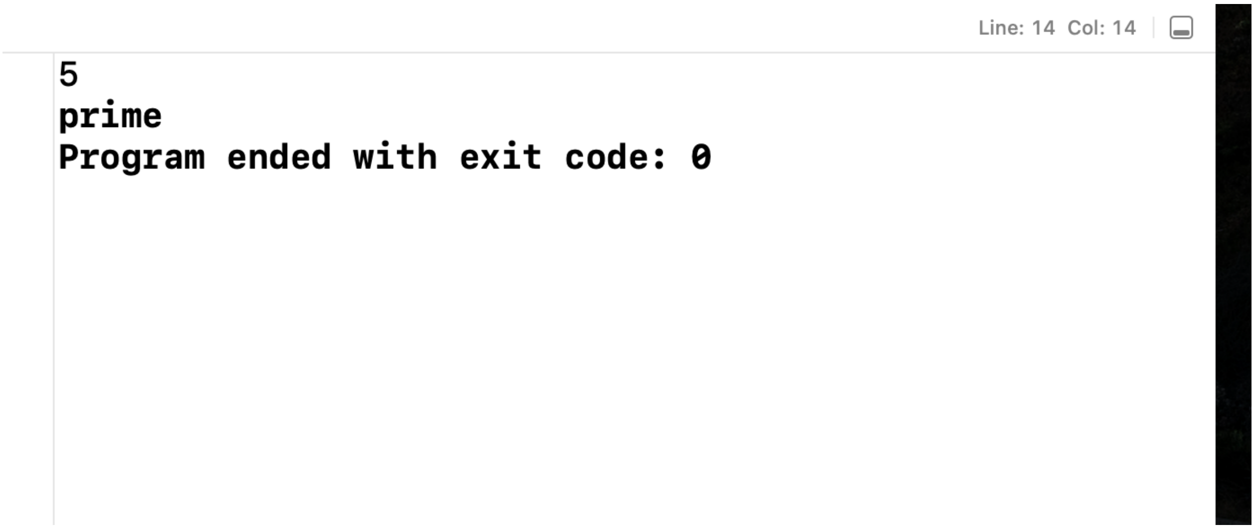
```
A
AB
ABC
ABCD
ABCDE
Program ended with exit code: 0
```

EXPERIMENT-2

Question 1

WAP to find out whether a given number is prime or not.

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

int main() {

    int n; cin >> n;
        int flag=0;
        for(int i=2;i<n/2;i++){
            if(n%i==0){
                flag=1;
                break;
            }
        }

        if(flag==1){
            cout<<"not prime"<<endl;
        }
        else{
            cout<<"prime"<<endl;
        }
        return 0;

    return 0;
}
```

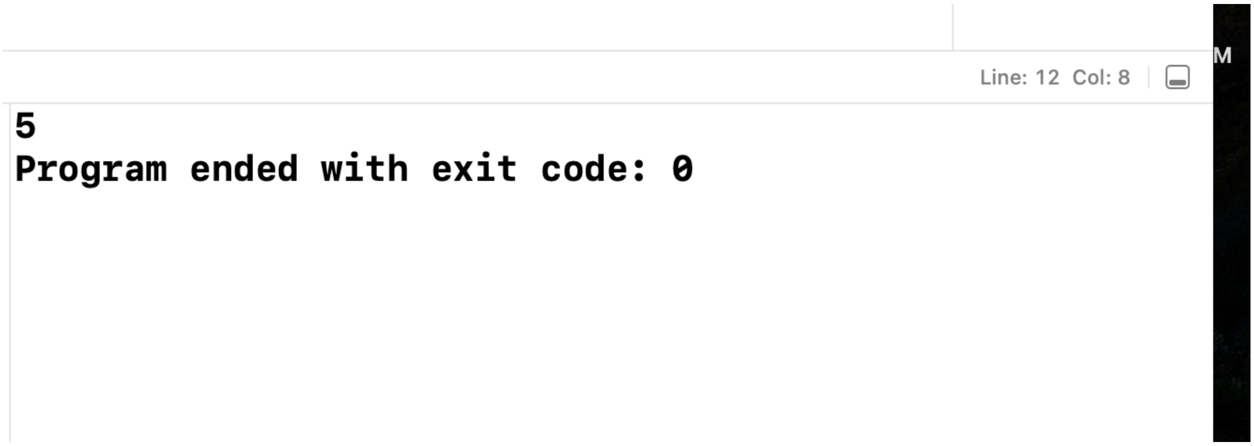


```
5
prime
Program ended with exit code: 0
```

Question 2

WAP to print the Fibonacci series.

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

int fib(int n){
    if(n==0 || n==1)
        return n;
    else
        return fib(n-1)+fib(n-2);
}

int main() {
    cout<< fib(5) << endl;
    return 0;
}
```

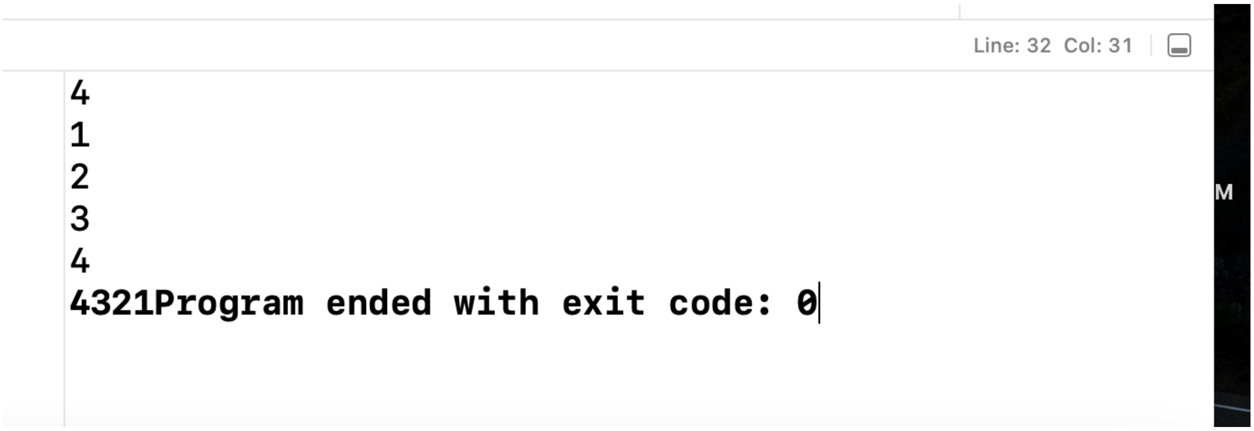


```
5
Program ended with exit code: 0
```

Question 3

WAP to print the reverse of a number.

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

void swap(int *a,int *b){
    int tmp;
    tmp=*a;
    *a=*b;
    *b=tmp;
}


int main() {
    int size;
    cin>>size;

    int num[size];

    for(int i=0;i<size;i++){
        cin>>num[i];
    }
    int j=0;
    int k=size-1;

    while(j<k){
        swap(&num[j],&num[k]);
        j++;
        k--;
    }
    for(int i=0;i<size;i++){
        cout<<num[i];
    }


    return 0;
}
```

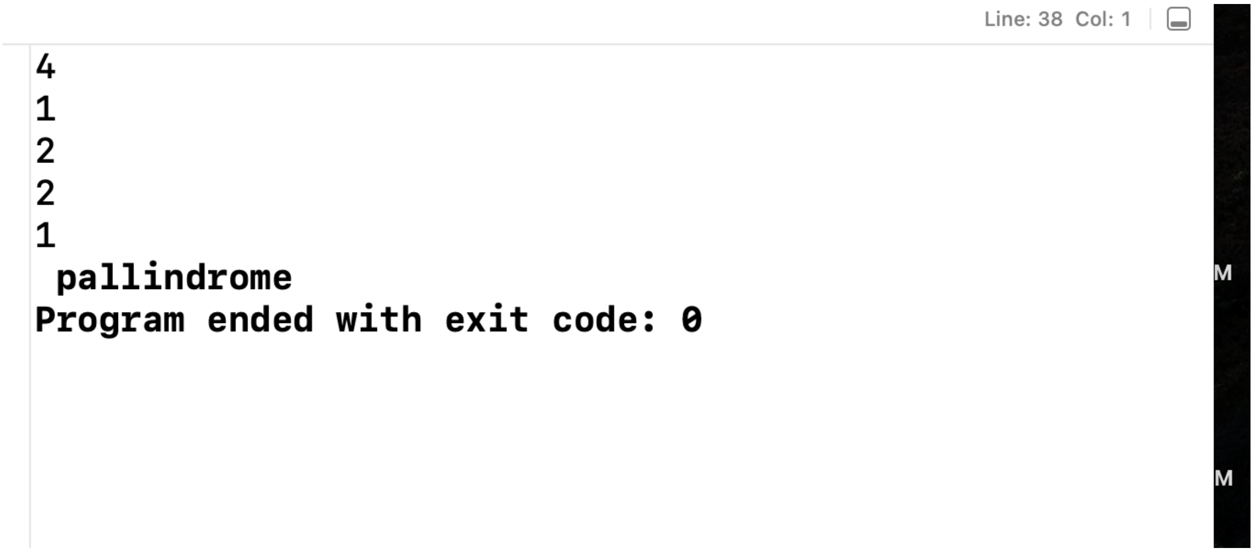


```
4
1
2
3
4
4321Program ended with exit code: 0
```

Question 4

WAP to check whether a given number is palindrome or not.

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

int main() {
    int n; cin>>n;
        int a[n];
        int flag=0;
        for(int i=0;i<n;i++){
            cin>>a[i];
        }
        int j=0;
        int k=n-1;
        while(j<k){
            if(a[j]!=a[k]){
                flag=1;
                break;
            }
            j++;
            k--;
        }

        if(flag==1){
            cout<<"Not pallindrome"<<endl;
        }
        else{
            cout<<" pallindrome"<<endl;
        }

    return 0;
}
```

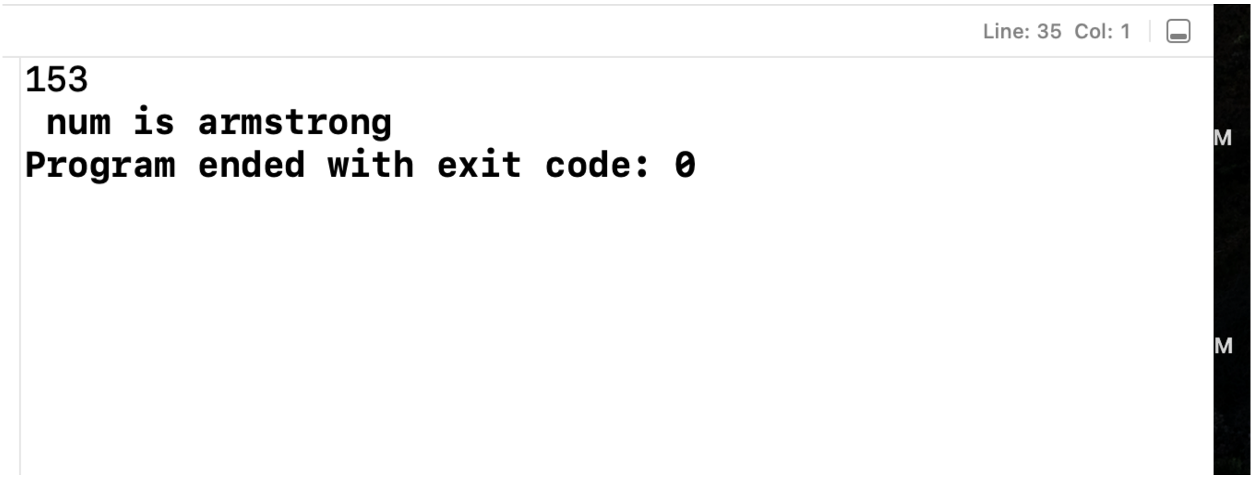


```
4
1
2
2
1
 pallindrome
Program ended with exit code: 0
```

Question 5

WAP to check whether a given number is Armstrong number or not.

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

int main() {
    int num; cin >> num;
   int  original_num=num;
    int n=0,power,result=0,remainder;

    while(original_num !=0){
        original_num /=10;
        n++;
    }
    original_num=num;
    while (original_num!=0) {
        remainder=original_num % 10;
        power = round(pow(remainder,n));
        result+=power;
        original_num/=10;

    }
    if(result==num)
        cout<<" num is armstrong"<<endl;
    else
        cout<<"num is not armstrong"<<endl;
    return 0;
}
```

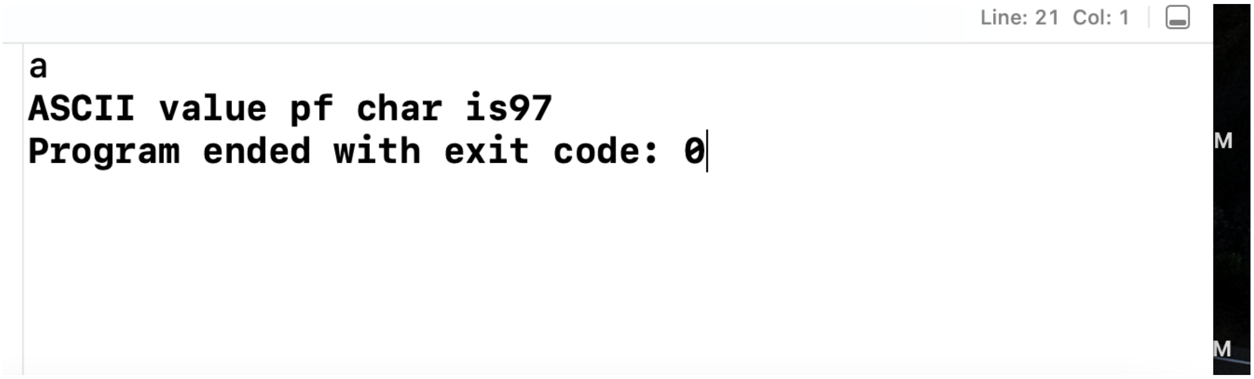


```
153
 num is armstrong
Program ended with exit code: 0
```

Question 6

WAP to print the ASCII value of a given character.

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

int main() {

    char c;

    cin>>c;

    cout << "ASCII value pf char is"<< int(c) << endl;
    return 0;
}
```

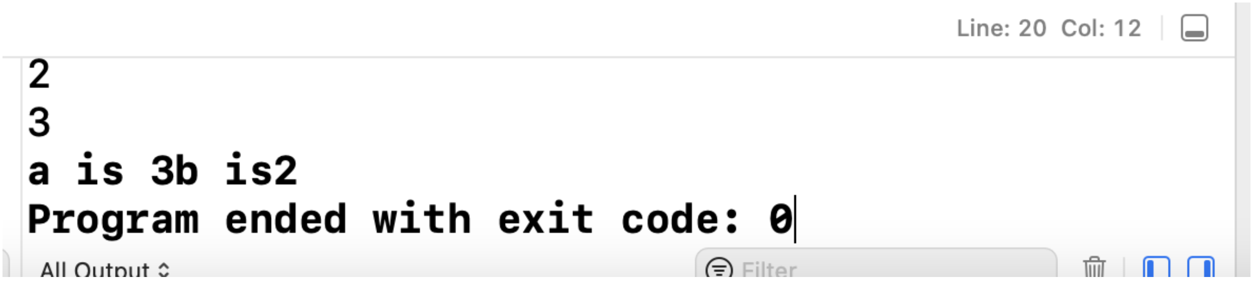


```
a
ASCII value pf char is97
Program ended with exit code: 0
```

Question 7

WAP to swap the values of two variables without using third variable and any arithmetic operator.

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

int main() {

    int a,b; cin >> a >> b;

    a = a^b;
    b = a^b;
    a = a^b;

    cout << "a is " << a << "b is" << b << endl;
    return 0;
}
```

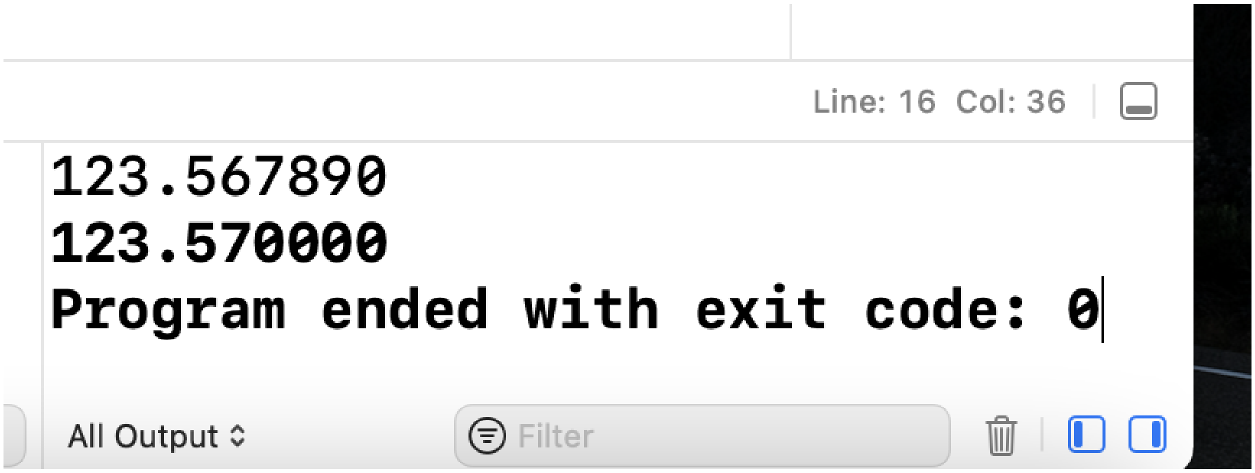


```
2
3
a is 3b is2
Program ended with exit code: 0
```

EXPERIMENT -3

Question 1
Write a program to implement a function that receives a positive floating point number and
rounds it to two decimal places. For example 127.565031 rounds to 127.570000. Print the
rounded number to six decimal places.

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

int main() {
    float num; cin >> num;

    float value= (int)(num*100+0.5);

    cout<< (float)value/100<<"0000"<<endl;
    return 0;
}
```



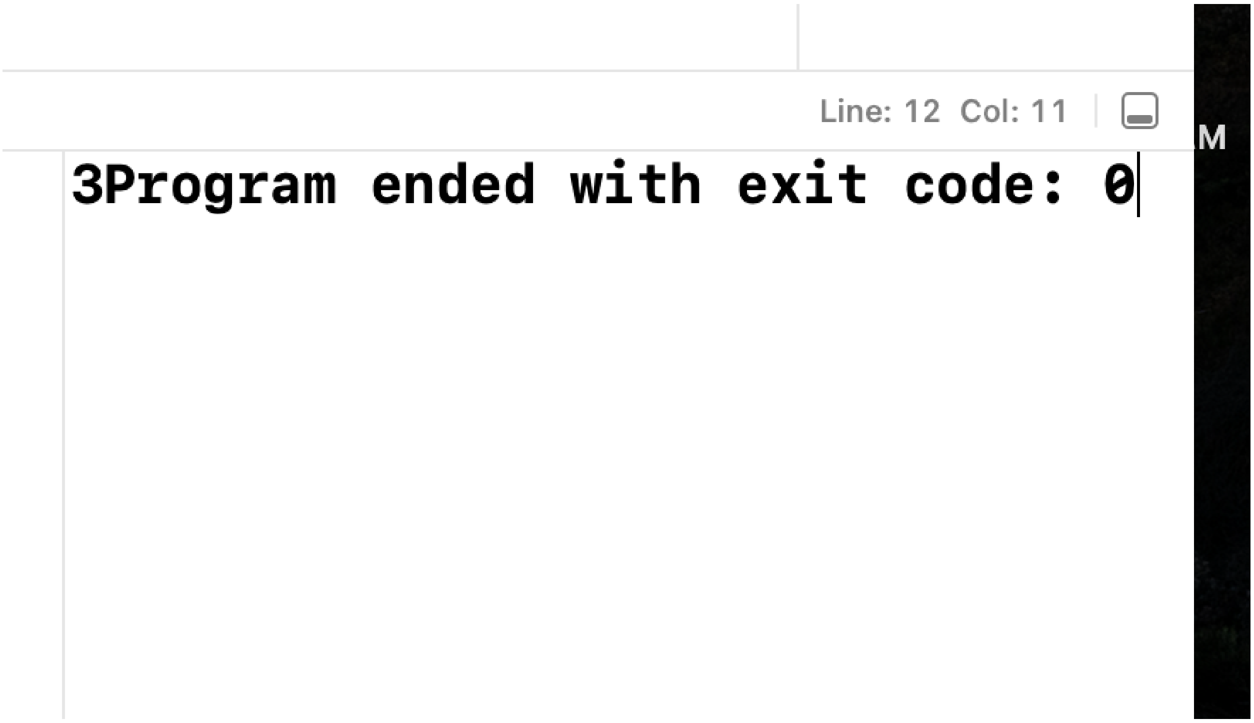
```
123.567890
123.570000
Program ended with exit code: 0
```

Question 2

WAP to generate the random number from following set without using any conditional statement.

1, 3, 9, 27, 81, 243, 729, 2187

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

int main() {
    int num=0;

    num = rand()%6;

    cout<<pow(3,num);
    return 0;
}
```

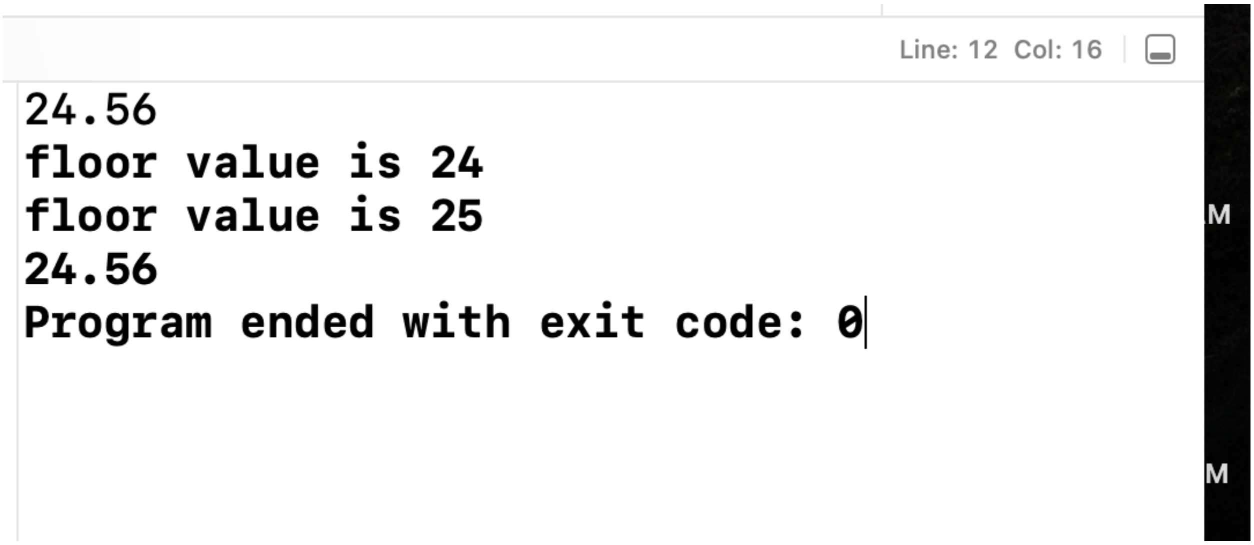


```
3Program ended with exit code: 0
```

Question 3

WAP that reads a floating-point number and prints the ceiling, floor, and rounded value.

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

int main() {
    double num;
    cin >> num;

    cout<<"floor value is "<<int(num)<<endl;
    cout<<"floor value is "<<int(num)+ 1<<endl;
    float value= (int)(num*100+0.5);

    cout<< (float)value/100<<endl;


    return 0;
}
```



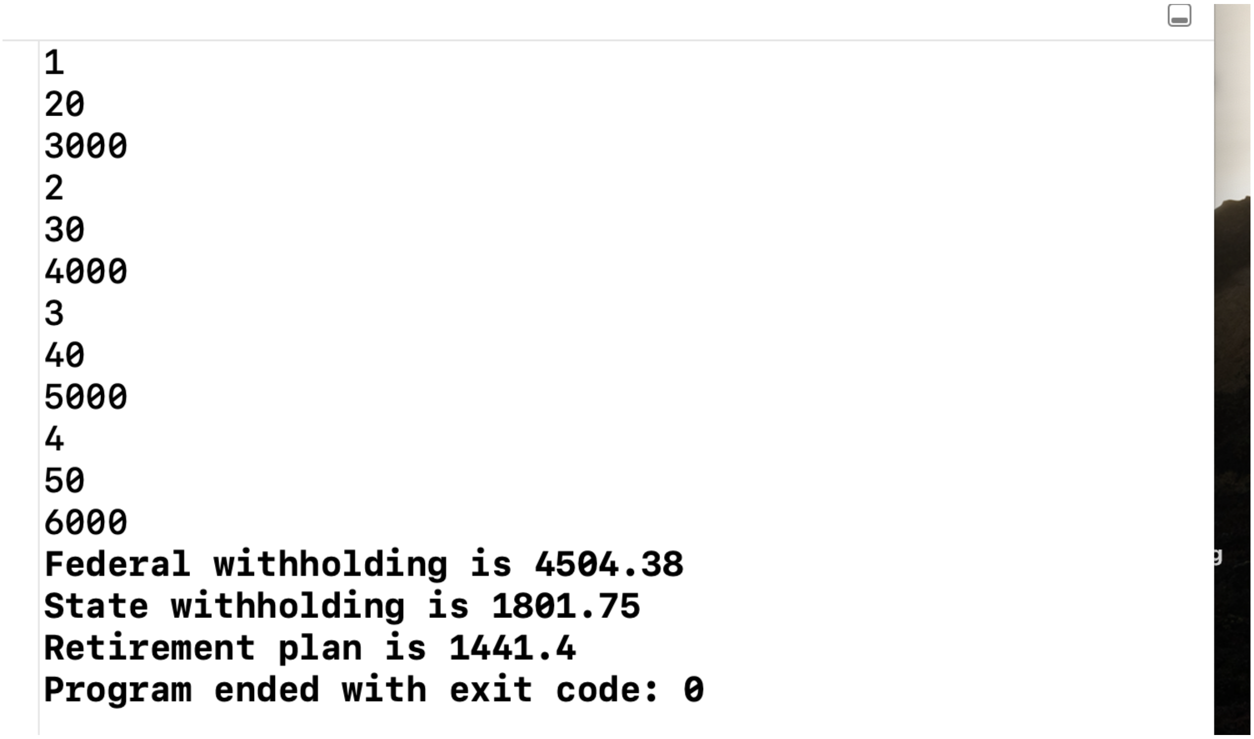
```
24.56
floor value is 24
floor value is 25
24.56
Program ended with exit code: 0
```

Question 4
Prepare a payroll earnings statement for the sales force at the Arctic Ice Company. All of
Arctic's employees are on a straight commission basis of 12.5 % of sales. Each month, they
also receive a bonus that varies depending on the profit of the month and their length of
service. The sales manager calculates the bonus separately and enters it with the

salesperson’s total sales for the month. Your program is also to calculate the withholding
taxes and retirement for the month based on the following rates:

a. Federal withholding: 25%
b. State withholding: 10%
c. Retirement plan: 8%
d. The test data to use for the program are shown in following table.

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

class Employee {
    int id, sales, bonus;
    public:
        Employee(int id, int sales, int bonus) {this->id = id; this->sales = sales; this->bonus = bonus;};
        Employee(void) {this->id = 0; this->sales = 0; this->bonus = 0;};
        void setAll(int id, int sales, int bonus) {this->id = id; this->sales = sales; this->bonus = bonus;};
        int getSales(void) { return this->sales; };
        int getBonus(void) { return this->bonus; };
        void setBonus(int b) { this->bonus = b; };
};

int main(void) {

    Employee arr[4];
    for (int i = 0; i < 4; i++) {
        int id, sales, bonus;
        cin >> id >> sales >> bonus;
        arr[i].setAll(id, sales, bonus);
    }

    float total = 0;
    for (int i = 0; i < 4; i++) {
        total += 0.125 * arr[i].getSales();
        total += arr[i].getBonus();
    }

    cout << "Federal withholding is " << (float) total / 4 << endl;
    cout << "State withholding is " << (float) total / 10 << endl;
    cout << "Retirement plan is " << (float) total * 0.08 << endl;

    return 0;

}
```



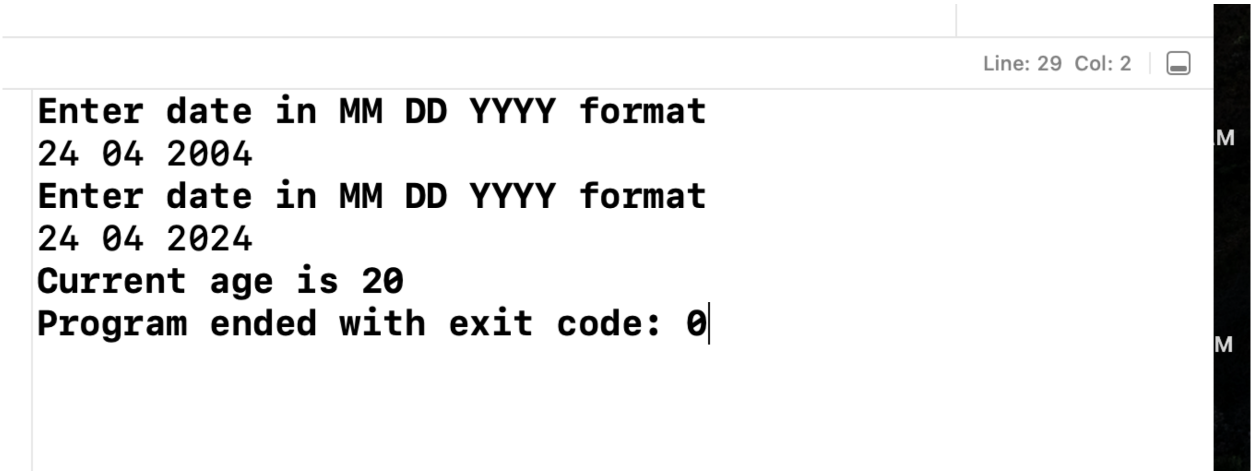
```
1
20
3000
2
30
4000
3
40
5000
4
50
6000
Federal withholding is 4504.38
State withholding is 1801.75
Retirement plan is 1441.4
Program ended with exit code: 0
```

Question 5

Write a program that asks the user to enter the current date and a person’s birth date in the form month, day, year. The program then calculates the person’s age in integral years. Use separate functions to enter the dates (pass by address), calculate the person’s age, and print

the results. Test your program with the following dates: 11/14/1957, 5/10/1989, and 1/5/2000.

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

void enterDate(int *m, int *d, int *y) {

    cout << "Enter date in MM DD YYYY format" << endl;
    cin >> *m >> *d >> *y;

}

int age(int doby, int cy) {

    return cy - doby;

}

int main(void) {

    int dobm, dobd, doby;
    int cm, cd, cy;

    enterDate(&dobm, &dobd, &doby);
    enterDate(&cm, &cd, &cy);

    cout << "Current age is " << age(doby, cy) << endl;

    return 0;

}
```



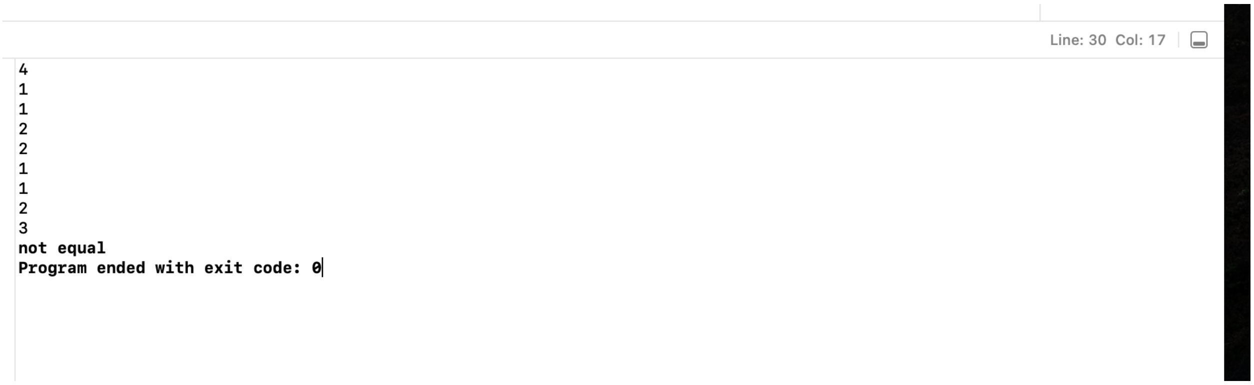
```
Enter date in MM DD YYYY format
24 04 2004
Enter date in MM DD YYYY format
24 04 2024
Current age is 20
Program ended with exit code: 0
```

Experiment-4

1. We have two arrays, A and B, each of 10 integers. Write a program that tests if every
element of array A is equal to its corresponding element in array B. In other words, the

function must check if A [0] is equal to B[0], A[1] is equal to B[1], and so forth.

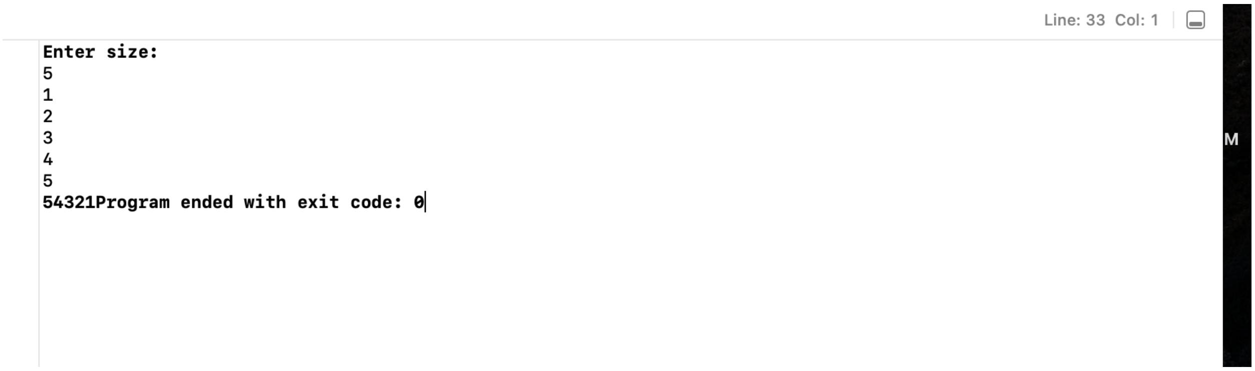
```
#include <iostream>
using namespace std;
int main() {
    int size=0;
    cin>>size;
    int a[size],b[size];
    for(int i=0;i<size;i++){
        cin>>a[i];
    }

    for(int i=0;i<size;i++){
        cin>>b[i];
    }

    bool flag=0;

    for(int i=0;i<size;i++){
        if(a[i]!=b[i]){
            flag=1;
            break;
        }
    }

    if(flag==1){
        cout<<"not equal"<<endl;
    }
    else{
        cout<<" equal"<<endl;
    }
    return 0;
}
```



```
4
1
1
2
2
1
1
2
3
not equal
Program ended with exit code: 0
```

2. Write a function that reverses the elements of an array so that the last element becomes the
first, the second from the last becomes second, and so forth. The function is to reverse the
elements in place—that is, without using another array.

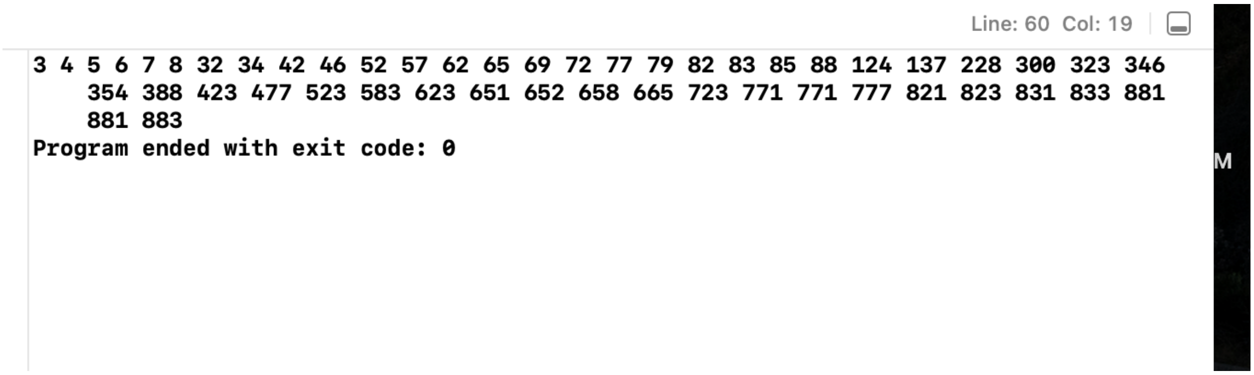
```
#include <iostream>
using namespace std;

void swap(int *a,int *b){
    int tmp;
    tmp=*a;
    *a=*b;
    *b=tmp;
}


int main() {
    int size;
    cout<<"Enter size:"<<endl;
    cin>>size;
    int a[size];

    for(int i=0;i<size;i++){
        cin>>a[i];
    }
    int j=0;
    int k=size-1;
    while (j<k) {
        swap(&a[j],&a[k]);
        j++;
        k--;
    }
    for(int i=0;i<size;i++){
        cout<<a[i];
    }
    return 0;
}
```



```
Enter size:
5
1
2
3
4
5
54321Program ended with exit code: 0
```

3. Write a program that sorts a 50-element array using the selection sort, the bubble sort, and
the insertion sort. Each sort is to be executed twice.

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

void swap(int *a , int *b){
    int *temp;
    temp=a;
    a=b;
    b=temp;
}

void selectionsort(int a[],int n){

    for(int i=0;i<n-1;i++){
       int  min=i;
        for(int j=i+1;j<n;j++){
            if(a[j]<a[min]){
                min=j;
            }
        }
        swap(&a[i],&a[min]);
    }
}

void bubbleSort(int a[], int n)
{
    int i, j;
    for (i = 0; i < n - 1; i++){
        for (j = 0; j < n - i - 1; j++){
            if (a[j] > a[j + 1])
                swap(&a[j], &a[j + 1]);
        }

    }
}

void insertionsort(int a[], int n){
    for(int i=1;i<n;i++){
        int temp=a[i];
        int j=i-1;
        while(j>=0 && a[j]>temp){
            a[j+1]=a[j];
            j--;
        }
        a[j+1]=temp;
    }
}

void dispay(int a[],int n){
    for(int i=0;i<n;i++){
        cout<<a[i]<< " ";
    }
    cout<<endl;
}

int main() {
    int a[50]={5,4,7,8,6,3,65,77,88,83
,34,124,137,228,346,354,665,477,388,583,52,42,72,82,62,32,651,771,881,831,57,46,79,85,69,300,658,777,883,833,
523,423,723,823,623,323,652,771,881,821};

    insertionsort(a, 50);
    dispay(a, 50);

    return 0;
}
```

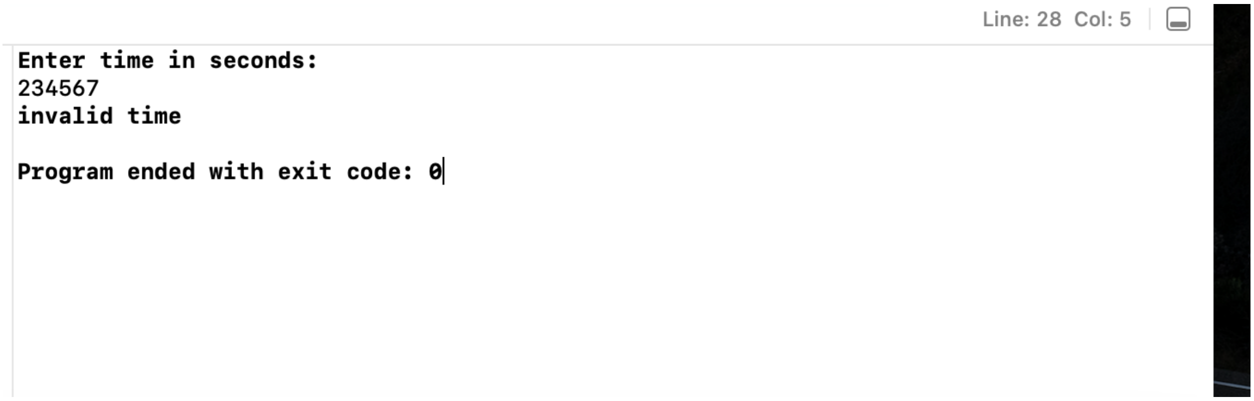


```
3 4 5 6 7 8 32 34 42 46 52 57 62 65 69 72 77 79 82 83 85 88 124 137 228 300 323 346
    354 388 423 477 523 583 623 651 652 658 665 723 771 771 777 821 823 831 833 881
    881 883
Program ended with exit code: 0
```

Experiment-5

1. Write a function using pointers that given the time in seconds passes back the time in hours,
minutes, seconds and a character indicating A.M. or P.M. If the number of seconds is more
than 24 hours, the function is to return false as an error indicator.

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

int main() {
    cout<<"Enter time in seconds:"<<endl;
    int t; cin >> t;
    string s;
    int  hrs_rem;
    if(t>86400)
        cout<<"invalid time"<<endl;
    else{
        int hrs;
        hrs=t/3600;
        hrs_rem=t%3600;
        s= s+ to_string(hrs)+" : ";

        int min;
        min=hrs_rem/60;
        int min_rem=hrs_rem%60;
        s=s+to_string(min)+" : " + to_string( min_rem);

        if(t>43200)
            s+="P.M";
        else
            s+="A.M";
    }

    cout<<s<<endl;;

    return 0;
}
```

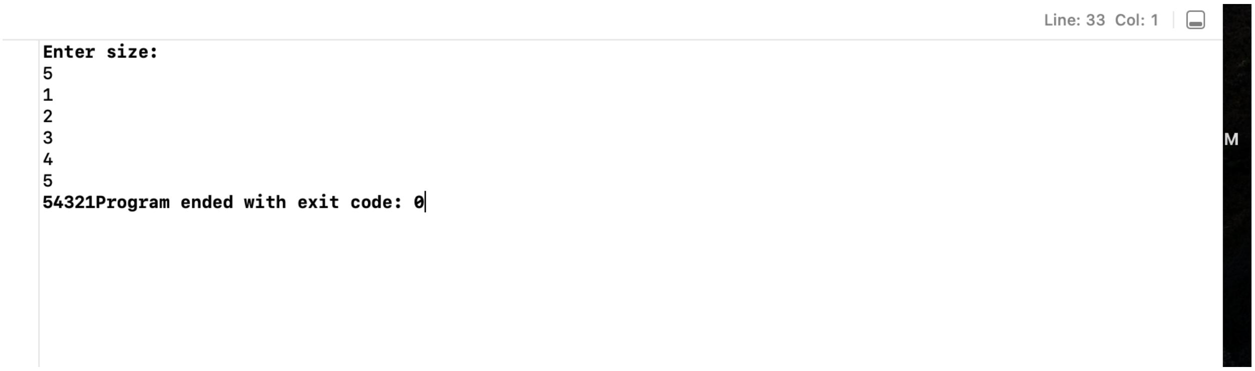


```
Enter time in seconds:
234567
invalid time

Program ended with exit code: 0
```

2. Write a function that reverses the elements of an array so that the last element becomes the

first, the second from the last becomes second, and so forth. The function must accept only
one pointer value and return void.

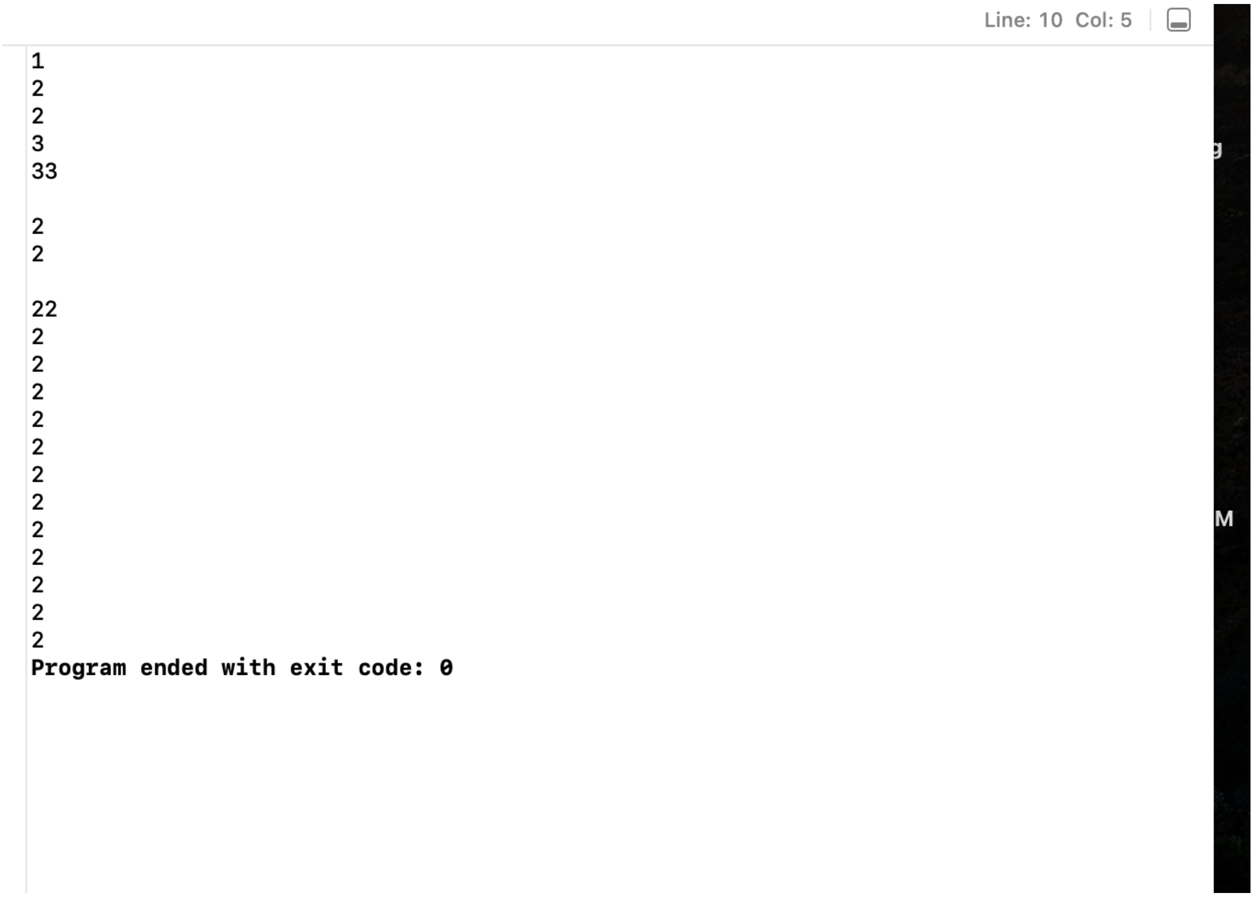
```
#include <iostream>
using namespace std;

void swap(int *a,int *b){
    int tmp;
    tmp=*a;
    *a=*b;
    *b=tmp;
}


int main() {
    int size;
    cout<<"Enter size:"<<endl;
    cin>>size;
    int a[size];

    for(int i=0;i<size;i++){
        cin>>a[i];
    }
    int j=0;
    int k=size-1;
    while (j<k) {
        swap(&a[j],&a[k]);
        j++;
        k--;
    }
    for(int i=0;i<size;i++){
        cout<<a[i];
    }
    return 0;
}
```



```
Enter size:
5
1
2
3
4
5
54321Program ended with exit code: 0
```

3. Given the following declaration and definition
int num[20];
and using only pointer notation, write a for loop to read integer values from the keyboard to
fill the array.

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

int main() {
    int num[20];

    for(int i=0;i<20;i++){
        cin>>num[i];
    }
    return 0;
}
```

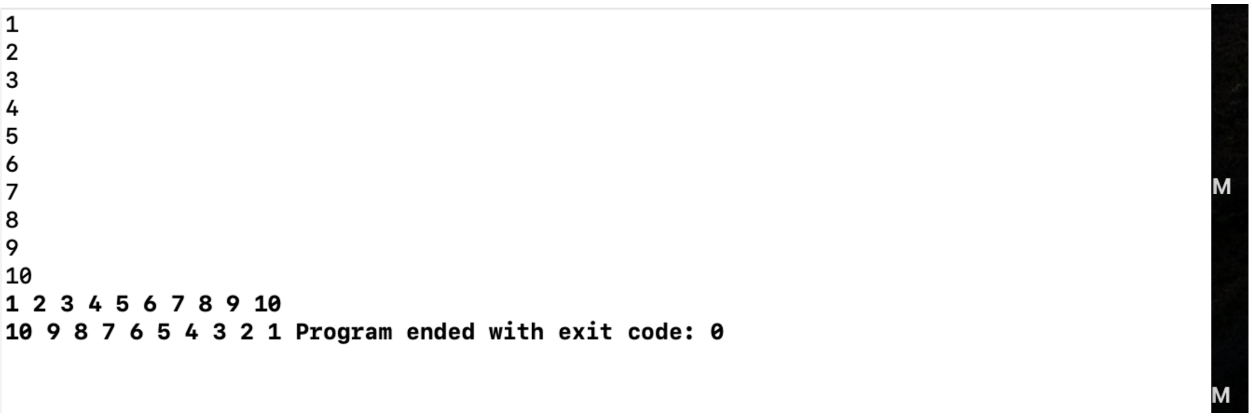


```
1
2
2
3
33

2
2

22
2
2
2
2
2
2
2
2
2
2
2
2
Program ended with exit code: 0
```

4. Write a program that will read 10 integers from the keyboard and place them in array. The program then will sort the array into ascending order and descending order and print the sorted lists. The program must not change the original array or create any other integer arrays.

The solution to this problem requires two pointer arrays. The first pointer array is rearranged so that it points to the data in ascending sequence. The second pointer array is rearranged to that it points to the data in descending sequence.

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

int main() {
    int a[10];

    for(int i=0;i<10;i++){
        cin>>a[i];
    }

    int *p[10];
    for(int i=0;i<10;i++){
        p[i]=&a[i];
    }

    //insertion sort to sort in ascending order

    for(int i=1;i<=10;i++){
        int * temp = p[i];
        int j=i-1;
        while(j>=0 && p[j]>temp){
            p[j+1]=p[j];
            j--;
        }
        p[j+1]=temp;
    }

    for(int i=0;i<10;i++){
        cout<<a[i]<<" ";
    }
    cout<<endl;
    //to print in descneding order
    for(int j=9;j>=0;j--){
        cout<<a[j]<<" ";
    }

    return 0;
}
```



```
1
2
3
4
5
6
7
8
9
10
1 2 3 4 5 6 7 8 9 10
10 9 8 7 6 5 4 3 2 1 Program ended with exit code: 0
```

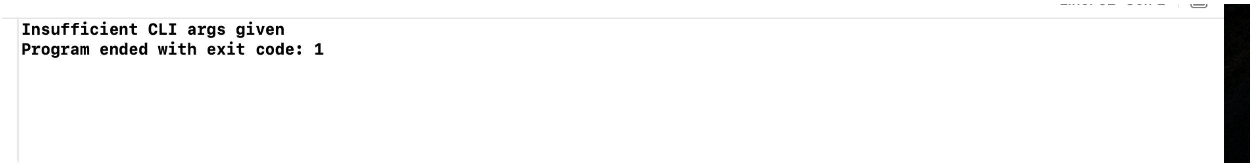
Miscellaneous questions

1.WAP to add , create , delete a new student

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

class node{
    public:
    int roll;
    char name[30];
    node*next;

    node(int r, char* s){
        roll=r;
        strcpy(name,s);
        next=NULL;
    }
};

void insert(node*&head, int r, char *s){
    if (head==NULL)
    {
        head=new node(r,s);
        return;
    }

    node*n=new node(r,s);
    node*temp=head;
    while (temp->next!=NULL)
    {
        temp=temp->next;
    }

    temp->next=n;
    return;
}

void deletenode(node*&head, int r){
    if (head==NULL)
    {
        return;
    }
    node*temp=head;
    while (temp->next->roll!=r)
    {
        temp=temp->next;
    }

    node*n=temp->next;
    temp->next=n->next;
    delete n;
}
```

```
void update(node*&head, int r){
    if (head==NULL)
    {
        return;
    }

    node*temp=head;
    while (temp->roll!=r)
    {
        temp=temp->next;
    }

    char n[20];
    cout<<"Enter the new name: \n";
    cin>>n;

    strcpy(temp->name,n);
    return;
}

void add_data(node*head){
    // int r;
    // char n[20];
    // cout<<"Enter roll number and name: \n";
    // cin>>r>>n;
    // insert(head,r,n);

    fstream file;
    file.open("check.txt", ios::in|ios::out);

    node*temp=head;

    while (temp!=NULL)
    {
        file<<"Roll:"<<temp->roll<<",name:"<<temp->name<<" ";
        temp=temp->next;
    }
    int l=file.tellp();

    char str[l];
    file.seekg(0,ios::beg);
    while (file)
    {
        file>>str;
        cout<<str<<endl;
    }

    file.close();
}

void delete_data(node*head){
    fstream file;
    file.open("check.txt", ios::in|ios::out);
    int r;
    cout<<"Enter roll number to be deleted: "<<endl;
    cin>>r;
    deletenode(head,r);
    node*temp=head;

    while (temp!=NULL)
    {
        file<<"Roll:"<<temp->roll<<",name:"<<temp->name<<" ";
        temp=temp->next;
    }

    file.close();
}

void update_data(node*head){
    fstream file;
    file.open("check.txt", ios::in|ios::out);

    int r;
    cout<<"Enter roll number to be updated: "<<endl;
    cin>>r;
    update(head,r);
    node*temp=head;

    while (temp!=NULL)
    {
        file<<"Roll:"<<temp->roll<<",name:"<<temp->name<<" ";
        temp=temp->next;
    }

    file.close();
}

void print(node*head){
    while (head!=NULL)
    {
        cout<<head->roll<<" "<<head->name<<"->";
        head=head->next;
    }
}

int main(){
    node*head=NULL;

    int r;
    char n[20];
    cin>>r>>n;
    insert(head,r,n);

    add_data(head);

    //delete_data(head);
    print(head);
```

```
return 0;
}
```



```
Insufficient CLI args given
Program ended with exit code: 1
```

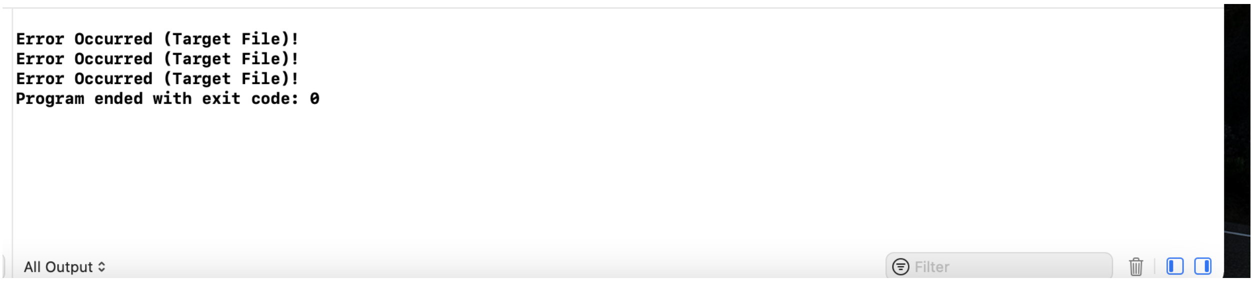
2.create a .bmp hader file

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

inline void writedata(int data, ofstream *out) {
  (*out).write((char *) &data, sizeof(data));
}

struct BMPHeader {

  int magic : 32;
  int size : 32;
  int reserved : 16;
  int reserved2 : 16;
  int offset : 32;

  void write(ofstream *out) {
    writedata(magic, out);
    writedata(size, out);
    writedata(reserved, out);
    writedata(reserved2, out);
    writedata(offset, out);
  }

};

struct DIBHeader {

  int headersize : 32;
  int imagewidth : 32;
  int imageheight : 32;
  int planes : 16;
  int bitsperpixel : 16;
  int compression : 32;
  int size : 32;
  int xppm : 32;
  int yppm : 32;
  int colors : 32;
  int impcolors : 32;

  void write(ofstream *out) {
    writedata(headersize, out);
    writedata(imageheight, out);
    writedata(imagewidth, out);
    writedata(planes, out);
    writedata(bitsperpixel, out);
    writedata(compression, out);
    writedata(size, out);
    writedata(xppm, out);
    writedata(yppm, out);
    writedata(colors, out);
    writedata(impcolors, out);
  }


};

class BMP {

  struct BMPHeader header;
  struct DIBHeader dib;
  int *pixeldata;

  public:
    BMP(int height, int width) {

      pixeldata = new int[4*width*height];

      header.magic = 0x4D42;
      header.reserved = 0;
      header.reserved2 = 0;

      dib.headersize = 40;
      dib.imagewidth = width;
      dib.imageheight = height;
      dib.bitsperpixel = 24;
      dib.planes = 1;
      dib.compression = 0;
      dib.size = 0;
      dib.colors = 1;
      dib.impcolors = 0;

      header.offset = 532;
      header.size = 54;

    }
    int writePixelsBlack(void) {
      for (int i = 0; i < sizeof(pixeldata); i++) {
        pixeldata[i] = 0;
      }
      return 0;
    }
    int writeToDisk(char *fname) {
      ofstream out;
      out.open(fname, ios::out | ios::binary);
```

```
   if (!out) {
    cout << "Writing error!" << endl;
    return 1;
   }
   header.write(&out);
   dib.write(&out);
   for (int i = 0; i < sizeof(pixeldata); i++) {
    writedata(pixeldata[i], &out);
   }
   out.close();
   return 0;
  }
  int loadFromDisk(char *fname) {
   return 0;
  }
  ~BMP(void) {
   delete [] pixeldata;
  }

};

int main(void) {

 BMP bmpfile(8,8);
 bmpfile.writePixelsBlack();
 bmpfile.writeToDisk("test.bmp");

 return 0;
}
```

3. write a program to store path of all files in a linked list

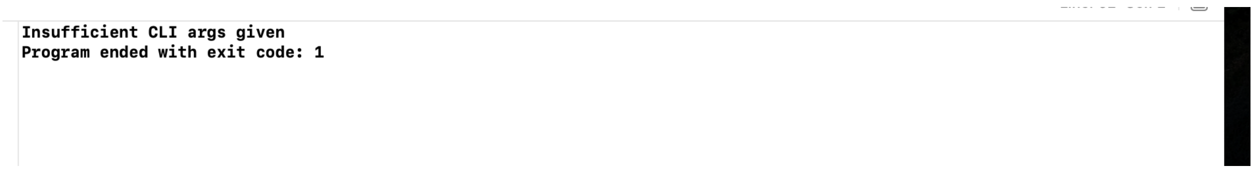
```
#include<iostream>
#include<dirent.h>
#include<fstream>
#include<stdio.h>
using namespace std;


int main()
{

 // to find total number of files in a folder/directory

  struct dirent *d;
  DIR *dr;
  dr = opendir(".");
  int count=0;
  string name[100];
  int j=0;
  if(dr!=NULL)
  {
      for(d=readdir(dr); d!=NULL; d=readdir(dr))
      {
          count++;
          name[j++]=d->d_name;
      }
      closedir(dr);
  }
  else{cout<<"\nError Occurred!";}



// to merge all files into one single file

  char fileTarget[1000], ch;
      fstream res;

  for(int i=0;i<count;i++)
  {
      fstream f1;
      f1.open(name[i], fstream::in);
      if(!f1)
      {
          cout<<"\nError Occurred (First Source File)!";
          return 0;
      }
      else
      {

          res.open(fileTarget, fstream::out);
          if(!res)
              cout<<"\nError Occurred (Target File)!";
          else
          {
              while(f1>>noskipws>>ch)
                  res<<ch;
              res<<"\n";

          }
      }
      f1.close();
  }
  res.close();

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



```
Error Occurred (Target File)!
Error Occurred (Target File)!
Error Occurred (Target File)!
Program ended with exit code: 0
```

4. Write a program to encrypt a file with the help of cli.

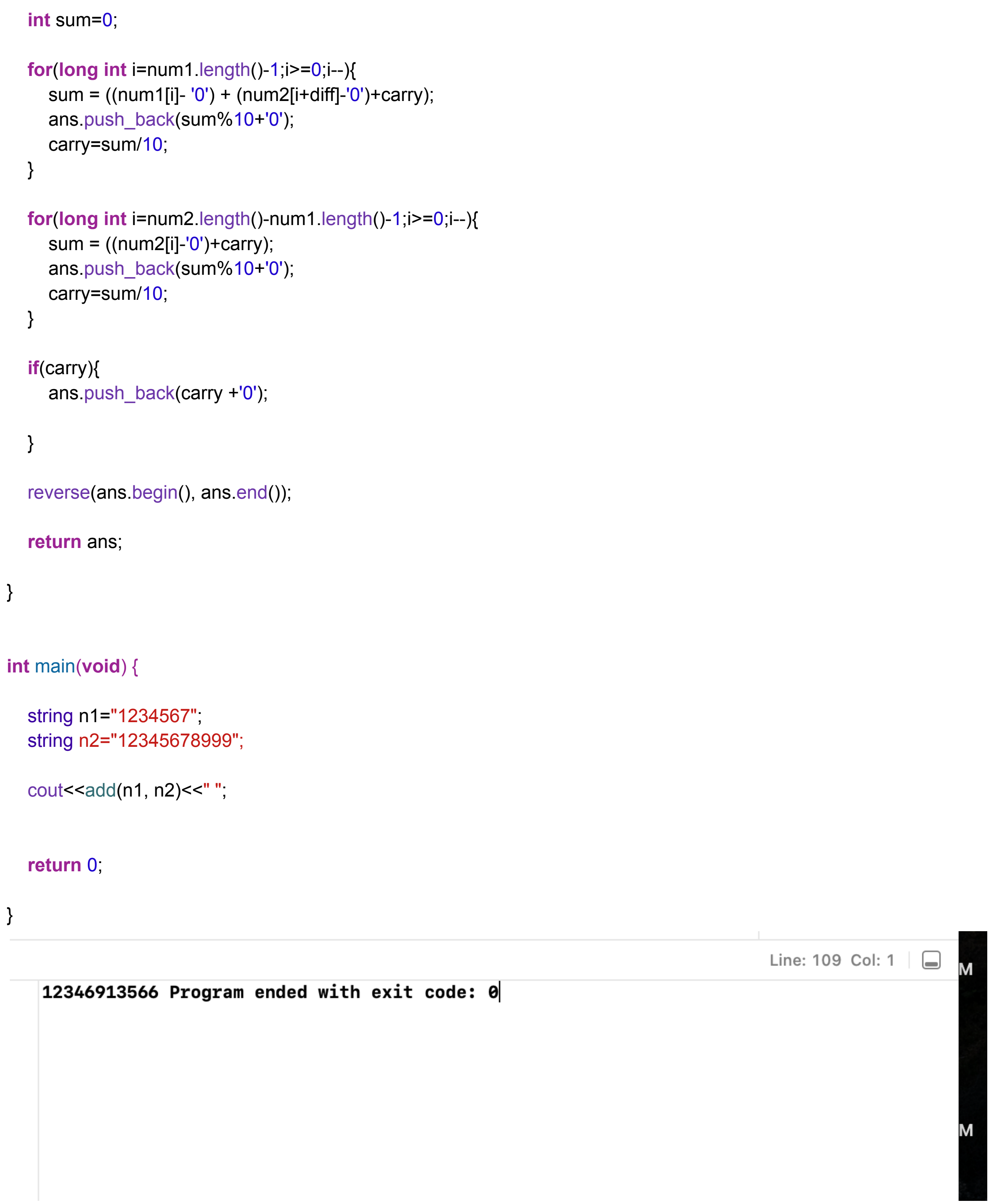
```
#include <iostream>
#include <fstream>
using namespace std;

int swapNibble(int num) {

    int firstpart = (num & 0b1111) << 4;
    int secondpart = (num & 0b11110000) >> 4;
    int newnum = firstpart + secondpart;
    return newnum;

}

void encryptFile(char *data, char *filename) {

    ofstream f(filename, ios::binary);

    for (int i = 0; data[i] != '\0'; i++) {
        int x = (int) data[i];
        x = swapNibble(x);
        char y = (char) x;
        f.write(&y, sizeof(y));
    }
    f.close();

}

void decryptFile(char *filename) {

    ifstream tempfile(filename, ios::binary | ios::ate);
    int size = tempfile.tellg();
    tempfile.close();

    ifstream f(filename, ios::binary);
    char encdata[size];
    f.read(encdata, size * sizeof(char));
    f.close();

    for (int i = 0; i < size; i++) {
        int x = (int) encdata[i];
        x = swapNibble(x);
        char y = (char) x;
        cout << y;

    }
    cout << endl;

}

int main(int argc, char *argv[]) {

    if (argc == 1) {
        cout << "Insufficient CLI args given" << endl;
        return 1;
    } else {
        encryptFile(argv[1], argv[2]);
        decryptFile(argv[2]);
    }

    return 0;

}
```



```
Insufficient CLI args given
Program ended with exit code: 1
```

5. WAP to add two long numbers

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

using namespace std;

void swap(int *a,int*b){
    int temp;
    temp=*a;
    *a=*b;
    *b=temp;
}

string add(string num1,string num2){
    if(num1.length()>num2.length()){
        swap(num1,num2);
    }

   long int diff = num2.length()-num1.length();
    int carry=0;
    string ans="";
```



```
    int sum=0;

    for(long int i=num1.length()-1;i>=0;i--){
        sum = ((num1[i]- '0') + (num2[i+diff]-'0')+carry);
        ans.push_back(sum%10+'0');
        carry=sum/10;
    }

    for(long int i=num2.length()-num1.length()-1;i>=0;i--){
        sum = ((num2[i]-'0')+carry);
        ans.push_back(sum%10+'0');
        carry=sum/10;
    }

    if(carry){
        ans.push_back(carry +'0');

    }

    reverse(ans.begin(), ans.end());

    return ans;

}


int main(void) {

    string n1="1234567";
    string n2="12345678999";

    cout<<add(n1, n2)<<" ";


    return 0;

}
```

```
12346913566 Program ended with exit code: 0
```