Software Development fundamentals - II

Tutorial - 10

OOPS Concepts

SHUBHAM GARG
9919103057
BATCH: F2

1. Specify the difference between procedural language and object oriented language.

Ans-

Procedural oriented Programming	object oriented Programming
* Here, program is divided into small parts called functions.	* Here, program is divided into small parts called Objects.
* follows top down approach	* follows bottom up approach.
* No access specifiers in this programming	* This have access specifiers like private, public, protected etc.
* Adding new data and functions is not easy.	* Adding new data and function is easy.
* We cannot hide anything so it is less secure	* It provides data hiding. so, it is more secure.
* Overloading is not possible	* Overloading is possible
* Here function is more important than data	* Here, data is more important than function.
Ex: C, FORTRAN, Pascal etc	Ex: C++, Python, Java, C# etc.

2. Define the terms:

a. Encapsulation

It is defined as wrapping up of data and information under a single unit. In object oriented programming, it is defined as binding together the data and the functions that manipulates them.

b. Inheritance

The capability of a class to derive properties and characteristics from another class is called Inheritance. It is one of most important feature of OOPs. The class that inherits properties from another class is Sub class and the class whose properties are inherited is called Super class.

c. Abstraction

Abstraction means displaying only essential information and hiding the details. Data Abstraction refers to providing only essential information about the data to the outside world, hiding the background details or implementation.

d. Polymorphism

The word Polymorphism means having many forms. Polymorphism is the ability of a message to be displayed in more than one form. It is of two types namely-

1. Compile Time
2. Run Time.

3. What is Inline functions?

Ans- Inline function is a function in C++ that is expanded in line when it is called. When the inline function is called whole code of the inline function gets inserted or substituted at the point of inline function call. This substitution is performed by C++ compiler at compile time.

Syntax:

```
inline returntype function_name(parameters)
{
  // Body of function.
}
```

Shubham Garg 9919103057

4. Explain function Overloading in Object Oriented Programming.

Ans- function overloading is a feature in C++ where two or more functions can have the same name but different parameters. function Overloading can be considered as an example of polymorphism feature in C++. for example -

```
#include <iostream>
using namespace std;
void print(int i)
{
    cout <<" Here is int" << i << endl;
}
void print(double f)
{
    cout<< "Here is float" << f << endl;
}
int main()
{
    print(10);
    print(10.10);
    return 0;
}
```

Output:
Here is int 10
Here is float 10.10

5. Run following C++ programs and analyze the output.

Outputs

a) 0

b) 4, 3

c) 38, 35 32 29 26 23 20 17 14 11 8 5 2