INSTITUTE OF SPACE TECHNOLOGY

DEPARTMENT OF ELECTRICAL ENGINEERING

108408– Programming Language

**Assessment: One Hour Test**   **Instructor: Dr. Komal Nain Sukhia**

Max Points: 30

Max Time Allowed: Solving 1 Hour + Uploading 15 Minutes

Question 1 **C++ Output Questions**   **[CLO-1] Marks = 5**

1. The only integer that can be assigned directly to a pointer is _____.

2. **ios::trunc is used for ?**

A. If the file is opened for output operations and it already existed, no action is taken.
B. If the file is opened for output operations and it already existed, then a new copy is created.
C. If the file is opened for output operations and it already existed, its previous content is deleted and replaced by the new one.
D. None of the above

3. **Consider this figure.**


```
   6 ◄────────┬──────┐        12

   x        newptr       y
```
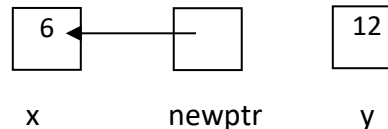
State whether the following statements are true/false.

1. 'y is accessible using direct as well as indirect addressing'. If true, write statements for changing the value of y to 6 using both modes. If false, write statement for changing the value of y to 6 using the applicable mode. You can only work with the variables shown in figure.

2. 'y and x are aliases since both contain the value 6 now'.

Question 2 **C++ Output Questions**   **[CLO-1] Marks = 5**

Write and explain output of the following code segment;

1. **Code Segment:**

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

void main()
{
        const int size=10;
        int b[size]={0,1,2,3,4,5,6,7,8,9};
        int *bptr1=&b[0];
        int *bptr2=&b[size-1];
        for (int i=0;i<size;i++)
        {
                if (i==0)
                        b[i]=b[i]*b[i];
                else
                        b[i]=b[i]*b[i]-b[i-1];
                cout<<b[i]<<endl;
        }
        cout << bptr2[-3] << endl;
        cout << * bptr1 + 4*2 << endl;
        bptr2-=4;
        cout << * (bptr2 - 2) << endl;
        cout << bptr2 - bptr1 <<endl;
        while (bptr1!=bptr2)
                bptr1++;
        cout<< bptr1[0] << " " << *bptr2 << endl;

}
```

Question 3   [CLO-2] Marks = 8

Write a C++ program for maintaining the library record of 100 books. Each record is composed of 6 fields which include book's library number, book's name, author's name, edition number, year of publishing and price of the book. Choose appropriate data types to represent each field. Your program should prompt the user to populate 10 records of this database with some values. Prompt the user to enter an integer from 1 to 6 (each number indicates a specific field of the library books' records). Based on the specific number entered, your program should ask the user to enter a specific value to search in the library database. If the specific value is found, your program should display the corresponding record(s), otherwise it should print 'No match found' on the screen. Your program should have the following interface.

```
You can search for a specific book. Press an integer from 1 to
6 to specify the search item.
1. Book's Library Number
2. Book's Name
3. Author's Name
4. Edition Number
5. Year of Publishing
6. Price of Book
```

Assume the user enters the integer 3.

```
3

Enter the Author's Name :
XYZ

1 Book Found.
Book 1: A Guide to C Language Programming by XYZ, Edition 1
published in 2013. Price 250
```

Question 4   [CLO-1] Marks = 7

Write a C++ function called **ReplaceNot** that takes a string as a parameter, searches all three letter words in the string passed as argument and replaces them with '**Not**'. Apply all the checks and cases. For example, if the passed string is: "**Every boy has a pen**". Your function should modify this string to: "**Every Not Not a Not**". Use this function in your main program to verify that your function works correctly.

Question 5   [CLO-1] Marks = 5

Suppose that scores is an array of 10 components of type double, and:
scores = {2.5, 3.9, 4.8, 6.2, 6.2, 7.4, 7.9, 8.5, 8.5, 9.9}
        The following is supposed to ensure that the elements of scores are in non-decreasing order. However, there are errors in the code. Find and correct the errors.
for (int i = 1; i <= 10; i++)
if (scores[i] >= scores[i + 1])
cout << i << " and " << (i + 1)<< " elements of scores are out of order." << endl;