**National University of Computer & Emerging Sciences, Karachi** **Spring 2021**S.M.HassanAli (20k-1052) Assignment 2

1-Why it is important to apply abstraction when solving a problem?

* Abstraction is the first step towards thinking process and it helps to simply the model by hiding the irrelevant information. It allows us to focus on the object rather than what goes behind. It tells us how deeper the things are related and seek to understand the relations.

2- How encapsulation secures data?

* Encapsulation hides our data inside of a unit. For instance, all our data is encapsulated under class and until and unless the objected is not created the data cannot be accessed. Furthermore, it restricts the access of data and function in a way that those can be kept in private and thus making the data even more secure.

3- What is a difference between a default and null contractor?

* A default constructor is created automatically in the absence of user defined constructor, a do-nothing constructor that only assigns memory to the object. Meanwhile, Null constructor is a user defined constructor which also assigns the memory to the object and it takes no parameter.

**class Student{**

**public:**

**Student(){ //definition of a default constructor**

**cout<<"this is a default constructor";**

**}**

**};**

**main(){**

**Student s1; //calling of constructor occurs as soon as object is created**

**}**

4- Why constant data and functions are important?

* They are used because the values that are stored in our program cannot be changed by mistake or by some unauthorized user. As we know that our code is usable and changes can be made for other programs using the same code. So a constant object will not be changed as well the constant data can only be used by constant function, a limitation in c++.

**class Student{**

**const int rollno; //constant data member that can be used in every object**

**public:**

**Student():rollno=1052{} //constant data members can only be initialized through member initializer**

**};**

**main(){**

**Student h1;**

**}**

5- Differentiate between a shallow and deep copy. Write a small code to explain this concept.

* Deep copy happens when the user defined copy constructor is created. The newly created objected that has been passed a parameter into an old object gets a new memory
* In shallow copy constructing, a new collection of objects is created and then populating it with references to the child objects in the original.

**class haha{**

**private:**

**int a;**

**public:**

**haha(int b){**

**a=b; }**

**haha(haha &h){ //a user defined copy constructor**

**cout<<"Here the deep and shallow copy will occur"<<endl;**

**} };**

**int main(){**

**haha h(3);**

**haha h1(h); //new address assigned to object h1 as it is deep copy**

**haha h2=h1; //same address assigned to the object as it is shallow copy }**

6-Explain why constructor and destructor do not have any return type, although they are functions?

* At the run time the space is allocated to store an object instance in the memory. It holds all the data members and the functions and after which the constructor is called for the initialization process, it initializes the content it occupies. So, when the constructor is exit, the run time creates the newly-created instance. It is being called by the memory allocation and the object initialization which shows that it is not directly connected to our code so it is not return a value just like a function does. Same goes with destructor, it just deallocates the memory hold by constructors and has no return type as well as it takes no parameters.