3333333333333333333333333333333333333333333333333333

National University of Computer & Emerging Sciences
FAST-Karachi Campus
CS218- Data Structures (Fall 2019)
Quiz#1

Dated: September 16, 2019 Marks: 25
Time: 20 min. Std-ID: ____Sol_____

Question No. 1 Indicate TRUE or FALSE. [10]

1. It is not possible to achieve inheritance of structures in c++. FALSE
2. A destructor of a class can have one parameter. FALSE
3. All objects of a class shared code and non-static methods. TRUE
4. Static function of a class cannot be called by an object. TRUE
5. A derived class need to implement all virtual functions of a base class. FALSE
6. In multi-level classes the order of constructors is same as order of derivation. TRUE
7. Friend function of a class is not allowed to access private data. FALSE
8. We can overload a function in three different ways. TRUE
9. We can override a function in only one possible way. FALSE
10. There are 3 different type of constructors in c++. TRUE

Question No. 2 Write a function for finding a Peak Element in an arry. The Peak element is an element which is not smaller than both of its neighbors (if exists). [5]

```
int FindPeak(int a[], int size) {
        int s = size;     int i=0;
        if (size == 1) return a[0];
        if (size == 2) return (a[0]>a[1]? a[0]: a[1]);
        else {
          for(i=1; i<s; i++)
          {  if ((a[i-1]<=a[i]) && (a[i]>=a[i+1]))  return(a[i]);        }
           }
}
```

Question No. 3 Explain each question by giving 1-2 lines of explanation. [10]

1. **There are int a and int b, you need to produce there sum without using + operator. Hint: a single c/c++ instruction (expression) required.**

   Sum= -(-a-b);

2. **Is inline function more efficient than normal functions in c/c++?**

- Yes, they are more effective and efficient concept, there is no function call overheads as the function is used with the code.
- No overhead for parameters passing on stack and returning value
- Compiler can do a lot of different optimizations.

3. **What are the best practices for defining an assignment operator of a new type?**

   - An assignment operator must check for self-assignment
   - The earlier memory of left hand side object is to wipe off
   - New memory should be grabbed exactly of the same size of right hand side object
   - Member-wise copy should be done for all members.

4. **If int *PtrX=0; int **dPtrX=0; int x=1; int y=0; what will be the errors in the following code:**

   PtrX=x; // Error: PtrX is an address but x is not an address
   ***PtrX=&x; // Error: *PtrX is a value pointed by PtrX and &x is an address**
   dPtrX=&PtrX; // No Error

5. **Why multiple inheritance is considered bad?**
   - Multiple inheritance is available in C/C++ is an important feature of the language.
   - The use of multiple inheritance gives rises to some ambiguity in code which can be rectify with proper understanding of the concept for multiple inheritance.
   - Overall, it is not a bad idea.