Part I:

A. (5 Marks) Indicate whether each of the following statements is **True** or **False:**

	TRUE/FALSE	Statement
1.		The following `for` loop prints ***. **for (int i = 3; i>=1; i--)** **cout<<"*";**
2.		The following loop does NOT print the word **Programming**. **int x = 7;** **do** **{** **cout<<"Programming";** **} while(x > 7);**
3.		The number of iterations of a counter-controlled loop is known in advance.
4.		A value-returning function can return only one value via the return statement.
5.		Variables declared inside a **main()** function can be accessed by all functions in the program file.

B. (5 Marks) **Circle** the letter of the choice that best completes the statement or answers the question.

1. Which of the following function prototypes correctly expect an array as the first argument?

   a. void func(int array, int size);
   b. void func(int& array, int size);
   c. void func(int array[], int size);
   d. void func(array[], int size);

2. Assuming **X** is a global variable, which of the following is the correct use of **::** operator?

   a. `cout<<X::;`
   b. `cout<<::X + 8;`
   c. `cout<<::X:: + 4;`
   d. `cout<<9 + X::;`

3. Which of the following is a legal C++ function definition?

a.
```
void func(int a, double b &)
{
      cout<<a<<" "<<b-a<<endl;
}
```
b.
```
void func(int 'a', double &b)
{
      cout<<a<<" "<<b<<endl;
}
```
c.
```
int func(int &a, double &b)
{
      cout<<a<<" "<<b<<endl;
}
```
d.
```
int func(int a, double& b)
{
      cout<<a<<" "<<b<<endl;
      return b-a;
}
```

4. What is the output of the following C++ code?

```
int X[ ] = {1, 2, 3, 4, 5, 6, 7, 8, 9};
for(int j=2; j<=5; j++)
      cout<<X[j]<<" ";
```

a. 0 1 2 3
b. 1 2 3 4
c. 2 3 4 5
d. 3 4 5 6

5. What is the output of the following program segment?

```
char s1[15]="Program";
char s2[] ="Program";.
cout<<strcmp(s1,s2);
```

a. -1
b. 0
c. 1
d. s1 and s2 cannot be compared

Part II. (9 marks)

1. **(2 Mark)** Circle each line in the following program that will result in a syntax error and the program will not compile.

```
#unclude<iostream>

using namespace std;

void swap(double, double);

int main()
{
    double x = 15.3, y = 35.1;
    void swap(x, y);
    cout<<x << y << endl;
    return 0;
}

void swap(double & a, double & b);
{
    double t;
    t = a;
    a = b;
    b = t;
    return a;
}
```

2. **(3 Marks)** write the output of the following program.

```
#include<iostream>

using namespace std;

int main()
{
    const int size=4;
    int list[size];
    int *p, *q, x=5, y=8, i;

    p = &x;
    q = &y;
    q = p;
    cout<<*p<<endl;
    cout<<y<<endl;

    p = list;
    for(i=0; i<size; i++)
    {
        *(p+i) = i+3;
        cout<<list[i]<<endl;
    }

    return 0;
}
```

Program output:

3. **(4 Marks)** Write the output of the following program.

`#include <iostream>`
 	
`using namespace std;`
 	
`int w = 8;`
 	
`void f1(int&, int );`
`int f2(int &);`
 	
`int main()`
`{`
`  int x = 5; int y = 10, w = 2;`
 	
`  f1(x, y);`
`  cout << "In main: "<<x<<" "<<y<<endl;`
`  y = 3;`
`  cout <<"In main: "<<f2(y)<<endl;`
`  cout <<"In main: "<<::w<<endl;`
 	
`  return 0;`
`}`
 	
`void f1 (int &a, int  b)`
`{`
`  int t = a;`
`  a = b ;`
`  b = t ;`
 	
`  cout <<"In f1: "<<a<<" "<<b<<endl;`
`}`
 	
`int f2 (int& n)`
`{`
`  cout<<"In f2: "<<n<<" "<<w<< endl;`
`  w++;`
`  return w;`
`}`
	Program output:

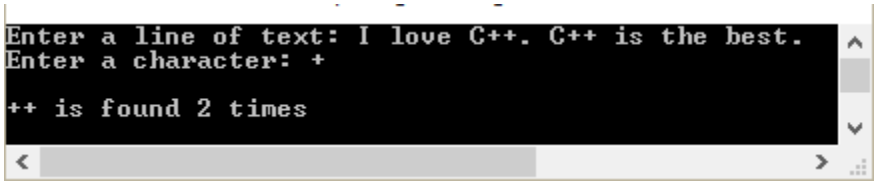
Part III

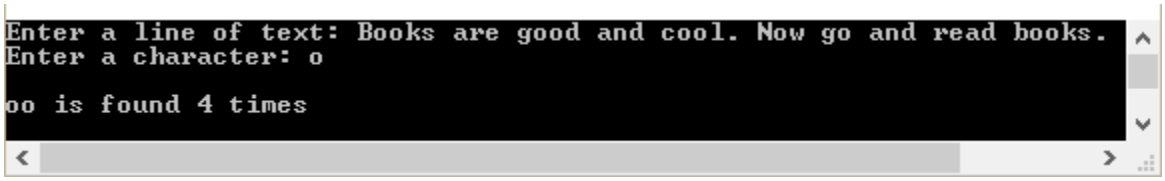
1. **(5 Marks)** Write a C++ program that reads a C-String (char array) **line** of the size 80 and a char **ch**. Program will display the count of two consecutive occurrences of **ch** in **line**.

   A consecutive occurrence is found if the character is found at a location and the immediate next location, e.g. if **ch** is found at the location 4 and also found at location 5 then it will define a consecutive occurrence.

   Display the count of all two consecutive occurrences, for example, if **ch** is **o** then program will find how many times **oo** appears in **line** as shown in the sample output.

Sample Input and Output:





Write the program on the next page:

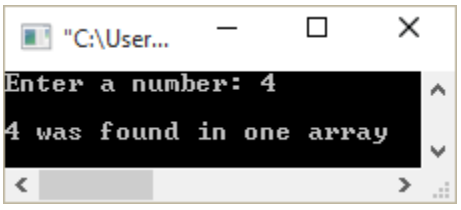
```
#include <iostream>
#include <cstring>
using namespace std;

int main()
{









    return 0;
}
```

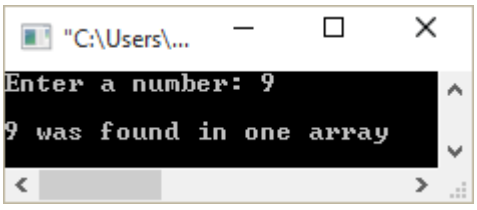
2. **(5 Marks)** Write the function **findInArrays** that will have the following parameters: int 1D array x, int 1D array y, int N, and int size. The function will find if N is in the array x, or y, or both. It will display the message as shown in the sample output if N is found in one of the arrays, both arrays or not found in any array.

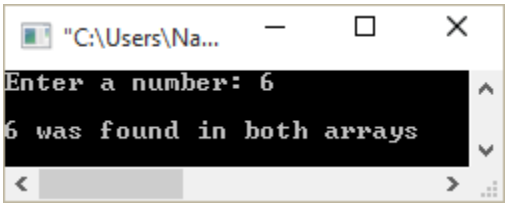
The **main()** function is provided. You only need to write the function on the next page.

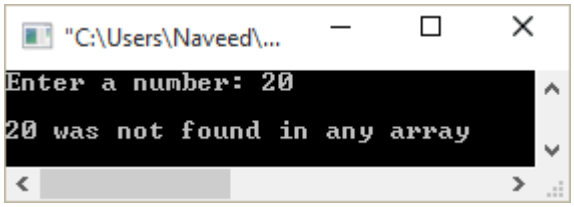
```
int main()
{
    int x[] = {2, 4, 6, 1};
    int y[] = {6, 3, 9, 8};

    int N, count;

    cout<<"Enter a number: ";
    cin>>N;

    findInArrays(x, y, N, 4);

    return 0;
}
```

Sample Input and Output:









Write the function on the next page:

Write the function here:

3. (6 Marks) Complete the following C++ program.

a. Write the function **SameSum** that will have the following parameters: int 2D array Table (2 Columns), int rowSize, and int colSize. The function will return **true** if the all the rows of the Table have the same sum otherwise it will return **false**. **(3 Marks)**

The **main()** function is provided and assume that the **Print** function is available. Only write the function on the next page.

```
int main()
{
    int Table1[3][2] = { {1, 3}, {2, 2}, {4, 0} };
    int Table2[3][2] = { {4, 5}, {3, 6}, {5, 2} };
    bool isMagic;

    cout<<"Table #1:\n";
    Print(Table1, 3, 2);

    isMagic = SameSum(Table1, 3, 2);

    if ( isMagic == true )
        cout<<"All rows have the same sum.\n";
    else
        cout<<"Rows have different sums\n";

    cout<<"\nTable #2:\n";

    Print(Table2, 3, 2);

    isMagic = SameSum(Table2, 3, 2);

    if ( isMagic == true )
        cout<<"All rows have the same sum.\n";
    else
        cout<<"Rows have different sums.\n";

    return 0;
}
```

Sample Output

```
Table #1:
  1  3
  2  2
  4  0

All rows have the same sum.

Table #2:
  4  5
  3  6
  5  2

Rows have different sums.
```

Write the function here:

b. Write the function **SumRowCol** that will have the parameters: int 2D array Table, int rowSize, int colSize, int R, int C. The function will return two values: the sum of the elements in Row R and Column C of the Table. **(3 Marks)**

The **main()** function is provided and assume that the **Init** and **Print** function is available. Only write the function on the next page.

```
int main()
{
    int Table[3][3], R, C, sumR, sumC;

    srand(time(0));

    Init(Table, 3, 3);
    Print(Table, 3, 3);

    cout<<"Enter a row and col: ";
    cin>>R>>C;

    SumRowCol(Table, 3, 3, R, C, sumR, sumC);

    cout<<endl;
    cout<<"Sum of Row "<<R<<" = "<<sumR<<endl;
    cout<<"Sum of Col "<<C<<" = "<<sumC<<endl;

    return 0;
}
```

Sample Output:

```
  1  3  6
  4  1  4
  6  6  2

Enter a row and col: 1 2

Sum of Row 1 = 9
Sum of Col 2 = 12
```

```
  0  5  1
  5  4  5
  2  4  5

Enter a row and col: 2 0

Sum of Row 2 = 11
Sum of Col 0 = 7
```

Write the function on the next page:

Write the function here: