***The Characteristics of an Object Orientated Paradigm***

Encapsulation is a fundamental concept in object-orientated programming, it depicts the idea of groups data and methods that work on the data within one unit, for example a class. Additionally, this method/concept is used to conceal the internal representation, or a state of an object from the outside. This is normally called ‘information hiding’. The base idea of this method is easy to understand, if for example you have an attribute that is invisible to the outside of an object and you then group it with methods that will add read or write access to it; you can then hide specific information and also control access to the state of the object.

Generic programming will provide similar abstraction the same as object-orientated programming, but the difference is within object-orientated programming the polymorphism will happen at runtime whereas within generic programming the polymorphism happens at compile time.

Within programming a template is known as a generic class or unit of source code, that is sued as the basis for unique units of code. Within C++ which is an object-orientated computing language, there are default template libraries known as Standard Template libraries where programmers can choose individual template classes they want to modify.

Method overriding, within computer science and object-orientated programming is a language feature that allows a child class or subclass to provide a specific implementation of a method that is provided by one of its parent’s classes or super classes also some languages will allow developers and programmers to prevent a method from being overridden.

Method overloading, is used within programming languages that allow the programmer to enforce type-checking in function class during compilation. When a method is overloaded the method that is chosen will be selected at compile time. generic programming oop

An object-orientated interface also known as an OOI, is the process of designing an interface that is built using object-orientated programming concepts. As part of the creation of an object-orientated interface it will incorporate either one or more than one interacting objects as the basis of the interface. An object-orientated interface is generally created for a general end user using an object-orientated user interface, which allows access and interaction with the system/software. The object-orientated interface approach is similar to object-orientated design and object-orientated programming in terms of interface design, in which a user interface’s (UI) components are defined and created through objects. To enable a functional interface each interface object interacts with back end objects as well as other objects.

An object within computer science refers to an instance of a class, the object be a collection of functions, data structures and variables. Additionally, a container within computer science refers to a data structure, abstract data type or a class whose instances are a collection of other objects. In simpler terms they hold objects in an organized way that follow specific access rules.

A base class is a class located within an object-orientated programming language from which other classes are derived. It aids the creation of other classes that have the option to reuse the code implicitly inherited from the base class exempting constructors and destructors. Finally, a developer or programmer can improve base class functionality by adding or overriding members that are relevant to the derived class.

A derived class is a class created from another existing class, the class from which the derived class is made/created through is the process of inheritance and is known and the base or super class. Derived classes are generally used for augmenting the functionality of base classes by adding or modifying the properties and methods of said class to match/suit the requirements of specialization needed for derived classes.

An abstract class is a common class, or type of object that is used for creating specific objects that tailor to its protocol, or the set of instructions and operations it supports. An abstract class will contain at least one abstract method and the method will not contain code within its base class, the code is added within the derived class.

A concrete class is a class that contains an employment for all its methods that were inherited from an abstract class or were implemented from an interface. Additionally, it does not define any abstract methods of its own, which means that an instance of the class can be allocated using the new keyword without having to implement any methods first.

A Constructor’s goal is to prepare an object for change/action, normally establishing initial values for all its data. Even though it performs an important action, it is still just another member function. Additionally, it can be given/passed information through its argument list and can be used to initialise it. For example, the name of the Constructor function must be the name of the class, so C++ knows it is a Constructor function.

A Destructor helps the Constructor by clearing resources which would be allocated within the Constructor by getting rid of the main class they are in but not erasing the data inside of the class. Additionally, it cannot return any value and gets called implicitly by the complier when its lifetime is over based on the storage class of that object.

Polymorphism is normally defined as something occurring in multiple forms. Within computer science, it is defined as the concept that objects of different types can be accessed using the same interface; where each type can show its own, independent implementation of the interface. Additionally, it is one of the more important core concepts of object-orientated programming. There is a simple test that can be done to determine if an object is Polymorphic or not; if the object successfully passes more than one ‘instanceof’ tests it is a Polymorphic object. Adding on to this, every object within Java is Polymorphic because Java classes extend the class ‘object’, meaning they all pass more than one ‘instanceof’ tests.