C LAB FILE

DEPARTMENT OF COMPUTER ENGINEERING

B-TECH(3RD SEMESTER)

C PROGRAMMING LAB(CEN-392)

SUBMITTED TO:

DR.FAIYAZ AHMED

DR WASEEM AHMED

SUBMITTED BY:

MOHD. FAIZAN

22BCS052

B-TECH(3RD SEM)

INDEX

S.NO	PROGRAMS	SIGNATURE
1	ASSIGNMENT-1	
2	ASSIGNMENT-2	
3	ASSIGNMENT-3	
4	ASSIGNMENT-4	
5	ASSIGNMENT-5	
6	ASSIGNMENT-6	
7	ASSIGNMENT-7	
8	ASSIGNMENT-8	
9	ASSIGNMENT-9	
10	ASSIGNMENT-10	
11	ASSIGNMENT-11	

12	**ASSIGNMENT-12**	
13	**ASSIGNMENT-13**	
14	**ASSIGNMENT-14**	
15	**ASSIGNMENT-15**	
16	**ASSIGNMENT-16**	
17	**ASSIGNMENT-17**	
18	**ASSIGNMENT-18**	
19	**ASSIGNMENT-19**	
20	**ASSIGNMENT-20**	
21	**ASSIGNMENT-21**	

ASSIGNMENT 1

```cpp
#include<iostream>
#include<string>
#include<sstream>
using namespace std;
int main(){
    // PROGRAM 1
    cout<<"Hello Students\n";
    // PROGRAM 2
    cout<<"Hello\nStudents\n";
    // PROGRAM 3
    cout<<"\"MySirG\"";
    cout<<endl;
    // PROGRAM 4
    int R;
    float A;
    cout<<"Enter the radius of circle:\n";
    cin>>R;
    A=3.14*R*R;
    cout<<"Area of ther circle is"<<" "<<A<<" "<<"having the radius"<<R;
    // PROGRAM 5
    string s;
    cout<<"\nEnter the string:\n";
    cin>>s;
    int count=0;
    for(int i=0;i<s.size();i++){
        count++;
    }
    printf("%d",count);
    // PROGRAM 6
    string name;
    cout<<"\nEnter the name:\n";
    cin>>name;
    cout<<"\"Hello,"<<name<<"\"";
    cout<<endl;
//    PROGRAM 7
cout<<"\"%d\""<<endl;
//    PROGRAM 8
cout<<"\"\\n\""<<endl;
//    PROGRAM 9
cout<<"\"\\"<<"\\"<<"\""<<endl;
// PROGRAM 10
    cout << "Enter a date in the format DD/MM/YYYY (e.g., 27/11/2022): ";
```

```
    string inputDate;
    cin >> inputDate;
    istringstream iss(inputDate);
    ostringstream oss;
    int day, month, year;
    char slash;
    if (iss >> day >> slash >> month >> slash >> year && slash == '/') {
        string months[] = {
            "January", "February", "March", "April", "May", "June",
            "July", "August", "September", "October", "November", "December"
        };
        if (month >= 1 && month <= 12) {
            oss << "Day-" << day << ", Month - " << months[month - 1] << ",
Year-" << year << " (Day-" << day << " Month-" << month << " Year - " << year
<< ")";
            cout << "Output format: " << oss.str() << endl;
        } else {
            cout << "Invalid month. Please enter a valid date." << endl;
        }
    } else {
        cout << "Invalid date format. Please enter a date in the format
DD/MM/YYYY." <<endl;
    }
    // PROGRAM 11
    cout << "Enter a time in the format HH:MM (e.g., 11:25): ";
    string inputTime;
    cin >> inputTime;
    istringstream iss1(inputTime);
    int hours, minutes;
    char colon;
    if (iss1 >> hours >> colon >> minutes && colon == ':') {
        cout << hours << " Hour and " << minutes << " Minute" << endl;
    } else {
        cout << "Invalid time format. Please enter a time in the format
HH:MM." << endl;
    }
    // PROGRAM 12
     int x= printf("ineuron");
     printf("%d",x);

return 0;
}
```

OUTPUT:

Hello Students

Hello

Students

"MySirG"

Enter the radius of circle:

5

Area of ther circle is 78.5 having the radius5

Enter the string:

saad

4

Enter the name:

saqib

"Hello,saqib"

"%d"

"\n"

"\\"

Enter a date in the format DD/MM/YYYY (e.g., 27/11/2022): 17/11/2023

Output format: Day-17, Month - November, Year-2023 (Day-17 Month-11 Year - 2023)

Enter a time in the format HH:MM (e.g., 11:25): 2:40

2 Hour and 40 Minute

ineuron7

ASSIGNMENT 2

```cpp
#include<iostream>
using namespace std;
int main(){
// PROGRAM 1
int num,num1;
cout<<"Enter the number:\n";
cin>>num;
num=num%10;
cout<<"unit digit is:"<<num<<endl;
// PROGRAM 2
cout<<"Enter the no.:\n";
cin>>num1;
num1=num1-num1%10;
num=num1/10;
cout<<"Number without last digit is:\n"<<num<<endl;
// PROGRam 3
int a,b;
cout<<"enter value of a,b\n";
cin>>a>>b;
int temp;
temp=a;
a=b;
b=temp;
cout<<"value after swap is:\n"<<a<<" "<<b<<endl;
// PROGRAM 4
int c,d;
cout<<"enter value of c,d\n";
cin>>c>>d;
c=c+d;
d=c-d;
c=c-d;
cout<<"value after swap is:\n"<<c<<" "<<d<<endl;
// PROGRAM 5
int n,dig;
int sum=0;
cout<<"Enter three digit no.\n";
cin>>n;
while(n!=0){
    dig=n%10;
    sum=sum+dig;
    n=n/10;
}
```

```
cout<<"sum of digits is:\n"<<sum<<endl;
// PROGRAM 6
char ch;
cout<<"Enter a character:\n";
cin>>ch;
cout<<"ascii code is:"<<int(ch)<<endl;
// PROGRAM 7
int no;
int pos=1;
cout<<"Enter the no.\n";
cin>>no;
if(no==0){
    return 0;
}
 while (!(no & 1)) {
    no>>=1;
    pos++;
 }
 cout<<""<<pos<<endl;
// PROGRAM 8
int no1;
cout<<"Enter the no.\n";
cin>>no1;
if (no1 & 1) {
        cout << no1 << " is an odd number." << std::endl;
    } else {
        cout << no1 << " is an even number." << std::endl;
    }
// PROGRAM 9
int a2;
float b2;
char c2;
double d2;
cout<<"size of int,float,char and double is:";
cout<<sizeof(a2)<<sizeof(b2)<<sizeof(c2)<<sizeof(d2);
// PROGRAM 10
int num3,digit;
cout<<"enter the no.\n";
cin>>num3;
digit=num3%10;
num3=num3-digit;
cout<<"after making last digit 0 ,the number becomes"<<num3;
cout<<endl;
// PROGRAM 11
int num4,digit1;
```

```
cout<<"enter the no.and digit\n";
cin>>num4>>digit1;
num4=num4*10+digit1;
cout<<"after appending last digit,the number becomes"<<num4;
cout<<endl;
// PROGRAM 12
float inr,usd;
cout<<"Enter price in inr:\n";
cin>>inr;
usd=inr/float(76.23);
cout<<"price in usd is:"<<usd<<endl;
// PROGRAM 13
int number;
cout<<"Enter 3-digit no.\n";
cin>>number;
 int dig1 = number % 10;
    number /= 10;
    int dig2 = number % 10;
    number /= 10;
    int dig3 = number;
    cout<<"the no. after rotation is:";
    cout<< (dig1* 100) + (dig3 * 10) + dig2;
    cout<<endl;
    return 0;
}
```

OUTPUT:

Enter the number:

12

unit digit is:2

Enter the no.:

23

Number without last digit is:

2

enter value of a,b

12 23

value after swap is:

23 12

enter value of c,d

23 45

value after swap is:

45 23

Enter three digit no.

123

sum of digits is:

6

Enter a character:

s

ascii code is:115

Enter the no.

64

7

Enter the no.

45

45 is an odd number.

size of int,float,char and double is:4418enter the no.

5

after making last digit 0 ,the number becomes0

enter the no.and digit

345

5

after appending last digit,the number becomes3455

Enter price in inr:

500

price in usd is:6.5591

Enter 3-digit no.

390

the no. after rotation is:39

ASSIGNMENT 3

```
#include<iostream>
using namespace std;
int main(){
    // PROGRAM 1
    int num;
    cout<<"Enter a no:\n";
    cin>>num;
    if(num>0){
        cout<<num<<" "<<"is positive no.\n";
    }
    else{
         cout<<num<<" "<<"is non-positive no.\n";
    }
    // PROGRAM 2
    if(num%5==0){
         cout<<num<<" "<<"is divisible by 5.\n";
    }
    else{
         cout<<num<<" "<<"is not divisible by 5.\n";
    }
    // PROGRAM 3
      if(num%2==0){
         cout<<num<<" "<<"is even.\n";
```

```
    }
    else{
         cout<<num<<" "<<"is odd\n";
    }
    // PROGRAM 4
    if((num/2)*2==num){
         cout<<num<<" "<<"is even.\n";
    }
    else{
         cout<<num<<" "<<"is odd\n";
    }
     // PROGRAM 5
    if(num>=100&& num<=999){
         cout<<num<<" "<<"is 3-digit no.\n";
    }
    else{
         cout<<num<<" "<<"not a 3-digit no.\n";
    }
     // PROGRAM 6
     int a,b;
     cout<<"Enter two no.:\n";
     cin>>a>>b;
     if(a>b){
        cout<<a<<"is greater \n";
     }
     if(a==b){
         cout<<a<<" \n";
     }
     else{
         cout<<b<<"is greater \n";
     }
        // PROGRAM 7
        int D,a1,b1,c;
        cout<<"Enter vaue of a1,b1,c for D:\n";
        cin>>a1>>b1>>c;
        D=b1*b1-4*a1*c;
        if(D>0){
            cout<<"Roots are real and distinct\n";
        }
        else if(D==0){
              cout<<"Roots are real and equal\n";
        }
        else{
            cout<<"Roots are imaginary\n";
        }
```

```
    // PROGRAM 8
        int year;
        cout<<"enter a year:\n";
        cin>>year;
        if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {
        cout << year << " is a leap year." <<endl;
    }  else {
        cout << year << " is not a leap year."<<endl;
    }
    // PROGRAM 9
    int p,q,r;
    cout<<"enter three no.";
    cin>>p>>q>>r;
    if(p>q&&p>r){
        cout<<p<<"is greatest \n";
    }
    else if(q>p&&q>r){
         cout<<q<<"is greatest \n";
    }
    else{
         cout<<r<<"is greatest \n";
    }
//   PROGRAM 10
  int sp,cp;
  cout<<"Enter the cost price and selling price:\n";
  cin>>cp>>sp;
  float at=(sp-cp)*100/cp;
  if(a>0){
     cout<<"profit percentage is:\n"<<at;
  }
  else{
      cout<<"loss percentage is:\n"<<at;
  }
// PROGRAM 11
int sub1,sub2,sub3,sub4,sub5;
float sumsub=0;
cout<<"Enter the marks of five subjects:\n";
cin>>sub1>>sub2>>sub3>>sub4>>sub5;
sumsub=(sub1+sub2+sub3+sub4+sub5)/5;
if(sumsub>33){
     cout<<"the candidate is passed:\n";
}
else{
     cout<<"the candidate is failed:\n";
}
```

```
// PROGRAM 12
    char alphabet;

    cout << "Enter an alphabet: ";
    cin >> alphabet;

    if (isalpha(alphabet)) {
        if (islower(alphabet)) {
            cout << "The alphabet is in lowercase.";
        } else if (isupper(alphabet)) {
            cout << "The alphabet is in uppercase.";
        }
    } else {
        cout << "Invalid input. Please enter an alphabet.";
    }
// PROGRAM 13
int no;
cout<<"enter the no.\n";
cin>>no;
if(no%2==0){
     cout<<"no. is divisible by 2\n";
}
else if(no%2==0 &&no%3==0){
          cout<<"no. is divisible by both 2 and 3\n";
}
if(no%3==0){
          cout<<"no. is divisible by 3\n";
}
// PROGRAM 14
if(no%7==0){
     cout<<"no. is divisible by 7\n";
}
else if(no%7==0 &&no%3==0){
          cout<<"no. is divisible by both 7 and 3\n";
}
if(no%3==0){
          cout<<"no. is divisible by 3\n";
}
// PROGRAM 15
  if(no>0){
        cout<<no<<" "<<"is positive no.\n";
    }
    else if(no==0)
    { cout<<no<<" "<<"is zero\n";
```

```
    }
    else{
     cout<<no<<" "<<"is negative no.\n";
     }
// PROGRAM 17
int side1,side2,side3;
cout<<"enter the sides of triangle:\n";
cin>>side1>>side2>>side3;
if(side1+side2>side3||side2+side3>side1||side1+side3>side2){
     cout<<"the triangle is valid:";
}
else {
     cout<<"the triangle is not valid:";
}
// PROGRAM 18
int month;
cout<<"Enter the month:\n";
cin>>month;
if(month==4||month==6||month==9||month==11){
     cout<<"no of days is 30";
}
else if(month==2){
     cout<<"no of days is 28";
}
else{
     cout<<"no of days is 31";
}
    return 0;
}
```

OUTPUT:

Enter a no:

45

45 is positive no.

45 is divisible by 5.

45 is odd

45 is odd

45 not a 3-digit no.

Enter two no.:

34 68

68is greater

Enter vaue of a1,b1,c for D:

2 3 4

Roots are imaginary

enter a year:

2345

2345 is not a leap year.

enter three no.23 45 56

56is greatest

Enter the cost price and selling price:

23 45

profit percentage is:

95Enter the marks of five subjects:

12 23 34 45 56

the candidate is passed:

Enter an alphabet: s

The alphabet is in lowercase.enter the no.

234

no. is divisible by 2

no. is divisible by 3

no. is divisible by 3

234 is positive no.

Enter a character: w

Lowercase alphabet.enter the sides of triangle:

23 34 45

the triangle is valid:Enter the month:

2

no of days is 28

ASSIGNMENT 4

```
#include<iostream>
using namespace std;
int main(){
    // PROGRAM 1
    for(int i=0;i<5;i++){
        cout<<"MySirG"<<endl;
    }
    // PROGRAM 2
     for(int i=1;i<=10;i++){
        cout<<i<<" ";
    }
    cout<<endl;
      // PROGRAM 3
      for(int i=10;i>=1;i--){
        cout<<i<<" ";
    }

    cout<<endl;
      // PROGRAM 4
      for(int i=1;i<20;i++){
        if(i%2!=0){
        cout<<i<<" ";
        }
    }
    cout<<endl;
      // PROGRAM 5
      for(int i=20;i>=1;i--){
```

```
        if(i%2!=0){
        cout<<i<<" ";
        }
    }
    cout<<endl;
      // PROGRAM 6
      for(int i=1;i<20;i++){
        if(i%2==0){
        cout<<i<<" ";
        }
    }
    cout<<endl;
      // PROGRAM 7
      for(int i=20;i>=1;i--){
        if(i%2==0){
        cout<<i<<" ";
        }
    }
     cout<<endl;
       // PROGRAM 8
       for(int i=1;i<=10;i++){
        cout<<i*i<<" ";
    }
     cout<<endl;
       // PROGRAM 9
       for(int i=1;i<=10;i++){
        cout<<i*i*i<<" ";
    }
    cout<<endl;
      // PROGRAM 10
       for(int i=1;i<=10;i++){
        cout<<"5 X "<<i<<"="<<5*i<<endl;
    }
    return 0;

}
```

OUTPUT:

MySirG

MySirG

MySirG

MySirG

MySirG

1 2 3 4 5 6 7 8 9 10

10 9 8 7 6 5 4 3 2 1

1 3 5 7 9 11 13 15 17 19

19 17 15 13 11 9 7 5 3 1

2 4 6 8 10 12 14 16 18

20 18 16 14 12 10 8 6 4 2

1 4 9 16 25 36 49 64 81 100

1 8 27 64 125 216 343 512 729 1000

5 X 1=5

5 X 2=10

5 X 3=15

5 X 4=20

5 X 5=25

5 X 6=30

5 X 7=35

5 X 8=40

5 X 9=45

5 X 10=50

ASSIGNMENT 5

```cpp
#include<iostream>
using namespace std;
int main(){
    // PROGRAM 1
    int n;
    cout<<"Enter the value of n:\n";
    cin>>n;
    for(int i=0;i<n;i++){
        cout<<"MySirG"<<endl;
    }
    // PROGRAM 2
     for(int i=1;i<=n;i++){
        cout<<i<<" ";
    }
    cout<<endl;
      // PROGRAM 3
      for(int i=n;i>=1;i--){
        cout<<i<<" ";
    }

    cout<<endl;
      // PROGRAM 4
      for(int i=1;i<2*n;i++){
        if(i%2!=0){
        cout<<i<<" ";
        }
    }
    cout<<endl;
      // PROGRAM 5
      for(int i=2*n;i>=1;i--){
        if(i%2!=0){
        cout<<i<<" ";
        }
    }
    cout<<endl;
      // PROGRAM 6
      for(int i=1;i<=2*n;i++){
        if(i%2==0){
        cout<<i<<" ";
        }
    }
    cout<<endl;
```

```
      // PROGRAM 7
      for(int i=2*n;i>=1;i--){
        if(i%2==0){
        cout<<i<<" ";
        }
    }
     cout<<endl;
       // PROGRAM 8
       for(int i=1;i<=n;i++){
        cout<<i*i<<" ";
    }
     cout<<endl;
       // PROGRAM 9
       for(int i=1;i<=n;i++){
        cout<<i*i*i<<" ";
    }
    cout<<endl;
      // PROGRAM 10
       for(int i=1;i<=n;i++){
         cout<<"5 X "<<i<<"="<<5*i<<endl;

    }

    return 0;

}
```

OUTPUT:

Enter the value of n:

2

MySirG

MySirG

1 2

2 1

1 3

3 1

2 4

4 2

1 4

1 8

5 X 1=5

5 X 2=10

ASSIGNMENT 6

```
#include<iostream>
using namespace std;
int main(){
// PROGRAM 1
    int n;
    cout<<"Enter the value of n:\n";
    cin>>n;
    int sum=0;
    for(int i=0;i<=n;i++){
        sum=sum+i;
    }
    cout<<"sum of first "<<n<<" natural no. is:\n"<<sum<<endl;
    // PROGRAM 2
    int sum1=0;
     for(int i=1;i<=2*n;i++){
        if(i%2==0){
            sum1=sum1+i;
        }
    }
     cout<<"sum of first "<<n<<"even natural no. is:\n"<<sum1<<endl;
 // PROGRAM 3
    int sum2=0;
     for(int i=1;i<=2*n;i++){
        if(i%2!=0){
            sum2=sum2+i;
        }
    }
     cout<<"sum of first "<<n<<"odd natural no. is:\n"<<sum2<<endl;
 // PROGRAM 4
```

```
    int sum3=0;
     for(int i=1;i<=n;i++){
            sum3=sum3+i*i;
    }
     cout<<"sum of squares of first "<<n<<" natural no. is:\n"<<sum3<<endl;
 // PROGRAM 5
    int sum4=0;
     for(int i=1;i<=n;i++){
            sum4=sum4+i*i*i;
    }
     cout<<"sum of cubes of first "<<n<<" natural no. is:\n"<<sum4<<endl;
 // PROGRAM 6
    int fact=1;
     for(int i=1;i<=n;i++){
            fact=fact*i;
    }
     cout<<"Factorial of "<<n<<"is:\n"<<fact<<endl;
// PROGRAM 7
int number;
cout<<"Enter the number:";
cin>>number;
int count=0;
while(number!=0){
if(number%10!=0){
   count++;
}
number=number/10;

}
cout<<"the no.of digits is "<<count<<endl;
// PROGRAM 8
int number1;
cout<<"enter the no.";
cin>>number1;
for(int i=2;i<number1;i++){
    if(number1%i==0){
         cout<<"the number  is not  prime";
          break;
    }
else{
       cout<<"the number is prime";
       break;
}
}

//   PROGRAM 9
```

```
int n1,n2;
cout<<"\nenter two no."<<endl;
cin>>n1>>n2;
int var=(n1>n2)?n1:n2;
int lcm=var;
while(true){
    if(lcm%n1==0 &&lcm%n2==0){
        cout<<"lcm is"<<lcm;
        break;
    }
    else{
    lcm=lcm+var;
    cout<<"lcm is"<<lcm;
    break;
}
}
// PROGRAM 10
int n3;
int rev=0;
cout<<"\n Enter the no."<<endl;
cin>>n3;
while(n3>0){
    int lastdig=n3%10;
    rev=rev*10+lastdig;
    n3=n3/10;

}
cout<<"reverse of no.is  "<<rev<<endl;
return 0;
}
```

OUTPUT:

Enter the value of n:

3

sum of first 3 natural no. is:

6

sum of first 3even natural no. is:

12

sum of first 3odd natural no. is:

9

sum of squares of first 3 natural no. is:

14

sum of cubes of first 3 natural no. is:

36

Factorial of 3is:

6

Enter the number:5

the no.of digits is 1

enter the no.54

the number is not prime

enter two no.

34 45

lcm is90

Enter the no.

23

reverse of no.is 32

ASSIGNMENT 7

```
#include <iostream>
#include<math.h>
using namespace std;
// PROGRAM 1
int fib(int n) {
    if (n == 0) {
        return 0;
```

```
    }
    if (n == 1) {
        return 1;
    }
    return fib(n - 1) + fib(n - 2);
}
bool isPrime(int n) {
    if (n <= 1) {
        return false;
    }
    if (n <= 3) {
        return true;
    }
    if (n % 2 == 0 || n % 3 == 0) {
        return false;
    }

    for (int i = 5; i * i <= n; i += 6) {
        if (n % i == 0 || n % (i + 2) == 0) {
            return false;
        }
    }

    return true;
}

int nextPrime(int prime) {
    int next = prime + 1;
    while (true) {
        if (isPrime(next)) {
            return next;
        }
        next++;
    }
}
// Function to check if a number is Armstrong or not
bool isArmstrong(int num) {
    int originalNum = num;
    int n = 0;
    int sum = 0;


    // Calculate the sum of nth powers of digits
    while (originalNum != 0) {
        int digit = originalNum % 10;
```

```
        sum += pow(digit, 3);
        originalNum /= 10;
    }

    // Check if the number is Armstrong
    return (sum == num);
}
int main() {
    int n=0;
    cout << "Enter the value of n: ";
    cin >> n;
    int a = fib(n);
    cout << "The nth term of the Fibonacci sequence is: " << a;
// PROGRAM 2
    cout << "\nThe first " << n << " terms of the Fibonacci sequence are: ";
    for (int i = 0; i < n; i++) {
        cout << fib(i) << " ";
    }
     int p;
    cout << "Enter the number you want to check: ";
    cin >> p;
// PROGRAM 3
    bool isPresent = false;
    for (int i = 0; i < n; i++) {
        if (fib(i) == p) {
            isPresent = true;
            break;
        }
    }

    if (isPresent) {
        cout << p << " is in the Fibonacci sequence." << endl;
    } else {
        cout << p << " is not in the Fibonacci sequence." << endl;
    }
// PROGRAM 4
    int q, r;
    cout << "Enter two numbers: ";
    cin >> q >> r;

    int a1 = q;
    int b1 = r;
    while (b1 != 0) {
        int temp = b1;
        b1 = a1 % b1;
```

```
        a1 = temp;
    }

    cout << "HCF is: " << a1 << endl;

// PROGRAM 5
    if(a1==1){
        cout<<"\n the no are coprime"<<endl;
    }
// PROGRAM 6&7
    int s, e;
    cout << "Enter two numbers: ";
    cin >> s >> e;
    // s=1 &e=100 for prime no. between 1 to 100
    for (int i = s; i <= e; i++) {
        int j;
        for (j = 2; j < i; j++) {
            if (i % j == 0) {
                break;
            }
        }
        if (j == i) {
            cout << i << " is a prime number." << endl;
        }
    }
          int k;
            cin>>k;
     for (int i = 1; i <= 1000; i++) {
        int j;
        for (j = 2; j < i; j++) {
            if (i % j == 0) {
                break;
            }
        }
        if (j == i) {
            int next = nextPrime(k);
            cout << "The next prime number after " << k<< " is: " << next <<
endl;
            break;
        }
    }

  // PROGRAM 9
    int d;
    cout << "Enter a number to check for Armstrong: ";
```

```
    cin >> d;

    int sum1 = 0;
    int originalNumber = d;

    // Calculate the sum of nth powers of digits
    while (originalNumber != 0) {
        int digit = originalNumber % 10;
        sum1 += pow(digit, 3);
        originalNumber /= 10;
    }

    if (sum1 == d) {
        cout << d << " is an Armstrong number." << endl;
    } else {
        cout << d << " is not an Armstrong number." << endl;
    }

    // PROGRAM 10
    cout << "Armstrong numbers under 1000 are:\n";
    for (int i = 0; i < 1000; ++i) {
        if (isArmstrong(i)) {
            cout << i << " ";
        }
    }

    return 0;
}
```

OUTPUT:

Enter the value of n: 4

The nth term of the Fibonacci sequence is: 3

The first 4 terms of the Fibonacci sequence are: 0 1 1 2

Enter the number you want to check: 2

2 is in the Fibonacci sequence.

Enter two numbers: 2 5

HCF is: 1

the no are coprime

Enter two numbers: 2 4

2 is a prime number.

3 is a prime number.

5

The next prime number after 5 is: 7

Enter a number to check for Armstrong: 112

112 is not an Armstrong number.

Armstrong numbers under 1000 are:

0 1 2 3 4 5 6 7 8 9 153 370 371 407

ASSIGNMENT 8

```
#include<iostream>
using namespace std;
void printPattern2(int n) {
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n - i; j++) {
            cout << "  ";
        }
        for (int k = 1; k <= i; k++) {
            cout << "* ";
        }
        cout << endl;
    }
}
void printPattern1(int n) {
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <=  i; j++) {
            cout << "* ";
        }
        for (int k = 1; k <=n-i; k++) {
            cout << "  ";
        }
```

```
        cout << endl;
    }
}
void printPattern3(int n) {
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n - i+1; j++) {
            cout << "* ";
        }
        for (int k = 1; k <= i; k++) {
            cout << "  ";
        }
        cout << endl;
    }
}
void printPattern4(int n) {
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <=  i; j++) {
            cout << "  ";
        }
        for (int k = 1; k <=n-i+1; k++) {
            cout << "* ";
        }
        cout << endl;
    }
}
void printPattern5(int n) {
    for (int i = 1; i <= n; i++) {

        for (int j = 1; j <= n - i; j++) {
            cout << " ";
        }

        for (int k = 1; k <= 2 * i - 1; k++) {
            cout << "*";
        }
        cout << endl;
    }
}

void printPattern6(int n) {
    for (int i = n; i >= 1; i--) {
        // Print spaces
        for (int j = 1; j <= n - i; j++) {
            cout << " ";
        }
```

```
        // Print stars
        for (int k = 1; k <= 2 * i - 1; k++) {
            cout << "*";
        }
        cout << endl;
    }
}

void printPattern7(int n) {
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n - i; j++) {
            cout << "*";
        }
        for (int k = 1; k <= 2 * i - 1; k++) {
            cout << " ";
        }
        for (int l = 1; l <= n - i; l++) {
            cout << "*";
        }
        cout << endl;
    }
}
int main() {
    int rows;
    cout << "Enter the number of rows: ";
    cin >> rows;
 printPattern1(rows);
cout<<endl;
printPattern2(rows);
cout<<endl;
printPattern3(rows);
cout<<endl;
printPattern4(rows);
cout<<endl;
printPattern5(rows);
cout<<endl;
printPattern6(rows);
cout<<endl;
printPattern7(rows);
    return 0;
}
```

OUTPUT:

```
PS C:\Users\ASUS\assignmentclab> cd

Enter the number of rows: 4
*
* *
* * *
* * * *


   *
  * *
 * * *
* * * *


* * * *
 * * *
  * *
   *


 * * * *
  * * *
   * *
    *
```

```
 *
***
*****
*******


*******
*****
***
*


*** ***
**   **
*     *
```

ASSIGNMENT 9

```cpp
#include <iostream>
#include<math.h>
using namespace std;
int main() {
    // PROGRAM 1
    int month;
    cout << "Enter the month: ";
    cin >> month;
    switch (month) {
        case 4:
        case 6:
        case 9:
        case 11:
            cout << "Number of days is 30";
            break;
        case 2:
```

```
            cout << "Number of days is 28";
            break;
        default:
            cout << "Number of days is 31";
    }
 // PROGRAM 2
    int choice;
    do {
        cout << "\n1. Add\n2. Subtract\n3. Multiply\n4. Divide\n5. Exit\n";
        cout << "Enter choice: ";
        cin >> choice;

        switch (choice) {
            case 1:
            case 2:
            case 3:
            case 4:
            {
                int num1, num2;
                cout << "Enter two numbers: ";
                cin >> num1 >> num2;
                cout << "Result: ";
                switch (choice) {
                    case 1: cout << num1 + num2; break;
                    case 2: cout << num1 - num2; break;
                    case 3: cout << num1 * num2; break;
                    case 4: cout << (num1) / num2; break;
                }
                break;
            }
            case 5: cout << " exit!"; break;
            default: cout << "Invalid choice. Try again.";
        }

        cout << "\n";

    } while (choice != 5);
// PROGRAM 3
    int dayNumber;
    cout << "Enter the day number (1-7): ";
    cin >> dayNumber;
    switch (dayNumber) {
        case 1: cout << "Sunday"; break;
        case 2: cout << "Monday"; break;
        case 3: cout << "Tuesday"; break;
```

```
        case 4: cout << "Wednesday"; break;
        case 5: cout << "Thursday"; break;
        case 6: cout << "Friday"; break;
        case 7: cout << "Saturday"; break;
        default: cout << "wrong day number";
    }
// PROGRAM 4
    int a, b, c;
    cout << "Enter three numbers: ";
    cin >> a >> b >> c;
    if (a == b || b == c || c == a) {
        cout << "Isosceles triangle";
    } else if (a * a + b * b == c * c || b * b + c * c == a * a || c * c + a * a
== b * b) {
        cout << "Right-angled triangle";
    } else if (a == b && b == c) {
        cout << "Equilateral triangle";
    } else {
        cout << "Not a special triangle";
    }

// PROGRAM 5
    int var;
    cout << "Enter a number (1-3): ";
    cin >> var;

    switch (var) {
        case 1: cout << "good"; break;
        case 2: cout << "better"; break;
        case 3: cout << "best"; break;
        default: cout << "invalid";
    }

// PROGRAM 6
    int year;
    cout << "Enter a year: ";
    cin >> year;
    switch ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {
        case 1: cout << "Leap year"; break;
        case 0: cout << "Not a leap year"; break;
    }
// PROGRAM 7

// PROGRAM 8
    char choice1;
```

```
    int number;
    cout << "Enter a number: ";
    cin >> number;
    do {
        cout << "\n8. Convert Number\n";
        cout << "a. Positive to Negative\nb. Negative to Positive\nc. Exit\n";
        cout << "Enter choice: ";
        cin >> choice1;
        switch (choice1) {
            case 'a':
                cout << "Converted number: " << -number;
                break;
            case 'b':
                cout << "Converted number: " << abs(number);
                break;
            case 'c':
                cout << "Exiting. Goodbye, MOHD SAQIB!";
                break;
            default:
                cout << "Invalid choice. Try again.";
        }
        cout << "\n";
    } while (choice1 != 'c');
// PROGRAM 9
    char choice2;
    int number1;
    cout << "Enter an even number: ";
    cin >> number1;
    do {
        cout << "\n9. Convert Even Number\n";
        cout << "a. To Upper Nearest Odd\nb. Exit\n";
        cout << "Enter choice: ";
        cin >> choice2;

        switch (choice2) {
            case 'a':
                cout << "Upper Nearest Odd: " << ((number1 % 2 == 0) ? number1 +
1 : number1);
                break;
            case 'b':
                cout << "Exit";
                break;
            default:
                cout << "Invalid choice.";
        }
```

```
        cout << "\n";

    } while (choice2 != 'b');

// PROGRAM 10
    char choice3;
    double a1, b1, c1, discriminant, root1, root2;

    cout << "Enter coefficients (a, b, c) of the quadratic equation: ";
    cin >> a1 >> b1 >> c1;
    do {
        cout << "\n10. Quadratic Equation Roots\n";
        cout << "a. Find Roots\nb. Exit\n";
        cout << "Enter choice: ";
        cin >> choice3;
        switch (choice3) {
            case 'a':
                discriminant = b1 * b1 - 4 * a1 * c1;
                if (discriminant > 0) {
                    root1 = (-b1 + sqrt(discriminant)) / (2 * a1);
                    root2 = (-b1 - sqrt(discriminant)) / (2 * a1);
                    cout << "Roots: " << root1 << " and " << root2;
                } else if (discriminant == 0) {
                    root1 = -b1 / (2 * a1);
                    cout << "Root: " << root1;
                } else {
                    cout << "Complex Roots";
                }
                break;
            case 'b':
                cout << "Exit";
                break;
            default:
                cout << "Invalid choice. Try again.";
        }
        cout << "\n";
    } while (choice3 != 'b');
    return 0;
}
```

OUTPUT:

Enter the month: 2

Number of days is 28

1. Add

2. Subtract

3. Multiply

4. Divide

5. Exit

Enter choice: 1

Enter two numbers: 2 5

Result: 7

1. Add

2. Subtract

3. Multiply

4. Divide

5. Exit

Enter choice: 5

exit!

Enter the day number (1-7): 3

TuesdayEnter three numbers: 3 4 5

Right-angled triangleEnter a number (1-3): 2

betterEnter a year: 2004

Leap yearEnter a number: 23

8. Convert Number

a. Positive to Negative

b. Negative to Positive

c. Exit

Enter choice: B

Invalid choice. Try again.

8. Convert Number

a. Positive to Negative

b. Negative to Positive

c. Exit

Enter choice: a

Converted number: -23

8. Convert Number

a. Positive to Negative

b. Negative to Positive

c. Exit

Enter choice: c

Exiting. Goodbye, MOHD SAQIB!

Enter an even number: 2

9. Convert Even Number

a. To Upper Nearest Odd

b. Exit

Enter choice: b

Exit

Enter coefficients (a, b, c) of the quadratic equation: 2 2 2

10. Quadratic Equation Roots

a. Find Roots

b. Exit

Enter choice: a

Complex Roots

10. Quadratic Equation Roots

a. Find Roots

b. Exit

Enter choice: b

Exit

ASSIGNMENT 10

```
#include<iostream>
#include<math.h>
using namespace std;
// PROGRAM 1
float area(int r){
    float a=3.14*r*r;
    return a;
}
```

```
// PROGRAM 2
float simpleint(float p,float r,int t){
    float si=(p*r*t)/100;
    return si;
}
//PROGRAM 3
bool isevenodd(int num){
    if(num%2==0){
        return true;
    }
    return false;
}
// PROGRAM 4
int firstnat(int n1){
     cout<<"\n first n natural no. is";
    for(int i=1;i<=n1;i++){
        cout<<i<<" ";
    }
}
// PROGRAM 5
int firstnatodd(int n1){
     cout<<"first n odd natural no. is";
    for(int i=1;i<=n1;i++){
        if(i%2!=0){
        cout<<i<<" ";
    }
    }
}
// PROGRAM 6
int factorial(int n2){
    int fact=1;
    for(int i=1;i<=n2;i++){
        fact=fact*i;
    }
    return fact;
}
// PROGRAM 7
int combinations(int N,int R){
    int c= factorial(N)/(factorial(N-R)*factorial(R));
    return c;
}
// PROGRAM 8
int arrangements(int N,int R){
    int p= factorial(N)/factorial(N-R);
    return p;
```

```
}
// PROGRAM 9
bool isdigitpresent(int n3,int dig){
    while(n3!=0){
        int digit=n3%10;
        if(digit==dig){
            return true;
        }
        n3=n3/10;
    }
    return false;
    }
// PROGRAM 10
void printPrimeFactors(int n) {
    while (n % 2 == 0) {
        cout << "2 ";
        n = n / 2;
    }
    for (int i = 3; i <= sqrt(n); i += 2) {
        while (n % i == 0) {
            cout << i << " ";
            n = n / i;
        }
    }
    if (n > 2) {
        cout << n << " ";
    }
    cout << endl;
}

int main(){
    int rad;
    cout<<"Enter the radius of circle:";
    cin>>rad;
   cout<<"area of circle is:"<< area(rad)<<endl;
    int p,r,t;
    cout<<"Enter the principal,rate,time:\n:";
    cin>>p>>r>>t;
   cout<<"simple interest is:"<<simpleint( p, r, t)<<endl;
   cout<<"Enter the number for odd or even\n";
   int n,n1;
   cin>>n;
   cout<<isevenodd(n);
   cout<<"\n enter the no.\n";
   cin>>n1;
```

```
  firstnat(n1);
  firstnatodd(n1);
  int n2,N,R;
  cout<<"\n enter thee number\n";
  cin>>n2;
  cout<<"\nfactorial of "<<n2<<" is "<<factorial(n2)<<endl;
  cout<<"Enter value of n and r for all posssible combinations:";
  cin>>N>>R;
  cout<<"the no of combinations is "<<combinations(N,R)<<endl;
  cout<<"the no of arrangements is "<<arrangements(N,R)<<endl;
  int n3,dig;
  cout<<"enter the number and digit:\n";
  cin>>n3>>dig;
  cout<<isdigitpresent(n3,dig)<<endl;
  int num2;
    cout << "Enter a number: ";
    cin >> num2;
    cout << "Prime factors of " << num2 << " are: ";
    printPrimeFactors(num2);
    return 0;
}
```

OUTPUT:

Enter the radius of circle:4

area of circle is:50.24

Enter the principal,rate,time:

:2 4 5

simple interest is:0.4

Enter the number for odd or even

2

1

enter the no.

3

first n natural no. is1 2 3 first n odd natural no. is1 3

enter thee number

3 4 5

factorial of 3 is 6

Enter value of n and r for all posssible combinations:the no of combinations is 0

the no of arrangements is 24

enter the number and digit:

23 4

0

Enter a number: 5

Prime factors of 5 are: 5

ASSIGNMENT 11

```
#include<iostream>
#include<math.h>
using namespace std;
#include <iostream>
int gcd(int a, int b) {
    if (b == 0) {
        return a;
    }
    return gcd(b, a % b);
}
int lcm(int a, int b) {
    return (a * b) / gcd(a, b);
}
bool isprime(int num) {
    if (num <= 1) {
        return false;
    }
    if (num == 2) {
```

```
        return true;
    }
    if (num % 2 == 0) {
        return false;
    }
    for (int i = 3; i <= sqrt(num); i += 2) {
        if (num % i == 0) {
            return false;
        }
    }
    return true;
}

int nextprime(int num) {
    num++;
    while (!isprime(num)) {
        num++;
    }
    return num;
}
void printPascalsTriangle(int numRows) {
    for (int i = 0; i < numRows; i++) {
        int num = 1;
        for (int j = 0; j < numRows - i; j++) {
            cout << "   ";
        }
        for (int j = 0; j <= i; j++) {
            cout << "      " << num << " ";
            num = num * (i - j) / (j + 1);
        }
        cout << endl;
    }
}
int squarenumber(int p){
    return p*p;
}
int factorial(int r){
    if(r==0||r==1){
        return r;
    }
    return factorial(r-1)*r;
}

int printseries(int m){
     int sum=0;
```

```
    for(int i=1;i<=m;i++){

        sum=sum+(factorial(i)/i);
    }
      return sum;
}




int main() {
    int num1, num2,num,a,b,n1,p;
    cout << "Enter the first number: ";
    cin >> num1;
    cout << "Enter the second number: ";
    cin >> num2;
    int result = lcm(num1, num2);
    int result1 = gcd(num1, num2);
    cout << "LCM of " << num1 << " and " << num2 << " is: " << result <<endl;
    cout << "HCF of " << num1 << " and " << num2 << " is: " << result1 <<endl;
    cout<<"Enter a number:\n";
    cin>>num;
    if(isprime(num)){
        cout<<num<<"is prime number";
    }
    else{
        cout<<num<<"is not prime number";

    }
    cout<<"\nnext prime no. is"<<nextprime(num);
    int n;
    cout << "\nEnter the value of n: ";
    cin >> n;
    int count = 0;
    int num5 = 2;  // Start with the first prime number
    cout << "The first " << n << " prime numbers are:" << endl;
    while (count < n) {
        if (isprime(num5)) {
            cout << num5 << " ";
            count++;
        }
        num5++;
    }
    cout << endl;
    cout << "Enter the values of a and b (a <= b): ";
```

```
    cin >> a >> b;
    cout << "Prime numbers between " << a << " and " << b << " are:" << endl;
    for (int num = a; num <= b; num++) {
        if (isprime(num)) {
            cout << num << " ";
        }
    }

    cout << endl;
    cout << "Enter the value of n1: ";
    cin >> n1;
    if (n1 <= 0) {
        cout << "Please enter a positive integer for n." << endl;
        return 1;
    }
    int first = 0, second = 1;
    cout << "The first " << n1 << " terms of the Fibonacci sequence are:" <<
endl;
    if (n1 >= 1) {
        cout << first << " ";
    }
    if (n1 >= 2) {
        cout << second << " ";
    }

    for (int i = 3; i <= n1; i++) {
        int next = first + second;
        cout << next << " ";
        first = second;
        second = next;
    }
    cout << endl;
    int numRows;
    cout << "Enter the number of rows for Pascal's Triangle: ";
    cin >> numRows;
    cout << "Pascal's Triangle with " << numRows << " rows:" << endl;
    printPascalsTriangle(numRows);
    cout<<"\nEnter a number\n";
    cin>>p;
    cout<<"the square of number is"<<squarenumber(p);
    int m;
    cout<<"no of terms for series 1!/1+2!/2+3!/3+. .. \n";
    cin>>m;
    cout<<"sum of series is"<<printseries(m);
    return 0;
```

OUTPUT:

```
Enter the first number: 2

Enter the second number: 3

LCM of 2 and 3 is: 6

HCF of 2 and 3 is: 1

Enter a number:

2

2is prime number

next prime no. is3

Enter the value of n: 4

The first 4 prime numbers are:

2 3 5 7

Enter the values of a and b (a <= b): 5 10

Prime numbers between 5 and 10 are:

5 7

Enter the value of n1: 2

The first 2 terms of the Fibonacci sequence are:

0 1

Enter the number of rows for Pascal's Triangle: 3

Pascal's Triangle with 3 rows:

          1

        1   1

      1   2   1
```

Enter a number

3

the square of number is9no of terms for series 1!/1+2!/2+3!/3+....

3

sum of series is4

ASSIGNMENT 12

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

// PROGRAM 1
void recfirstnat(int n) {
    if (n > 0) {
        recfirstnat(n - 1);
        cout << n << " ";
    }
}

// PROGRAM 2
void recfirstnatrev(int n) {
    if (n > 0) {
        cout << n << " ";
        recfirstnatrev(n - 1);
    }
}

// PROGRAM 3
void recfirstoddnat(int n) {
    if (n > 0) {
        recfirstoddnat(n - 1);
        if (n % 2 != 0) {
            cout << n << " ";
        }
    }
}

// PROGRAM 4
void recfirstoddnatrev(int n) {
    if (n > 0) {
```

```
        if (n % 2 != 0) {
            cout << n << " ";
        }
        recfirstoddnatrev(n - 1);
    }
}

// PROGRAM 5
void recfirstevennat(int n) {
    if (n > 0) {
        recfirstevennat(n - 1);
        if (n % 2 == 0) {
            cout << n << " ";
        }
    }
}

// PROGRAM 6
void recfirstevennatrev(int n) {
    if (n > 0) {
        if (n % 2 == 0) {
            cout << n << " ";
        }
        recfirstevennatrev(n - 1);
    }
}

// PROGRAM 7
void recfirstsquarenat(int n) {
    if (n > 0) {
        recfirstsquarenat(n - 1);
        cout << n * n << " ";
    }
}

// PROGRAM 8
void recdecimaltobinary(int num) {
    if (num > 0) {
        recdecimaltobinary(num / 2);
        cout << num % 2 << " ";
    }
}

// PROGRAM 9
int recoctaltodecimal(int num) {
```

```
    int decimalNum = 0, i = 0, remainder;
    while (num != 0) {
        remainder = num % 10;
        decimalNum += remainder * pow(8, i);
        ++i;
        num /= 10;
    }
    return decimalNum;
}

// PROGRAM 10
int reversenum(int num) {
    int reversedNum = 0;
    while (num != 0) {
        reversedNum = reversedNum * 10 + num % 10;
        num /= 10;
    }
    return reversedNum;
}

int main(){
    int n;
    cout << "Enter the value of n: ";
    cin >> n;
    recfirstnat(n);
    cout << endl;
    recfirstnatrev(n);
    cout << endl;
    recfirstoddnat(n);
    cout << endl;
    recfirstoddnatrev(n);
    cout << endl;
    recfirstevennat(n);
    cout << endl;
    recfirstevennatrev(n);
    cout << endl;
    recfirstsquarenat(n);
    cout << endl;
    recdecimaltobinary(n);
    cout << endl;
    cout << "Enter an octal number: ";
    int octalNum;
    cin >> octalNum;
    cout << "Decimal representation: " << recoctaltodecimal(octalNum) << endl;
    cout << "Enter a number for Program 10: ";
```

```
    int numToReverse;
    cin >> numToReverse;
    cout << "Reversed number: " << reversenum(numToReverse) << endl;
    return 0;
}
```

OUTPUT:

Enter the value of n: 3

1 2 3

3 2 1

1 3

3 1

2

2

1 4 9

1 1

Enter an octal number: 123

Decimal representation: 83

Enter a number for Program 10: 23

Reversed number: 32

ASSIGNMENT-13

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

//PROGRAM 1
int sumOfFirstN(int n) {
    if (n == 0) {
        return 0;
    } else {
        return n + sumOfFirstN(n - 1);
    }
}

//PROGRAM 2
int sumOfFirstNOdd(int n) {
    if (n == 0) {
        return 0;
    } else {
        return (2 * n - 1) + sumOfFirstNOdd(n - 1);
    }
}

//PROGRAM 3
int sumOfFirstNEven(int n) {
    if (n == 0) {
        return 0;
    } else {
        return 2 * n + sumOfFirstNEven(n - 1);
    }
}

//PROGRAM 4
int sumOfSquares(int n) {
    if (n == 0) {
        return 0;
    } else {
        return n * n + sumOfSquares(n - 1);
    }
}

// //PROGRAM 5
int sumOfDigits(int num) {
    if (num == 0) {
```

```
        return 0;
    } else {
        return num % 10 + sumOfDigits(num / 10);
    }
}

//PROGRAM 6
int factorial(int n) {
    if (n == 0 || n == 1) {
        return 1;
    } else {
        return n * factorial(n - 1);
    }
}

// //PROGRAM 7
int hcf(int a, int b) {
    if (b == 0) {
        return a;
    } else {
        return hcf(b, a % b);
    }
}

//PROGRAM 8
int fibonacci(int n) {
    if (n == 0) {
        return 0;
    } else if (n == 1) {
        return 1;
    } else {
        return fibonacci(n - 1) + fibonacci(n - 2);
    }
}

//PROGRAM 9
int power(int base, int exponent) {
    if (exponent == 0) {
        return 1;
    } else {
        return base * power(base, exponent - 1);
    }
}

//PROGRAM 10
```

```
int countDigits(int num) {
    if (num == 0) {
        return 0;
    } else {
        return 1 + countDigits(num / 10);
    }
}

int main() {
    int n, num, base, exponent;

    cout << "Enter the value of N for sum of first N natural numbers: ";
    cin >> n;
    cout << "Sum of first N natural numbers: " << sumOfFirstN(n) << endl;

    cout << "Enter the value of N for sum of first N odd natural numbers: ";
    cin >> n;
    cout << "Sum of first N odd natural numbers: " << sumOfFirstNOdd(n) << endl;

    cout << "Enter the value of N for sum of first N even natural numbers: ";
    cin >> n;
    cout << "Sum of first N even natural numbers: " << sumOfFirstNEven(n) <<
endl;

    cout << "Enter the value of N for sum of squares of first N natural numbers:
";
    cin >> n;
    cout << "Sum of squares of first N natural numbers: " << sumOfSquares(n) <<
endl;

    cout << "Enter a number to calculate the sum of its digits: ";
    cin >> num;
    cout << "Sum of digits: " << sumOfDigits(num) << endl;

    cout << "Enter a number to calculate its factorial: ";
    cin >> num;
    cout << "Factorial: " << factorial(num) << endl;

    int a, b;
    cout << "Enter two numbers to calculate their HCF: ";
    cin >> a >> b;
    cout << "HCF: " << hcf(a, b) << endl;

    cout << "Enter the value of N for Fibonacci series: ";
    cin >> n;
```

```
    cout << "Fibonacci series up to first N terms: ";
    for (int i = 0; i < n; ++i) {
        cout << fibonacci(i) << " ";
    }
    cout << endl;

    cout << "Enter the base and exponent to calculate power: ";
    cin >> base >> exponent;
    cout << "Power: " << power(base, exponent) << endl;

    cout << "Enter a number to count its digits: ";
    cin >> num;
    cout << "Number of digits: " << countDigits(num) << endl;

    return 0;
}
```

OUTPUT:

Enter the value of N for sum of first N natural numbers: 5

Sum of first N natural numbers: 15

Enter the value of N for sum of first N odd natural numbers: 5

Sum of first N odd natural numbers: 25

Enter the value of N for sum of first N even natural numbers: 5

Sum of first N even natural numbers: 30

Enter the value of N for sum of squares of first N natural numbers: 5

Sum of squares of first N natural numbers: 55

Enter a number to calculate the sum of its digits: 554

Sum of digits: 14

Enter a number to calculate its factorial: 5

Factorial: 120

Enter two numbers to calculate their HCF: 4 5

HCF: 1

Enter the value of N for Fibonacci series: 3

Fibonacci series up to first N terms: 0 1 1

Enter the base and exponent to calculate power: 2 4

Power: 16

Enter a number to count its digits: 234556

Number of digits: 6

ASSIGNMENT-14

```
#include<iostream>
using namespace std;
int main(){
//PROGRAM 1
    int a[10]; int a2=-10000000;
    float sum=0;
    cout<<"enter value of numbers\n";
    for(int i=0;i<10;i++){
        cin>>a[i];
         sum=sum+a[i];
    }
cout<<"the sum of numbers is:"<<sum<<endl;
//PROGRAM 2
cout<<"the average of numbers is:"<<(sum/10)<<endl;
// PROGRAM 3
int a1[10]; int a3=100000000;
int sum1=0; int sum2=0;
// PROGRAM 4&5
 cout<<"enter value of numbers\n";
    for(int i=0;i<10;i++){
        cin>>a1[i];
        if(a1[i]%2==0){
         sum1=sum1+a1[i];
        }
        else{
            sum2=sum2+a1[i];
        }
    }
 // PROGRAM 6&7
```

```
    for(int i=0;i<10;i++){

        if(a1[i]>a2){
            a2=a1[i];
        }
        if(a1[i]<a3){
            a3=a1[i];
        }

    }
    // PROGRAM 8
    int arr[10];
    cout<<"enter value of elements in array:";
     for(int i=0;i<10;i++){
        cin>>arr[i];
    }
    // PROGRAM 9
      for(int i=0;i<10;i++){
        int temp=arr[i];
        int j=i-1;
        while(j>=0&&arr[j]>temp){
                arr[j+1]=arr[j];
                j=j-1;
        }
        arr[j+1]=temp;
        }


    cout<<"Sorted array is:\n";
    for(int i=0;i<10;i++){
        cout<<arr[i]<<" ";
    }
    cout<<"2nd largest& 2nd smallest element is:\n";
    cout<<arr[9]<<" "<<arr[1];


    cout<<"\nthe sum of even numbers is:"<<sum1<<endl;
    cout<<"the sum of odd numbers is:"<<sum2<<endl;
    cout<<"the greatest numbeer is:"<<a2<<endl;
    cout<<"the smallest numbeer is:"<<a3<<endl;
    int n;
    cout<<"Enter value of n:\n";
    cin>>n;
    int arr1[n],arr2[n];
    for(int i=0;i<n;i++){
```

```
        cin>>arr1[i];
    }
    cout<<"array in reverse order is:\n";
    for(int i=n-1;i>=0;i--){
        cout<<arr1[i]<<" ";
    }
     for(int i=0;i<n;i++){
        arr2[i]=arr1[i];
    }
    // PROGRAM 10
    cout<<"copied elements are:\n";
    for(int i=0;i<n;i++){
        cout<<arr2[i]<<" ";
    }

    return 0;
}
```

OUTPUT:

enter value of numbers

1 2 3 4 5 6 7 8 9 10

the sum of numbers is:55

the average of numbers is:5.5

enter value of numbers

1 2 3 4 5 6 7 8 9 10

enter value of elements in array:1 2 3 4 5 6 7 8 9 10

Sorted array is:

1 2 3 4 5 6 7 8 9 10 2nd largest& 2nd smallest element is:

10 2

the sum of even numbers is:30

the sum of odd numbers is:25

the greatest numbeer is:10

the smallest numbeer is:1

Enter value of n:

3

12 23 34

array in reverse order is:

34 23 12 copied elements are:

12 23 34

ASSIGNMENT-15

```
#include<iostream>
using namespace std;
//PROGRAM 1&2
void elementarray(int arr[],int N){
    int a=100000000;
    int b=-1000000000;
    for(int i=0;i<N;i++){
         if(arr[i]>b){
            b=arr[i];
        }
        if(arr[i]<a){
            a=arr[i];
        }
    }
 cout<<"the largest and smallest element are:\n"<<b<<" "<<a;
}
//PROGRAM 3

void sortarray(int arr[],int N){
    for(int i=0;i<N;i++){
        for(int j=0;j<N-i;j++){
            if(arr[j]>arr[j+1]){
                swap(arr[j],arr[j+1]);
            }
        }
    }
    cout<<"\n sorted array is:\n";
    for(int i=0;i<N;i++){
        cout<<arr[i]<<" ";
```

```
    }
}
//PROGRAM 4

void shiftarrbypos(int arr[], int N, int d, int pos) {
    int temp[N];

    for (int i = 0; i < N; i++) {
        if (d == 0) {
            temp[i] = arr[(i + pos) % N];
        }
        if (d == 1) {
            temp[i] = arr[(i + N - pos) % N];
        }
    }

    cout << "\nShifted array is:\n";
    for (int i = 0; i < N; i++) {
        arr[i] = temp[i];
        cout << arr[i] << " ";
    }
    cout<<endl;
}
//PROGRAM 5

int adjduplicate(int arr[],int N){
    for(int i=0;i<N;i++){
        if(arr[i]==arr[i-1]){
            return arr[i];
        }
    }
    return -1;
}
//PROGRAM 6

void reverseprint(int arr2[],int N){
    for(int i=N-1;i>=0;i--){
          cout << arr2[i] << " ";
    }
}
//PROGRAM 7

void noofduplicates(int arr2[], int N) {
    int count = 0;
```

```
    for (int i = 0; i < N - 1; i++) {
        for (int j = i + 1; j < N; j++) {
            if (arr2[i] == arr2[j]) {
                count++;
                break;
            }
        }
    }
    cout << "Number of duplicates: " << count << endl;
}
//PROGRAM 8

bool isUnique(int arr2[], int index, int current) {
    for (int i = 0; i < index; i++) {
        if (arr2[i] == current) {
            return false;
        }
    }
    return true;
}

void printUniqueElements(int arr2[], int N) {
    cout << "Unique elements in the array are: ";
    for (int i = 0; i < N; i++) {
        if (isUnique(arr2, i, arr2[i])) {
            cout << arr2[i] << " ";
        }
    }

    cout << endl;
}
//PROGRAM 9

void mergedarray(int a[],int a2[], int N) {
    int mergedArray[2*N];
    for (int i = 0; i < N; i++) {
        mergedArray[i] = a[i];
    }

    for (int i = 0; i <N; i++) {
        mergedArray[N + i] = a2[i];
    }
    cout << "Merged array is: ";
    for (int i = 0; i <2*N; i++) {
        cout << mergedArray[i] << " ";
```

```
    }

    cout << endl;
}
//PROGRAM 10

void countfreqeachelem(int arr3[], int N) {
    int maxElement = arr3[0];
    for (int i = 1; i < N; i++) {
        if (arr3[i] > maxElement) {
            maxElement = arr3[i];
        }
    }
    const int MAX_SIZE = maxElement + 1;
    int freq[MAX_SIZE] = {0};
    for (int i = 0; i < N; i++) {
        freq[arr3[i]]++;
    }
    for (int i = 0; i < MAX_SIZE; i++) {
        if (freq[i] > 0) {
            cout << "Element " << i << " occurs " << freq[i] << " times." <<
endl;
        }
    }
}
int main(){
    int n,d,pos;
    cout<<"enter the size of array:\n";
    cin>>n;
    int arr[n],arr2[n];
    cout<<"enter array elements:\n";
    for(int i=0;i<n;i++){
        cin>>arr[i];
    }
    elementarray(arr,n);
    sortarray(arr,n);
    cout<<"\n enter 1 for right and 0 for left shifting\n";
    cin>>d;
    cout<<"enter position\t";
    cin>>pos;
    shiftarrbypos(arr,n,d,pos);
    cout<<adjduplicate(arr,n);
    cout<<"\nenter array elements:\n";
    for(int i=0;i<n;i++){
        cin>>arr2[i];
```

```
    }
    cout<<"array in reverse order is:\n";
    reverseprint(arr2,n);
    noofduplicates(arr2,n);
    printUniqueElements(arr2, n);
    int n1,n2;
    cout<<"enter size of array:\n";
    cin>>n1;
    int a[n1],a2[n1];
    cout<<"\n Enter elements of first array:\n";
    for(int i=0;i<n1;i++){
        cin>>a[i];
    }
     cout<<"\n Enter elements of second array:\n";
      for(int i=0;i<n1;i++){
        cin>>a2[i];
    }
    mergedarray(a,a2,n1);
      cout<<"enter size of array:\n";
    cin>>n2;
    int arr3[n2];
    cout<<"\n Enter elements of array:\n";
      for(int i=0;i<n2;i++){
        cin>>arr3[i];
    }
    countfreqeachelem(arr3,n2);
    return 0;
}
```

OUTPUT:

enter the size of array:

5

enter array elements:

12 23 34 45 56

the largest and smallest element are:

56 12

sorted array is:

12 23 34 45 56

enter 1 for right and 0 for lef t shifting

1

enter position 2

Shifted array is:

45 56 12 23 34

-1

enter array elements:

12 22 34 45 56

array in reverse order is:

56 45 34 22 12 Number of duplicates: 0

Unique elements in the array are: 12 22 34 45 56

enter size of array:

3

Enter elements of first array:

12 23 34

Enter elements of second array:

23 344 567

Merged array is: 12 23 34 23 344 567

enter size of array:

4

Enter elements of array:

12 23 34 45

Element 12 occurs 1 times.

Element 23 occurs 1 times.

Element 34 occurs 1 times.

Element 45 occurs 1 times.

ASSIGNMENT-16

```cpp
#include<iostream>
using namespace std;
int main() {
    // PROGRAM 1&2
    int m, n, p;
    cout << "Enter dimensions for matrices (m n p): ";
    cin >> m >> n >> p;
    int mat1[m][n], mat2[n][p], mat3[m][p];
    cout << "Enter values for matrix 1:\n";
    for(int i = 0; i < m; i++) {
        for(int j = 0; j < n; j++) {
            cin >> mat1[i][j];
        }
    }
    cout << "Enter values for matrix 2:\n";
    for(int i = 0; i < n; i++) {
        for(int j = 0; j < p; j++) {
            cin >> mat2[i][j];
        }
    }

    cout << "Addition of matrices is:\n";
    for(int i = 0; i < m; i++) {
        for(int j = 0; j < p; j++) {
            mat3[i][j] = mat1[i][j] + mat2[i][j];
            cout << mat3[i][j] << " ";
        }
        cout << endl;
    }
```

```
    cout << "Multiplication of matrices is:\n";
    for(int i = 0; i < m; i++) {
        for(int j = 0; j < p; j++) {
            mat3[i][j] = 0;
            for(int k = 0; k < n; k++) {
                mat3[i][j] += mat1[i][k] * mat2[k][j];
            }
            cout << mat3[i][j] << " ";
        }
        cout << endl;
    }
    // PROGRAM 3
    int a,b;
    cout<<"enter dimensions of matrix:\n";
    cin>>a>>b;
    int mat4[a][b];
    cout<<"Enter matrix elements:\n";
      for (int i = 0; i < a; i++) {
        for (int j = 0; j < b; j++) {
            cin>>mat4[i][j];
        }
      }
    cout << "Transposed Matrix is:\n";
    for (int i = 0; i < a; i++) {
        for (int j = 0; j < b; j++) {
            cout << mat4[j][i] << " ";
        }
        cout << endl;
    }
    // PROGRAM 4&5
      int c,d;
    cout<<"enter dimensions of matrix:\n";
    cin>>c>>d;
    int mat5[c][d];
    cout<<"enter matrix elements:\n";
      for (int i = 0; i < c; i++) {
        for (int j = 0; j < d; j++) {
            cin>> mat5[i][j];
     }
      }
    cout << "Sum of right &left diagonal are:\n";
    int sum=0; int sum1=0;
    for (int i = 0; i < c; i++) {
        for (int j = 0; j < d; j++) {
            if(i==j){
```

```
                sum=sum+mat5[i][j];
            }
            sum1=sum1+mat5[i][n-1-i];
        }
    }
    cout<<sum<<endl;
    cout<<sum1<<endl;
    // PROGRAM 6
    int rowsum=0;
    int colsum=0;
    for (int i = 0; i < c; i++) {
        for (int j = 0; j < d; j++) {
                rowsum=rowsum+mat5[i][j];
        }
    }
     for (int j = 0; j < d; j++) {
         for (int i = 0; i < c; i++){
            colsum=colsum+mat5[i][j];
        }
    }
    cout<<"sum of rows and columns are:\n"<<rowsum<<" "<<colsum;
    // PROGRAM 7
     cout << "\nUpper Triangular Matrix:\n";
    for (int i = 0; i < c; i++) {
        for (int j = 0; j <d; j++) {
            if (i <= j) {
                cout << mat5[i][j] << " ";
            } else {
                cout << "0 ";
            }
        }
        cout << endl;
    }
// PROGRAM 8
   cout << "lower Triangular Matrix:\n";
    for (int i = 0; i < c; i++) {
        for (int j = 0; j <d; j++) {
            if (i >= j) {
                cout << mat5[i][j] << " ";
            } else {
                cout << "0 ";
            }
        }
        cout << endl;
    }
```

```
    // PROGRAM 9
     int zeroCount = 0;
    int nonZeroCount = 0;
    for (int i = 0; i < c; ++i) {
        for (int j = 0; j <d; ++j) {
            if (mat5[i][j] == 0) {
                zeroCount++;
            } else {
                nonZeroCount++;
            }
        }
    }

     if (zeroCount>nonZeroCount) {
        cout << "The matrix is sparse.\n";
    } else {
        cout << "The matrix is not sparse.\n";
    }
    // PROGRAM 10
    int maxOnesCount = 0;
    int rowIndex = -1;
    for (int i = 0; i <c; ++i) {
        int onesCount = 0;
        for (int j = 0; j <d; ++j) {
            if (mat5[i][j] == 1) {
                onesCount++;
            }
        }

        if (onesCount > maxOnesCount) {
            maxOnesCount = onesCount;
            rowIndex = i;
        }
    }

    if (rowIndex != -1) {
        cout << "Row with the maximum number of 1's is: " << rowIndex + 1 <<
endl;
    } else {
        cout << "No 1's found in the matrix.\n";
    }

    return 0;
}
```

OUTPUT :

Enter dimensions for matrices (m n p): 3 3 3

Enter values for matrix 1:

1 2 3 4 5 6 7 8 9

Enter values for matrix 2:

2 4 6 8 1 2 3 4 5

Addition of matrices is:

3 6 9

12 6 8

10 12 14

Multiplication of matrices is:

27 18 25

66 45 64

105 72 103

enter dimensions of matrix:

3 3

Enter matrix elements:

1 2 3 4 5 6 7 8 9

Transposed Matrix is:

1 4 7

2 5 8

3 6 9

enter dimensions of matrix:

2 2

enter matrix elements:

1 2 3 4

Sum of right &left diagonal are:

5

14

sum of rows and columns are:

10 10

Upper Triangular Matrix:

1 2

0 4

lower Triangular Matrix:

1 0

3 4

The matrix is not sparse.

Row with the maximum number of 1's is: 1

ASSIGNMENT-17

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

// PROGRAM 1
int countVowels(char* str) {
    int count = 0;
    while (*str) {
        char ch = tolower(*str);
        if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u')
            count++;
```

```
        str++;
    }
    return count;
}
void bubbleSort(char arr[][50], int n) {
    for (int i = 0; i < n - 1; ++i) {
        for (int j = 0; j < n - i - 1; ++j) {
            if (strcmp(arr[j], arr[j + 1]) > 0) {
                // Swap the strings
                char temp[50];
                strcpy(temp, arr[j]);
                strcpy(arr[j], arr[j + 1]);
                strcpy(arr[j + 1], temp);
            }
        }
    }
}

bool isPalindrome(const char* str) {
    int length2 = strlen(str);
    for (int i = 0; i < length2 / 2; ++i) {
        if (str[i] != str[length2 - i - 1]) {
            return false;
        }
    }
    return true;
}
bool isValidIPAddress(const char* ipAddress) {
    const char* delimiters = ".";
    char* token = strtok(const_cast<char*>(ipAddress), delimiters);

    int count = 0;
    while (token) {
        int num = atoi(token);
        if (num < 0 || num > 255) {
            return false;
        }

        token = strtok(nullptr, delimiters);
        count++;
    }

    return count == 4; // IP address should have 4 parts
}
```

```
int minDistance(const char* word1, const char* word2, const char* words[], int
wordCount) {
    int minDist = wordCount + 1;
    int pos1 = -1, pos2 = -1;

    for (int i = 0; i < wordCount; ++i) {
        if (strcmp(words[i], word1) == 0) {
            pos1 = i;
        } else if (strcmp(words[i], word2) == 0) {
            pos2 = i;
        }

        if (pos1 != -1 && pos2 != -1) {
            int dist = abs(pos1 - pos2);
            if (dist < minDist) {
                minDist = dist;
            }
        }
    }

    return minDist;
}
bool isUsernameValid(const char* username, const char* userNames[], int
userCount) {
    for (int i = 0; i < userCount; ++i) {
        if (strcmp(username, userNames[i]) == 0) {
            return true;
        }
    }
    return false;
}

int factorial(int n) {
    if (n == 0 || n == 1) {
        return 1;
    }
    return n * factorial(n - 1);
}

bool authenticate(const char* username, const char* password) {
    // Replace this with your authentication logic
    return (strcmp(username, "admin") == 0 && strcmp(password, "admin123") == 0);
}
int main() {
     int stringCount = 5;
```

```
     int maxLength = 50;

    char strings[stringCount][maxLength];
    for (int i = 0; i < stringCount; ++i) {
        cout << "Enter string " << i + 1 << ": ";
        cin.getline(strings[i], maxLength);
    }
    for (int i = 0; i < stringCount; ++i) {
        cout << "String " << i + 1 << " has " << countVowels(strings[i]) << "
vowels.\n";
    }
    // PROGRAM 2
const int cityCount = 10;
const int length = 50;
 char cityNames[cityCount][length];
    for (int i = 0; i < cityCount; ++i) {
        cout << "Enter city name " << i + 1 << ": ";
        cin.getline(cityNames[i], length);
    }

    bubbleSort(cityNames, cityCount);
    cout << "Sorted city names:\n";
    for (int i = 0; i < cityCount; ++i) {
        cout << cityNames[i] << "\n";
    }
    // PROGRAM 3
     int rows,cols;
    cout<<"enter the no. of rows and columns\n";
    cin>>rows>>cols;
    char strings1[rows][cols];
    for (int i = 0; i < rows; ++i) {
        printf("Enter string %d: ", i + 1);
        scanf("%s", strings1[i]);
    }
    for (int i = 0; i < rows; ++i) {
        printf("String %d: %s\n", i + 1, strings1[i]);
    }
    // PROGRAM 4
    const int numStrings = 5;
    const int stringLength = 50;
    char string[numStrings][stringLength];
    char searchStr[stringLength];
    for (int i = 0; i < numStrings; ++i) {
        cout << "Enter string " << i + 1 << ": ";
        cin.getline(string[i], stringLength);
```

```
    }

    // Input string to search
    cout << "Enter a string to search: ";
    cin.getline(searchStr, stringLength);

    // Search for the string
    for (int i = 0; i < numStrings; ++i) {
        if (strcmp(string[i], searchStr) == 0) {
            cout << "String found at position " << i + 1 << "\n";
            break;
        }
    }
    // program 5
     int numEmails = 5;
    int emailLength = 50;
    char emails[numEmails][emailLength];
    for (int i = 0; i < numEmails; ++i) {
        cout << "Enter email address " << i + 1 << ": ";
        cin.getline(emails[i], emailLength);
    }
    for (int i = 0; i < numEmails; ++i) {
        if (strchr(emails[i], '@') == nullptr) {
            cout << "Email without '@': " << emails[i] << "\n";
            break;
        }
    }
// PROGRAM 6
     int rows1 = 5;
    int maxLength1 = 50;
    char strings2[rows1][maxLength1];
    for (int i = 0; i < rows1; ++i) {
        cout << "Enter string " << i + 1 << ": ";
        cin.getline(strings2[i], maxLength1);
    }
    cout << "Palindrome Strings:\n";
    for (int i = 0; i < rows1; ++i) {
        if (isPalindrome(strings2[i])) {
            cout << strings2[i] << "\n";
        }
    }
    // PROGRAM 7
     int  ipCount  = 3;
     int maxLength3 = 20;
    char ipAddresses[ipCount][maxLength3];
```

```
    for (int i = 0; i < ipCount; ++i) {
        cout << "Enter IP address " << i + 1 << ": ";
        cin.getline(ipAddresses[i], maxLength3);
    }
    for (int i = 0; i < ipCount; ++i) {
        cout << "IP address " << i + 1 << " is " <<
(isValidIPAddress(ipAddresses[i]) ? "valid" : "invalid") << "\n";
    }
// PROGRAM 8
     int wordCount = 5;
     int l = 20;
     const char* words[wordCount] = {"the", "quick", "brown", "fox", "quick"};
     const char* word1 = "the";
     const char* word2 = "fox";

    int distance = minDistance(word1, word2, words, wordCount);

    cout << "Minimum distance between '" << word1 << "' and '" << word2 << "': "
<< distance << endl;
// PROGRAM 9
    const int userCount = 3; // Adjust the number of usernames as needed
    const int L = 50;
    const char* userNames[userCount] = {"user1", "user2", "user3"};

    char inputUsername[L];
    cout << "Enter username: ";
    cin.getline(inputUsername, L);

    if (isUsernameValid(inputUsername, userNames, userCount)) {
        int num;
        cout << "Enter a number to calculate its factorial: ";
        cin >> num;
        cout << "Factorial of " << num << " is: " << factorial(num) << "\n";
    } else {
        cout << "Error: Invalid username\n";
    }

// PROGRAM 10

    const int maxL = 50;
    char username[maxL];
    char password[maxL];

    cout << "Enter username: ";
    cin.getline(username, maxL);
```

```
    cout << "Enter password: ";
    cin.getline(password, maxL);

    if (authenticate(username, password)) {
        cout << "Authentication successful. Welcome, " << username << "!\n";

        // Place your menu options here

    } else {
        cout << "Authentication failed. Invalid username or password.\n";
    }
    return 0;
}
```

OUTPUT:

Enter string 1: nayab

Enter string 2: saqib

Enter string 3: qasim

Enter string 4: saad

Enter string 5: saif

String 1 has 2 vowels.

String 2 has 2 vowels.

String 3 has 2 vowels.

String 4 has 2 vowels.

String 5 has 2 vowels.

Enter city name 1: delhi

Enter city name 2: tokyo

Enter city name 3: japan

Enter city name 4: allahabad

Enter city name 5: gujarat

Enter city name 6: washington

Enter city name 7: america

Enter city name 8: dhaka

Enter city name 9: ghaziabaad

Enter city name 10: noida

Sorted city names:

allahabad

america

delhi

dhaka

ghaziabaad

gujarat

japan

noida

tokyo

washington

enter the no. of rows and columns

2

3

Enter string 1: saqib

Enter string 2: saad

String 1: saqsaad

String 2: saad

Enter string 1: Enter string 2: asad

Enter string 3: saqib

Enter string 4: qasim

Enter string 5: nayab

Enter a string to search: saqib

String found at position 3

Enter email address 1: saqib29abubakar@gmail.com

Enter email address 2: nayabkhan64@gmail.com

Enter email address 3: imdad1213@gmail.com

Enter email address 4: mohdsaqib12@gmail.com

Enter email address 5: saqib@123.com

Enter string 1: saqib

Enter string 2: saad

Enter string 3: nayab

Enter string 4: qasim

Enter string 5: rizwan

Palindrome Strings:

Enter IP address 1: 192.134.45.56

Enter IP address 2: 195.56.56.78

Enter IP address 3: 23.345.567.78

IP address 1 is valid

IP address 2 is valid

IP address 3 is invalid

Minimum distance between 'the' and 'fox': 3

Enter username: saqib

Error: Invalid username

Enter username: saqib@123

Enter password: 12345

Authentication failed. Invalid username

ASSIGNMENT-18

```cpp
#include<iostream>
#include<cstring>
#include<limits>
#include<algorithm>
using namespace std;

// PROGRAM 1
void swapValues(int &a, int &b) {
    int temp = a;
    a = b;
    b = temp;
}

// PROGRAM 2
void swapStrings(char str1[], char str2[]) {
    char temp[50];
    strcpy(temp, str1);
    strcpy(str1, str2);
    strcpy(str2, temp);
}
// PROGRAM 3

void sortArray(int ptr[], int size) {
    sort(ptr, ptr + size);
}


int main() {
    int x, y;
    cout << "Enter two numbers to swap:\n";
    cin >> x >> y;
    cin.ignore(numeric_limits<streamsize>::max(), '\n');
```

```
    cout << "Before swapping: x = " << x << ", y = " << y << endl;
    swapValues(x, y);
    cout << "After swapping: x = " << x << ", y = " << y << endl;

    char s1[10];
    char s2[10];

    cout << "Enter the first string: ";
    cin.getline(s1, 10);

    cout << "Enter the second string: ";
    cin.getline(s2, 10);

    cout << "Before swapping: s1 = " << s1 << ", s2 = " << s2 << endl;
    swapStrings(s1, s2);
    cout << "After swapping: s1 = " << s1 << ", s2 = " << s2 << endl;
    int arr[10];
    cout<<"Enter array elements:\n";
    int size = sizeof(arr) / sizeof(arr[0]);
    for(int i=0;i<size;i++){
        cin>>arr[i];
    }
    sortArray(arr, size);
    cout << "Sorted array: ";
    for (int i = 0; i < size; i++) {
        cout << arr[i] << " ";
    }
// PROGRAM 4
int n=64;
int *ptr=&n;
cout<<"value of n is :\n"<<n<<endl;
cout<<"address of n (by ptr)is :\n"<<ptr<<endl;
cout<<"address  of n(by &n) is :\n"<<&n<<endl;
// PROGRAM 5
int n1,n2;
cout<<"enter two numbers:\n";
cin>>n1>>n2;
int *ptr1=&n1;
int *ptr2=&n2;
int c= (*ptr1>*ptr2)?*ptr1:*ptr2;
cout<<"maximum of two numbers is:\n"<<c<<endl;
// PROGRAM 6
cin.ignore(numeric_limits<streamsize>::max(), '\n');
cout << "Enter a string: ";
char str[100];
```

```
cin.getline(str, 100);

    char *ptr3 = str;
    int length = 0;
    int v=0,co=0;
    while (*ptr3 != '\0') {
        length++;
        ptr3++;
    }
    ptr3=str;
    while (*ptr3 != '\0') {
        if (*ptr3 == 'a' || *ptr3 == 'e' || *ptr3 == 'i' || *ptr3 == 'o' || *ptr3
== 'u' ||
            *ptr3 == 'A' || *ptr3 == 'E' || *ptr3 == 'I' || *ptr3 == 'O' || *ptr3
== 'U') {
            v++;
        } else if ((*ptr3 >= 'a' && *ptr3 <= 'z') || (*ptr3 >= 'A' && *ptr3 <=
'Z')) {
            co++;
        }
        ptr3++;
    }

    cout << "Length of the string: " << length << endl;
    cout<<"no of vowels and coinsonants are\n"<<v<<" "<<co;
    // PROGRAM 8
     int size1 = 5;
    int arr1[size1];
    int *ptr4 = arr1;
    int sum = 0;
    cout << "\nEnter " << size1 << " elements of the array:" << endl;
    for (int i = 0; i < size1; i++) {
        cin >> *ptr4;
        sum += *ptr4;
        ptr4++;
    }
    cout << "Sum of array elements: " << sum << endl;
// PROGRAM 9
    int *ptr5 = arr1 + size1 - 1;
    cout << "Enter " << size1 << " elements of the array:" << endl;
    for (int i = 0; i < size1; i++) {
        cin >> *ptr5;
        ptr5--;
    }

    cout << "Array elements in reverse order: ";
```

```
    ptr5 = arr1;
    for (int i = 0; i < size1; i++) {
        cout << *ptr5 << " ";
        ptr5++;
    }

//PROGRAM  10.
    char str5[100];
    cin.ignore(numeric_limits<streamsize>::max(), '\n');
    cout << "\nEnter a string: ";
    cin.getline(str5, 100);

    char *ptr6 = str5 + strlen(str5) - 1;

    cout << "String in reverse order: ";
    while (ptr6 >= str5) {
        cout << *ptr6;
        ptr6--;
    }
cout<<endl;
    return 0;
}
```

OUTPUT:

Enter two numbers to swap:

23 34

Before swapping: x = 23, y = 34

After swapping: x = 34, y = 23

Enter the first string: saad

Enter the second string: nayab

Before swapping: s1 = saad, s2 = nayab

After swapping: s1 = nayab, s2 = saad

Enter array elements:

1 2 3 4 5

6 7 8 9 10

Sorted array: 1 2 3 4 5 6 7 8 9 10 value of n is :

64

address of n (by ptr)is :

0x61fe64

address of n(by &n) is :

0x61fe64

enter two numbers:

64 32

maximum of two numbers is:

64

Enter a string: saqib

Length of the string: 5

no of vowels and coinsonants are

2 3

Enter 5 elements of the array:

12 23 34 45 56

Sum of array elements: 170

Enter 5 elements of the array:

23 34 45 56 67

Array elements in reverse order: 67 56 45 34 23

Enter a string: saqibatlaptop

String in reverse order: potpaltabiqas

ASSIGNMENT-19

```cpp
#include <iostream>
#include <algorithm>
using namespace std;
struct Employee {
    int id;
    string name;
    double salary;
};
void inputEmployeeData(Employee& emp) {
    cout << "Enter Employee ID: ";
    cin >> emp.id;
    cout << "Enter Employee Name: ";
    cin.ignore();
    getline(cin, emp.name);
    cout << "Enter Employee Salary: ";
    cin >> emp.salary;
}
void displayEmployeeData(const Employee& emp) {
    cout << "Employee ID: " << emp.id << endl;
    cout << "Employee Name: " << emp.name << endl;
    cout << "Employee Salary: " << emp.salary << endl;
}
Employee findHighestSalaryEmployee(Employee arr[], int size) {
    return *max_element(arr, arr + size, [](const Employee& a, const Employee& b)
{
        return a.salary < b.salary;
    });
}

void sortEmployeesBySalary(Employee arr[], int size) {
    sort(arr, arr + size, [](const Employee& a, const Employee& b) {
        return a.salary < b.salary;
    });
}
void sortEmployeesByName(Employee arr[], int size) {
    sort(arr, arr + size, [](const Employee& a, const Employee& b) {
        return a.name < b.name;
    });
}
struct Time {
    int hours;
    int minutes;
    int seconds;
```

```
};

Time calculateTimeDifference(const Time& startTime, const Time& endTime) {
    Time difference;

    int startSeconds = startTime.hours * 3600 + startTime.minutes * 60 +
startTime.seconds;
    int endSeconds = endTime.hours * 3600 + endTime.minutes * 60 +
endTime.seconds;

    int timeDifference = endSeconds - startSeconds;

    difference.hours = timeDifference / 3600;
    difference.minutes = (timeDifference % 3600) / 60;
    difference.seconds = (timeDifference % 3600) % 60;

    return difference;
}

struct Student {
    int rollNo;
    string name;
    float marks;
};

void inputStudentData(Student& student) {
    cout << "Enter Roll Number: ";
    cin >> student.rollNo;

    cout << "Enter Name: ";
    cin.ignore();
    getline(cin, student.name);

    cout << "Enter Marks: ";
    cin >> student.marks;
}

void displayStudentData(const Student& student) {
    cout << "Roll Number: " << student.rollNo << endl;
    cout << "Name: " << student.name << endl;
    cout << "Marks: " << student.marks << endl;
}

int main() {
    const int numEmployees = 10;
```

```
    Employee employees[numEmployees];

    for (int i = 0; i < numEmployees; ++i) {
        cout << "\nEnter details for Employee " << i + 1 << ":\n";
        inputEmployeeData(employees[i]);
    }

    cout << "\nDisplaying Employee Data:\n";
    for (int i = 0; i < numEmployees; ++i) {
        cout << "\nDetails for Employee " << i + 1 << ":\n";
        displayEmployeeData(employees[i]);
    }

    Employee highestSalaryEmployee = findHighestSalaryEmployee(employees,
numEmployees);
    cout << "\nEmployee with the highest salary:\n";
    displayEmployeeData(highestSalaryEmployee);

    sortEmployeesBySalary(employees, numEmployees);
    cout << "\nEmployees sorted by salary:\n";
    for (int i = 0; i < numEmployees; ++i) {
        displayEmployeeData(employees[i]);
    }

    sortEmployeesByName(employees, numEmployees);
    cout << "\nEmployees sorted by name:\n";
    for (int i = 0; i < numEmployees; ++i) {
        displayEmployeeData(employees[i]);
    }

    Time startTime, endTime, timeDifference;
    cout << "Enter start time (hh mm ss): ";
    cin >> startTime.hours >> startTime.minutes >> startTime.seconds;

    cout << "Enter end time (hh mm ss): ";
    cin >> endTime.hours >> endTime.minutes >> endTime.seconds;

    timeDifference = calculateTimeDifference(startTime, endTime);
    cout << "Time Difference: " << timeDifference.hours << " hours, "
         << timeDifference.minutes << " minutes, " << timeDifference.seconds << "
seconds" << endl;

    const int numStudents = 10;
    Student students[numStudents];
```

```
    for (int i = 0; i < numStudents; ++i) {
        cout << "Enter details for Student " << i + 1 << ":\n";
        inputStudentData(students[i]);
    }

    cout << "\nDetails of Students:\n";
    for (int i = 0; i < numStudents; ++i) {
        cout << "\nDetails for Student " << i + 1 << ":\n";
        displayStudentData(students[i]);
    }

    int numAdditionalStudents;
    cout << "Enter the number of additional students: ";
    cin >> numAdditionalStudents;

    Student* additionalStudents = new Student[numAdditionalStudents];

    for (int i = 0; i < numAdditionalStudents; ++i) {
        cout << "Enter details for additional Student " << i + 1 << ":\n";
        inputStudentData(additionalStudents[i]);
    }

    cout << "\nDetails of Additional Students:\n";
    for (int i = 0; i < numAdditionalStudents; ++i) {
        cout << "\nDetails for Additional Student " << i + 1 << ":\n";
        displayStudentData(additionalStudents[i]);
    }

    delete[] additionalStudents;

    const int numMarksStudents = 5;
    struct Marks {
        int rollNo;
        string name;
        int chemMarks;
        int mathsMarks;
        int phyMarks;
    };
    Marks marksStudents[numMarksStudents];

    // Input marks for 5 students
    for (int i = 0; i < numMarksStudents; ++i) {
        cout << "Enter details for Student " << i + 1 << ":\n";
        cout << "Roll Number: ";
        cin >> marksStudents[i].rollNo;
```

```
        cout << "Name: ";
        cin.ignore();
        getline(cin, marksStudents[i].name);
        cout << "Chemistry Marks: ";
        cin >> marksStudents[i].chemMarks;
        cout << "Mathematics Marks: ";
        cin >> marksStudents[i].mathsMarks;
        cout << "Physics Marks: ";
        cin >> marksStudents[i].phyMarks;
    }

    cout << "\nPercentage of Students:\n";
    for (int i = 0; i < numMarksStudents; ++i) {
        float totalMarks = marksStudents[i].chemMarks +
marksStudents[i].mathsMarks + marksStudents[i].phyMarks;
        float percentage = (totalMarks / 300) * 100;

        cout << "\nDetails for Student " << i + 1 << ":\n";
        cout << "Roll Number: " << marksStudents[i].rollNo << endl;
        cout << "Name: " << marksStudents[i].name << endl;
        cout << "Percentage: " << percentage << "%" << endl;
    }

    return 0;
}
```

OUTPUT:

Enter details for Employee 1:

Enter Employee ID: 1001

Enter Employee Name: saqib

Enter Employee Salary: 10000

Enter details for Employee 2:

Enter Employee ID: 1002

Enter Employee Name: nayab

Enter Employee Salary: 100000

Enter details for Employee 3:

Enter Employee ID: 1003

Enter Employee Name: qasim

Enter Employee Salary: 35000

Enter details for Employee 4:

Enter Employee ID: 1004

Enter Employee Name: saif

Enter Employee Salary: 45000

Enter details for Employee 5:

Enter Employee ID: 1005

Enter Employee Name: tauheed

Enter Employee Salary: 70000

Enter details for Employee 6:

Enter Employee ID: 1006

Enter Employee Name: rizwan

Enter Employee Salary: 5000

Enter details for Employee 7:

Enter Employee ID: 1007

Enter Employee Name: tejas

Enter Employee Salary: 80000

Enter details for Employee 8:

Enter Employee ID: 1008

Enter Employee Name: asad

Enter Employee Salary: 25000

Enter details for Employee 9:

Enter Employee ID: 1009

Enter Employee Name: imdad

Enter Employee Salary: 50990

Enter details for Employee 10:

Enter Employee ID: 1010

Enter Employee Name: abubakar

Enter Employee Salary: 230000

Displaying Employee Data:

Details for Employee 1:

Employee ID: 1001

Employee Name: saqib

Employee Salary: 10000

Details for Employee 2:

Employee ID: 1002

Employee Name: nayab

Employee Salary: 100000

Details for Employee 3:

Employee ID: 1003

Employee Name: qasim

Employee Salary: 35000

Details for Employee 4:

Employee ID: 1004

Employee Name: saif

Employee Salary: 45000

Details for Employee 5:

Employee ID: 1005

Employee Name: tauheed

Employee Salary: 70000

Details for Employee 6:

Employee ID: 1006

Employee Name: rizwan

Employee Salary: 50000

Details for Employee 7:

Employee ID: 1007

Employee Name: tejas

Employee Salary: 80000

Details for Employee 8:

Employee ID: 1008

Employee Name: asad

Employee Salary: 25000

Details for Employee 9:

Employee ID: 1009

Employee Name: imdad

Employee Salary: 50990

Details for Employee 10:

Employee ID: 1010

Employee Name: abubakar

Employee Salary: 230000

Employee with the highest salary:

Employee ID: 1010

Employee Name: abubakar

Employee Salary: 230000

Employees sorted by salary:

Employee ID: 1001

Employee Name: saqib

Employee Salary: 10000

Employee ID: 1008

Employee Name: asad

Employee Salary: 25000

Employee ID: 1003

Employee Name: qasim

Employee Salary: 35000

Employee ID: 1004

Employee Name: saif

Employee Salary: 45000

Employee ID: 1006

Employee Name: rizwan

Employee Salary: 50000

Employee ID: 1009

Employee Name: imdad

Employee Salary: 50990

Employee ID: 1005

Employee Name: tauheed

Employee Salary: 70000

Employee ID: 1007

Employee Name: tejas

Employee Salary: 80000

Employee ID: 1002

Employee Name: nayab

Employee Salary: 100000

Employee ID: 1010

Employee Name: abubakar

Employee Salary: 230000

Employees sorted by name:

Employee ID: 1010

Employee Name: abubakar

Employee Salary: 230000

Employee ID: 1008

Employee Name: asad

Employee Salary: 25000

Employee ID: 1009

Employee Name: imdad

Employee Salary: 50990

Employee ID: 1002

Employee Name: nayab

Employee Salary: 100000

Employee ID: 1003

Employee Name: qasim

Employee Salary: 35000

Employee ID: 1006

Employee Name: rizwan

Employee Salary: 50000

Employee ID: 1004

Employee Name: saif

Employee Salary: 45000

Employee ID: 1001

Employee Name: saqib

Employee Salary: 10000

Employee ID: 1005

Employee Name: tauheed

Employee Salary: 70000

Employee ID: 1007

Employee Name: tejas

Employee Salary: 80000

Enter start time (hh mm ss): 12 34 56

Enter end time (hh mm ss): 12 34 58

Time Difference: 0 hours, 0 minutes, 2 seconds

Enter details for Student 1:

Enter Roll Number: 12

Enter Name: wahid

Enter Marks: 45

Enter details for Student 2:

Enter Roll Number: 64

Enter Name: nayb

Enter Marks: 99

Enter details for Student 3:

Enter Roll Number: 59

Enter Name: saqib

Enter Marks: 0

Enter details for Student 4:

Enter Roll Number: 57

Enter Name: qasim

Enter Marks: 34

Enter details for Student 5:

Enter Roll Number: 23

Enter Name: saad

Enter Marks: 34

Enter details for Student 6:

Enter Roll Number: 45

Enter Name: bahauddin

Enter Marks: 56

Enter details for Student 7:

Enter Roll Number: 39

Enter Name: tejas

Enter Marks: 45

Enter details for Student 8:

Enter Roll Number: 49

Enter Name: danish

Enter Marks: 34

Enter details for Student 9:

Enter Roll Number: 1

Enter Name: saifullah

Enter Marks: 34

Enter details for Student 10:

Enter Roll Number: 2

Enter Name: imdad

Enter Marks: 98

Details of Students:

Details for Student 1:

Roll Number: 12

Name: wahid

Marks: 45

Details for Student 2:

Roll Number: 64

Name: nayb

Marks: 99

Details for Student 3:

Roll Number: 59

Name: saqib

Marks: 0

Details for Student 4:

Roll Number: 57

Name: qasim

Marks: 34

Details for Student 5:

Roll Number: 23

Name: saad

Marks: 34

Details for Student 6:

Roll Number: 45

Name: bahauddin

Marks: 56

Details for Student 7:

Roll Number: 39

Name: tejas

Marks: 45

Details for Student 8:

Roll Number: 49

Name: danish

Marks: 34

Details for Student 9:

Roll Number: 1

Name: saifullah

Marks: 34

Details for Student 10:

Roll Number: 2

Name: imdad

Marks: 98

Enter the number of additional students: 2

Enter details for additional Student 1:

Enter Roll Number: 3

Enter Name: turab

Enter Marks: 34

Enter details for additional Student 2:

Enter Roll Number: 4

Enter Name: asadullah

Enter Marks: 45

Details of Additional Students:

Details for Additional Student 1:

Roll Number: 3

Name: turab

Marks: 34

Details for Additional Student 2:

Roll Number: 4

Name: asadullah

Marks: 45

Enter details for Student 1:

Roll Number: 2

Name: ert

Chemistry Marks: 5

Mathematics Marks: 5

Physics Marks: 5

Enter details for Student 2:

Roll Number: 2

Name: saqib

Chemistry Marks: 23

Mathematics Marks: 34

Physics Marks: 45

Enter details for Student 3:

Roll Number: 56

Name: munna

Chemistry Marks: 34

Mathematics Marks: 45

Physics Marks: 67

Enter details for Student 4:

Roll Number: 56

Name: munna2

Chemistry Marks: 56

Mathematics Marks: 67

Physics Marks: 78

Enter details for Student 5:

Roll Number: 56

Name: rishi

Chemistry Marks: 25

Mathematics Marks: 34

Physics Marks: 98

Percentage of Students:

Details for Student 1:

Roll Number: 2

Name: ert

Percentage: 5%

Details for Student 2:

Roll Number: 2

Name: saqib

Percentage: 34%

Details for Student 3:

Roll Number: 56

Name: munna

Percentage: 48.6667%

Details for Student 4:

Roll Number: 56

Name: munna2

Percentage: 67%

Details for Student 5:

Roll Number: 56

Name: rishi

Percentage: 52.3333%

ASSIGNMENT-20

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

// PROGRAM 1
void inputString(string& str) {
    cout << "Enter a string: ";
    getline(cin, str);
}
int main() {
    string Str;
    inputString(Str);
// Program 2
    int n;
    cout << "Enter the number of data values: ";
    cin >> n;
    int* dataArray = new int[n];
    cout << "Enter " << n << " data values:\n";
    double sum = 0;
    for (int i = 0; i < n; ++i) {
        cin >> dataArray[i];
        sum += dataArray[i];
    }
    double average = sum / n;
    cout << "Average of data values: " << average << endl;
    delete[] dataArray;
// Program 3
    int n1;
    cout << "Enter the number of values: ";
    cin >> n1;
    int* dataArray1 = (int*)malloc(n1 * sizeof(int));
    cout << "Enter " << n1 << " numbers:\n";
    int sum1 = 0;
    for (int i = 0; i < n1; ++i) {
        cin >> dataArray1[i];
        sum1 += dataArray1[i];
    }
    cout << "Sum of numbers: " << sum1 << endl;
    free(dataArray1);
    // PROGRAM 4
    cout << "Enter text: ";
    cin.ignore();
```

```
    char* text = new char[100];
    cin.getline(text, 100);
    cout << "Entered text: " << text << endl;
    delete[] text;
// PROGRAM 5
    int n2;
    cout << "Enter the size of the array: ";
    cin >> n2;
    int* arr2 = new int[n2];
    cout << "Enter " << n2 << " elements:\n";
    int sum2 = 0;
    for (int i = 0; i < n2; ++i) {
        cin >> arr2[i];
        sum2 += arr2[i];
    }
    cout << "Array elements: ";
    for (int i = 0; i < n2; ++i) {
        cout << arr2[i] << " ";
    }
    cout << "\nSum of array elements: " << sum2 << endl;
    delete[] arr2;
// program 6
    int n5;
    cout << "Enter the size of the array: ";
    cin >> n5;
    int* arr = new int[n5];
    cout << "Enter " << n5 << " elements:\n";
    for (int i = 0; i < n5; ++i) {
        cin >> arr[i];
    }
    int maxElement = arr[0];
    for (int i = 1; i < n5; ++i) {
        if (arr[i] > maxElement) {
            maxElement = arr[i];
        }
    }
    cout << "Largest element: " << maxElement << endl;
    delete[] arr;

//   PROGRAM 7
    int* data = new int;
    *data = 42;
    // No 'delete' statement

// PROGRAM 8
```

```
    int* ptr = new int;
    *ptr = 10;
    delete ptr;
    cout << "Value at dangling pointer: " << *ptr << endl;
// #PROGRAM 9
    int size;
    cout << "Enter the size in bytes: ";
    cin >> size;
    int* data1 = new (nothrow) int[size];
    if (data1 == nullptr) {
        cout << "Memory allocation failed.\n";
    } else {
        cout << "Memory allocated successfully.\n";
        delete[] data1;
    }
    // PROGRAM 10
    int n10;
    cout << "Enter the size of the array: ";
    cin >> n10;
    int* arr5 = new int[n10];
    cout << "Enter " << n10 << " elements:\n";
    for (int i = 0; i < n10; ++i) {
        cin >> arr5[i];
    }
    int maxElem = arr5[0];
    int minElem = arr5[0];

    for (int i = 1; i < n10; ++i) {
       if (arr5[i] > maxElem) {
           maxElem= arr[i];
        }
        if (arr5[i] < minElem) {
            minElem= arr[i];
        }
    }
    cout << "Maximum element: " << maxElem << endl;
    cout << "Minimum element: " << minElem << endl;
    delete[] arr5;
    return 0;
}
```

OUTPUT:

Enter a string: saqib

Enter the number of data values: 2

Enter 2 data values:

12 23

Average of data values: 17.5

Enter the number of values: 2

Enter 2 numbers:

23 56

Sum of numbers: 79

Enter text: nayab

Entered text: nayab

Enter the size of the array: 2

Enter 2 elements:

23 47

Array elements: 23 47

Sum of array elements: 70

Enter the size of the array: 3

Enter 3 elements:

1 2 3

Largest element: 3

Value at dangling pointer: 9269376

Enter the size in bytes: 4

Memory allocated successfully.

Enter the size of the array: 4

Enter 4 elements:

12 23 34 45

Maximum element: 45

Minimum element: 12

ASSIGNMENT 21

(C LAB PROGRAMS)

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

// program 2: to print pyramid and diamond pattern of stars
void printpyramid(int n) {
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= i; j++) {
            printf("* ");
        }
        printf("\n");
    }
cout<<endl;
}

void printDiamond(int n) {
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n - i; j++)
            cout << "  ";

        for (int k = 1; k <= 2 * i - 1; k++)
            cout << "* ";

        cout << endl;
    }

    for (int i = n - 1; i >= 1; i--) {
        for (int j = 1; j <= n - i; j++)
            cout << "  ";

        for (int k = 1; k <= 2 * i - 1; k++)
            cout << "* ";

        cout << endl;
    }
}
// program 3: to traverse element of an array:
//  a) Find minimum.
// b) Find Maximum.
```

```
// c) Sum of elements of the array.
// d) Insert element to array.
void traverseArray(int arr[], int size) {
    int sum = 0, minVal = INT_MAX, maxVal = INT_MIN;
    cout << "Array Elements: ";
    for (int i = 0; i < size; ++i) {
        cout << arr[i] << " ";
        sum += arr[i];
        if (arr[i] < minVal) {
            minVal = arr[i];
        }
        if (arr[i] > maxVal) {
            maxVal = arr[i];
        }
    }
    cout << "\nMinimum Element: " << minVal << "\nMaximum Element: " << maxVal <<
"\nSum of Elements: " << sum << endl;
}
void insertElement(int arr[], int& size, int element, int position) {
    if (position < 0 || position > size) {
        cout << "Invalid position for insertion." << endl;
        return;
    }
    for (int i = size; i > position; --i) {
        arr[i] = arr[i - 1];
    }
    arr[position] = element;
    ++size;
    cout << "Element inserted successfully." << endl;
       for (int i = 0; i < size; ++i) {
        cout << arr[i] << " ";
       }
}

int main(){
    int p;
    printf("\nEnter the number of rows: ");
    scanf("%d", &p);
    printpyramid(p);
   printDiamond(p);
   int maxSize = 100;
    int arr[maxSize], size, element, position;
    cout << "Enter the size of the array: ";
    cin >> size;
    cout << "Enter the array elements: ";
```

```
    for (int i = 0; i < size; ++i) {
        cin >> arr[i];
    }
    traverseArray(arr, size);
    cout << "Enter the element to insert: ";
    cin >> element;
    cout << "Enter the position to insert: ";
    cin >> position;
    insertElement(arr, size, element, position);

    return 0;
}

#include<iostream>
using namespace std;
int main() {

    int m, n, p;
    cout << "Enter dimensions for matrices (m n p): ";
    cin >> m >> n >> p;
    int mat1[m][n], mat2[n][p], mat3[m][p];
    cout << "Enter values for matrix 1:\n";
    for(int i = 0; i < m; i++) {
        for(int j = 0; j < n; j++) {
            cin >> mat1[i][j];
        }
    }
    cout << "Enter values for matrix 2:\n";
    for(int i = 0; i < n; i++) {
        for(int j = 0; j < p; j++) {
            cin >> mat2[i][j];
        }
    }




    while(true){
        cout << "a) Addition of matrices.\n";
        cout << "b) Subtraction of matrices.\n";
        cout << "c) Multiplication of matrices.\n";
        cout << "d) Display even position elements of the first matrix.\n";
        cout << "e) Exit.\n";
        cout << "Enter your choice: ";
        int choice;
        cin >> choice;
```

```
    switch(choice){
        case 1:
             cout << "Addition of matrices is:\n";
for(int i = 0; i < m; i++) {
    for(int j = 0; j < p; j++) {
        mat3[i][j] = mat1[i][j] + mat2[i][j];
        cout << mat3[i][j] << " ";
    }
    cout << endl;
}
        break;
                case 2:
             cout << "Subtraction  of matrices is:\n";
for(int i = 0; i < m; i++) {
    for(int j = 0; j < p; j++) {
        mat3[i][j] = mat1[i][j] -mat2[i][j];
        cout << mat3[i][j] << " ";
    }
    cout << endl;
}
        break;
     case 3:
            cout << "Multiplication of matrices is:\n";
for(int i = 0; i < m; i++) {
    for(int j = 0; j < p; j++) {
        mat3[i][j] = 0;
        for(int k = 0; k < n; k++) {
            mat3[i][j] += mat1[i][k] * mat2[k][j];
        }
        cout << mat3[i][j] << " ";
    }
    cout << endl;
}
break;
case 4:

             cout << "Displaying even position elements of matrix 1 :\n";
for(int i = 0; i < m; i++) {
    for(int j = 0; j < p; j++) {
      if((i+j)%2==0){
        cout << mat1[i][j] << " ";
    }
    }
    cout << endl;
}
```

```
            break;
            default:
            cout<<"invalid choice! \n";
            exit(0);
        }
    }

    }
```

```
#include <stdio.h>
void findSaddlePoint(int matrix[100][100], int n);
int main() {
    int n;
    printf("Enter the order of the matrix (n x n): ");
    scanf("%d", &n);
    int matrix[100][100];
    printf("Enter the elements of the matrix:\n");
    for (int i = 0; i < n; ++i) {
        for (int j = 0; j < n; ++j) {
            scanf("%d", &matrix[i][j]);
        }
    }
    findSaddlePoint(matrix, n);
    return 0;
}

void findSaddlePoint(int matrix[100][100], int n) {
    for (int i = 0; i < n; ++i) {
        // Find the minimum element in the current row
        int minElement = matrix[i][0];
        int minIndex = 0;
        for (int j = 1; j < n; ++j) {
            if (matrix[i][j] < minElement) {
                minElement = matrix[i][j];
                minIndex = j;
            }
        }

        // Check if the minimum element is also the maximum element in its column
        int isSaddlePoint = 1;
        for (int k = 0; k < n; ++k) {
            if (matrix[k][minIndex] > minElement) {
                isSaddlePoint = 0;
                break;
            }
```

```
        }

        // If the minimum element is also the maximum element in its column, it
is a saddle point
        if (isSaddlePoint) {
            printf("Saddle Point Found: %d at position (%d, %d)\n", minElement, i
+ 1, minIndex + 1);
            return;
        }
    }

    // If no saddle point is found
    printf("No Saddle Point Found.\n");
}

#include <iostream>
using namespace std;

const int MAX_EMPLOYEES = 100;

struct Employee {
    int empID;
    int age;
    int experience;
    float basicSalary;
    float bonus;
    float grossSalary;
};

void displayGrossSalary(Employee employees[], int numEmployees);
int findMostExperiencedEmployee(Employee employees[], int numEmployees);
int findHighestPaidEmployee(Employee employees[], int numEmployees);

int main() {
    int numEmployees;

    cout << "Enter the number of employees: ";
    cin >> numEmployees;

    if (numEmployees > MAX_EMPLOYEES) {
        cout << "Number of employees exceeds the maximum limit.\n";
        return 1; // Exit with an error code
    }

    Employee employees[MAX_EMPLOYEES];
```

```
    // Input employee details
    cout << "Enter employee details (EMP ID, AGE, EXPERIENCE, BASIC SALARY,
BONUS):\n";
    for (int i = 0; i < numEmployees; ++i) {
        cin >> employees[i].empID >> employees[i].age >> employees[i].experience
>> employees[i].basicSalary >> employees[i].bonus;
        employees[i].grossSalary = employees[i].basicSalary + employees[i].bonus;
    }

    // Display Gross Salary of each employee
    displayGrossSalary(employees, numEmployees);

    // Display most experienced employee
    int mostExperiencedEmpIndex = findMostExperiencedEmployee(employees,
numEmployees);
    cout << "Most experienced employee: EMP ID " <<
employees[mostExperiencedEmpIndex].empID << endl;

    // Display Highest Paid employee
    int highestPaidEmpIndex = findHighestPaidEmployee(employees, numEmployees);
    cout << "Highest Paid employee: EMP ID " <<
employees[highestPaidEmpIndex].empID << endl;

    return 0;
}

void displayGrossSalary(Employee employees[], int numEmployees) {
    cout << "Gross Salary of each employee:\n";
    for (int i = 0; i < numEmployees; ++i) {
        cout << "EMP ID " << employees[i].empID << ": " <<
employees[i].grossSalary << endl;
    }
}

int findMostExperiencedEmployee(Employee employees[], int numEmployees) {
    int mostExpIndex = 0;

    for (int i = 1; i < numEmployees; ++i) {
        if (employees[i].experience > employees[mostExpIndex].experience ||
            (employees[i].experience == employees[mostExpIndex].experience &&
employees[i].age < employees[mostExpIndex].age)) {
            mostExpIndex = i;
        }
    }
```

```
    return mostExpIndex;
}

int findHighestPaidEmployee(Employee employees[], int numEmployees) {
    int highestPaidIndex = 0;

    for (int i = 1; i < numEmployees; ++i) {
        if (employees[i].grossSalary > employees[highestPaidIndex].grossSalary ||
            (employees[i].grossSalary == employees[highestPaidIndex].grossSalary
&&
             (employees[i].experience > employees[highestPaidIndex].experience ||
              (employees[i].experience == employees[highestPaidIndex].experience
&& employees[i].age < employees[highestPaidIndex].age)))) {
            highestPaidIndex = i;
        }
    }

    return highestPaidIndex;
}
```

OUTPUT:

Enter the number of rows: 4

```
*

* *

* * *

* * * *
```

```
    *
   * * *
  * * * * *
* * * * * * *
  * * * * *
   * * *
    *
```

Enter the size of the array: 5

Enter the array elements: 12 23 34 45 56

Array Elements: 12 23 34 45 56

Minimum Element: 12

Maximum Element: 56

Sum of Elements: 170

Enter the element to insert: 21

Enter the position to insert: 2

Element inserted successfully.

12 23 21 34 45 56

Enter dimensions for matrices (m n p): 3 3 3

Enter values for matrix 1:

1 2 3 4 5 6 7 8 9

Enter values for matrix 2:

1 2 3 4 5 6 7 8 9

a) Addition of matrices.

b) Subtraction of matrices.

c) Multiplication of matrices.

d) Display even position elements of the first matrix.

e) Exit.

Enter your choice: 1

Addition of matrices is:

2 4 6

8 10 12

14 16 18

a) Addition of matrices.

b) Subtraction of matrices.

c) Multiplication of matrices.

d) Display even position elements of the first matrix.

e) Exit.

Enter your choice: 2

Subtraction of matrices is:

0 0 0

0 0 0

0 0 0

a) Addition of matrices.

b) Subtraction of matrices.

c) Multiplication of matrices.

d) Display even position elements of the first matrix.

e) Exit.

Enter your choice: 3

Multiplication of matrices is:

30 36 42

66 81 96

102 126 150

a) Addition of matrices.

b) Subtraction of matrices.

c) Multiplication of matrices.

d) Display even position elements of the first matrix.

e) Exit.

Enter your choice: 4

Displaying even position elements of matrix 1 :

1 3

5

7 9

a) Addition of matrices.

b) Subtraction of matrices.

c) Multiplication of matrices.

d) Display even position elements of the first matrix.

e) Exit.

Enter your choice: 5

invalid choice!

Enter the order of the matrix (n x n): 3 3

Enter the elements of the matrix:

1 2 3 5 7 8 6 9 Saddle Point Found: 6 at position (3, 2)