National University of Computer and Emerging Sciences

| School of Computing | Spring 2019 | Islamabad Campus |

CS118 Programming Fundamentals

Friday May 17, 2019

Course Instructor

Serial No:

Final EXAM
Total Time: 180 Minutes
Total Marks: 120

_____
Signature of Invigilator

_____ _____ _____ _____

Student Name   Roll No   Section   Signature

DO NOT OPEN THE QUESTION BOOK OR START UNTIL INSTRUCTED.

Instructions:

1. Attempt on question paper. Attempt all of them. Read the question carefully, understand the question, and then attempt it.
2. Please read the complete paper before attempting any question and manage your time intelligently.
3. After asked to commence the exam, please verify that you have **Twenty one (21)** different printed pages including this title page. There are total of **3 questions**.
4. Use permanent ink pens only. Any part done using soft pencil will not be marked and cannot be claimed for rechecking.
5. Use **proper Syntax** while writing code and make sure that your code is legible. Failing to do so can cost you marks.

	Q-1	Q-2	Q-3	Total
Marks Obtained				
Total Marks	60	30	30	120

National University of Computer and Emerging Sciences

School of Computing   **Spring 2019**   **Islamabad Campus**

Question I …......……………..…………………………………..……..………… (60 Marks)

Write the output of the given C++ code segments. Please note there are no syntax errors in the given set of codes.

1) What will be the output of the following code?   [2]

```cpp
int main()
{
int z = 5, j = 7, k = 6, n = 3;
cout << z + j % k + k * n - 15 << endl;
cout << z % n + 5 << endl;
return 0;
}
```

2) What is the output of variable counter in the following code?   [2]

```cpp
int i = 12, counter = 5;
      while ((i - 1))
      {
            ++counter;
            i--;
      }

cout<<counter;
```

3) What is the output of the following code?   [4]

```cpp
int fun(int m, int n)
{
      while (n != m)
      {
            if (n > m)
                  n = n - m;
            else if (m > n)
                  m = m - n;
      }
      return n;
}
int main()
{

      cout << fun(88, 33) << endl;
      cout << fun(172,140)<<endl;

return 0;
}
```

What is the purpose of above function fun()?

4) **What is the output of the following code?**   [2]

```
void main()
{
        int x = 5;
        int y = 10;
        int z = ++x * y--;
        cout<<(z+y);
}
```

5) **What is the output of the following code?**   [2]

```
void main()
{
        int y = 981;
        if(y % 4 == 0 && y % 100 != 0 || y % 400 == 0)
                cout<<"Condition is True";
        else cout<<"Not True";
}
```

6) **What is the output of the following code?**   [2]

```
void main()
{
        int z, x=5, y=-10, a=4, b=2;
        z = x++ - --y * b / a;
        cout<<z;
}
```

7) **What is the output of the following code?**   [2]

```
void main()
{
        int a = 10, b = 0xFF;
        b = a++ - a++;
        cout << ++b;
}
```

8) **What is the output of the following code?**   [2]

```
int x=10;

void main()
{       int x=20;
        cout<<::x++;
        cout<<x+::x;
}
```

9) **What is the output of the following code?** [2]

```cpp
void main()
{
        int i=0, n = 0;
        if ((i < 1) && (++i < n))
        {
                cout << "Condition True!";
        }
        else cout << "Not True!";
}
```

10) **What will be the output of the following code?** [3]

```cpp
int counter (int value)
{
 static int count =0;
 count = count +value;;
 return count;
 }
int main()
{
  int i , j;
  for (i=0; i <=5; i++)
    j = counter(i);
   cout<<"J =" <<j<<endl;
 return 0;
}
```

11) **What will be the output of the following code?** [4]

```cpp
int  mystery(int x, int  n)
{
        int val;
        val = 1;
        if (n >= 0)
        {
                if (n % 3 > 1)
                        val = val * x;
                else
                        val = val * 2;
        }
        return val;

}
void main()
{
cout << "The mysterious value is: " << mystery(10, 3);

}
```

12) What will be the output of the following code?   **[2]**

```
int main()
{
    int  a[5] = { 1,2,3,4,5 };
    int *ptr = a + 5;
    cout << *(a + 1) << " , " << *(ptr - 1);
return 0;
 }
```

13) What will be the output of the following code?   **[3]**

```
void function(int b[][3])
{
        ++b;
        b[1][1] = 9;
}



int main()
{
        int a[3][3] = { { 1,2,3} ,  { 4,5,6} , {7,8,9} };
        function(a);
        cout << a[2][1];
        return 0;
}
```

14) What will be the output of the following code?   **[3]**

```
void e(int n)
{
        static int x = 3;
        if (n > 0)
                cout << n << " "<< x<<endl;
        n--;
        x--;
}
int main()
{
        int a;
        a = 10;
        e(a--);
        e(a--);
        e(a--);

}
```

15) What will be the output of the following code? **[5]**

```
void f1(int *, int*);
void f2(int *, int);
int  main()
{
        int a;
        int b;
        a = 3;
        b = 5;
        f1(&a, &b);
        cout << a << "," << b << ",";
        f2(&a, b);
        cout << a << "," << b;


}
void f1(int* p, int *q)
{
        int tmp;
        tmp = *p;
        *p = *q;
        *q = tmp;
}
void f2(int* p, int q)
{
        int tmp;
        tmp = *p;
        *p = q;
        q = tmp;
}
```

16) What will be the output of the following code? **[4]**

```
int main() {
        int i, j, m, answer;
        m = 0;
        j = 4;
        while (m < 5) {
                for (i = 0; i < j; i++) {
                        answer = i * m;
                        cout << answer;
                }
                m = m + 1;
                cout << endl;
        }
        return 0;
}
```

17) What will be the output of the following code? [4]

```
int main ( )
{
char *s[4] = {"black", "white", "yellow", "violet"};

        cout<<(*(s+1)+2)<<endl;
        cout<<*(*(s+2)+3);
return 0;
}
```

18) What is the value of Arr[7][9]; in the sample code below? [4]

```
void main()
{
        int i, j;
        int counter = 0;
        int Arr[8][9];
        for (i = 0; i < 9; i++)
                for (j = 0; j < 8; j++)
                {
                        Arr[j][i] = counter;
                        ++counter;
                }
        cout << Arr[7][8];

}
```

19) What is the output of the following code? [4]

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


int main()
{
   char s1[20] = "Final ";
   char s2[] = "Examination ";
   cout << "s1 = " << s1 << "\ns2 = " << s2;
   strncat(s1, s2, 6);
   cout << "\nS1 = " << s1 << " S2= " << s2 ;
return 0;
}
```

20) What is the output of the following code?   **[4]**

```cpp
int check1(char *x, char *y)
{
    return strcmp(x,y);

}

int check2(char *x, char *y)
{
    return strncmp(x,y,3);

}

int main()
{
    char e1[] = "Alpha";
    char e2[] = "Bravo";
     char e3[] = "Alpak";

     cout<<check1(e1, e2)<<endl;
     cout<<check1(e1, e1)<<endl;
     cout<<check2(e1, e3)<<endl;

return 0;
}
```

National University of Computer and Emerging Sciences

Question II …........…………..…………………………….…..……….. **(20 + 10 = 30 Marks)**

   A) Select appropriate option. (MCQ's)   [**20 X 1 =20**]

   1. Which one of the following C++ operators is right associative?

   a) =
   b) ,
   c) []
   d) ^
   e) ?:

   2. What will the output of the sample code be?

int arr[10] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}; int *ptr = arr + 2; cout<<ptr[7];	a) 6 b) 7 c) 8 d) 9

   3. What will be printed when the sample code above is executed?

char *buffer = " Hello World";
char *ptr = buffer;
ptr += 5;
cout<<ptr<<endl;
cout<<buffer;
	a) HelloWorld
	   World
	
	b) World
	   WelloWorld
	
	c) World
	   World
	
	d) HelloWorld
	   HelloWorld
	
	e) World
	   HelloWorld

   4. What value does testarray[2][1][0] in the sample code contain?

int testarray[3][2][2] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12};	a) 3 b) 5 c) 7 d) 9 e) 11

5. Consider implementing a function to dynamically allocate an array of integers and set all its elements to zero: Which of the following choices for the blank preceding the formal parameter A is best?

```
void ZeroIt(_____ A, const int Size) {

    A = new int[Size];
    for (int Idx = 0; Idx < Size; Idx++) {
        A[Idx] = 0;
    }
}
```

a) int*
b) int* const
c) int*&
d) const int* const
e) const int*

6. Not initializing variables in a C++ program can lead to _____.

    a) syntax errors

    b) logic errors

    c) linking errors

    d) runtime errors

7. Array passed as an argument to a function interpreted as _____.

    a) value of the first element of the array

    b) address of the first element of the array

    c) number of elements of the array

    d) size of the array in bytes

8. The portion of a function prototype that includes the name of the function and the types of its arguments is called the _____.

    a) function header

    b) function call

    c) function signature

    d) function parameter list

9. A _____ should be used to declare the size of an array because it makes the program more scalable.

    a) constant variable

    b) local variable

    c) null character

    d) pointer variable

10. The naming conflict arising from the definition of variables with same names in different scopes has been solved in C++ using _____.
    a) scope resolution operator

    b) type casting

    c) storage classes

    d) namespaces)

11. The compiler refers to the _____ to check that calls to that function are correct in terms of the number, type and order of the arguments that the function takes.
    a) function header

    b) function definition

    c) function parameter list

    d) function prototype

12. Global variables declarations are placed _____.
    a) Inside a block

    b) outside all function definitions but inside main

    c) outside any function definition

    d) outside the headers of control statements

13. In C++, an empty parameter list is specified by writing either void or _____.
    a) Nothing in parentheses

    b) NULL in parentheses

    c) 0 in parentheses

    d) omitting the parentheses

14. A function receiving an address as an argument must have a/an _____ as its corresponding parameter.
    a) Identifier

b) single dereferenced value

c) static variable

d) pointer

For questions 15 to 19, refer to the following code segment;

```
# include <iostream>
using namespace std;
f_1();
f_2(int,int);
f_3();
f_4(int);

main()
{
        int x=6;                        //Line 1
        f_1();                          //Line 2
        f_3();                          //Line 3
        f_4(x);                         //Line 4
}
f_1()
{
        int x=7, int y=5;
        f_2(x,y);
}
f_2(int x, int y)
{
        cout<<"X is "<<x<<" and Y is "<<y<<" in function f_1()."<<endl;
        f_3();
}
f_3()
{
        static int x=5;
        cout<<"A static X found in one of the functions with the value
"<<x<<"."<<endl;
        x++;
        f_4(x);
}
f_4(int x)
{
        cout<<"Value of X in f_4() is "<<x<<"."<<endl;
}
```

15. The variable 'x' defined in Line 1 has;
   a) global scope since it is defined outside the functions' f_1, f_2,f_3 and f_4 bodies
   b) function prototype scope since it is used in calling function f_4
   c) file scope since it is known in all functions of the file
   d) local scope since it is defined inside the body of main

16. How many activation records will be pushed onto the function call stack as a result of execution of Line 2?
   a) One
   b) Two
   c) Three
   d) Four

17. During execution of Line 2, after two activation records have been popped off the function call stack, the stack top will contain the activation record of;
   a) f_1
   b) f_2
   c) f_3
   d) f_4

18. During execution of Line 3, the following statement will be printed on screen;
   a) A static X found in one of the functions with the value 5.
   b) A static X found in one of the functions with the value 6.
   c) A static X found in one of the functions with the value 7.
   d) A static X found in one of the functions with the value 8.

19. When this program is executed, six statements will be printed on screen. Which two statements will be identical?
   a) Statements 2 and 4
   b) Statements 3 and 5
   c) Statements 3 and 6
   d) Statements 5 and 6

20. An array is a consecutive group of memory locations that all have the same address that is the name of the array.
   a) True
   b) False

B) Identify what's wrong with the following set of code segments and correct it. Please assume that all variables have been defined.   [10]

I. int main()
   { const double PI;
   int n;
   PI = 3.14159265358979;
   return 0;
   }

II. while (n <= 100)
          sum += n*n;

III. float x = 3.14159;
    float* p = &x;
    short d = 44;
    short* q = &d;
    p = q;

IV. if (response = "yes" or "YES")
     cout << "You said yes.";

V. if (x < y < z)
   cout << x << " < " << y << " < " << z << endl;

Question III …........……………...……………...……...……..……….. **(10 + 10 + 10 = 30 Marks)**

A) Write a complete C++ code that inputs a character array (string) from a user. After taking input it calls a function *Reverse()* that reverses the string word by word as shown below:

Original Character Array:

Raining in summers makes your life pretty

After calling Function *Reverse()*:

pretty life your makes summers in Raining

Note: *You have to use pointers in the function. You can use any builtin Cstring function of your choice for this task.*

B) Write a function highLow that takes an integer as an argument and returns whether or not the number has alternating "high" and "low" digits. 0 through 4 are "low" digits and 5 through 9 are "high" digits. Your function should return true if the number passed alternates between "high" and "low" digits, and false if not. You may assume the number passed is positive. If the number passed consists of a single digit, highLow should return true.

Note: highLow returns true if the number alternates starting with a "high" digit or starting with a "low" digit. What is important is that the digits alternate. For example, both 9292 and 2929 passed to highLow should return true.

C) Write a function named *reverseChunks* that accepts three parameters, an array of integers a, its size and an integer "chunk" size s, and reverses every s elements of a. For example, if s is 2 and array a stores {1, 2, 3, 4, 5, 6}, a is rearranged to store {2, 1, 4, 3, 6, 5}. With an s of 3 and the same elements {1, 2, 3, 4, 5, 6}, array a is rearranged to store {3, 2, 1, 6, 5, 4}. The chunks on this page are underlined for convenience.

If a's length is not evenly divisible by s, the remaining elements are untouched. For example, if s is 4 and array a stores {5, 4, 9, 2, 1, 7, 8, 6, 2, 10}, a is rearranged to store {2, 9, 4, 5, 6, 8, 7, 1, 2, 10}.

It is also possible that s is larger than a's entire length, in which case the array is not modified at all. You may assume that s is 1 or greater (an s of 1 would not modify the array). If array a is empty, its contents should remain unchanged.

The following table shows some calls to your method and their expected results:

Arrays and Call	Array Contents After Call
`int a1[] = {20, 10, 30, 60, 50, 40};` `reverseChunks(a1,6, 2);`	`{10, 20, 60, 30, 40, 50}`
`int a2[] = {2, 4, 6, 8, 10, 12, 14, 16};` `reverseChunks(a2,8, 3);`	`{6, 4, 2, 12, 10, 8, 14, 16}`
`Int a3[] = {7, 1, 3, 5, 9, 8, 2, 6, 4, 10, 0, 12};` `reverseChunks(a3,12, 5);`	`{9, 5, 3, 1, 7, 10, 4, 6, 2, 8, 0, 12}`
`int a4[] = {1, 2, 3, 4, 5, 6};` `reverseChunks(a4,6, 8);`	`{1, 2, 3, 4, 5, 6}`
`int a5[]= {};` `reverseChunks(a5,0, 2);`	`{}`