

Object-Oriented Definitions

Class - A class defines a type along with a collection of operations that are related to that type (this is commonly referred to as the blueprint that defines the variables and the methods common to all objects of a certain type).

Object - An instance/instantiation of a class. In other words - A specific realization of any class.

Class and object relationship can be visualized [here](#).

Method/Member Function - a method/member function is a function that is defined as part of a class. Member functions are sometimes referred to as methods.

Member Variable - A variable defined as part of a class.

Public - Members defined after a public specifier are accessible to all parts of the program. The public members define the interface to the class

Private - Members defined after a private specifier are accessible to the member functions of the class but are not accessible to code that uses the class. The private sections encapsulate (i.e., hide) the implementation.

Protected - Protected members are willing to share with its derived classes but wants to protect from general access. The protected specifier can be thought of as a blend of private and public:

this-> - The **this** keyword this refers to an instance variable (as opposed to a local or global variable) That is, a variable specific to an object.

Constructor - A special member function used to initialize objects. Each constructor should give each data member a well-defined initial value.

Initialize - Give an object a value at the same time that it is created. We use constructors to initialize objects (aka: a specific instance of a class).

dot (.) operator - Used to access methods or member variables

Header File - A mechanism whereby the definitions of a class or other names are made available to multiple programs. A program uses a header through a `#include` directive. (Commonly referred to as an interface).

Function/Method overloading - A programming concept that allows programmers to define two or more functions with the same name and in the same scope. This is a classic example of Polymorphism.

Compile time polymorphism - This type of polymorphism is achieved by function overloading of operator overloading.

Inheritance - Programming technique for defining a new class (known as a derived class) in terms of an existing class (known as the base class). The derived class inherits the members of the base class.

Base Class - Class you inherit from (Class with all the methods and member variables you that are “giving” to another class).

Derived class - Class that inherits methods and member variables. (Class that “gets” methods and member variables from base class.)

Recursion Definitions

Recursion

Recursion in computer science is a method of solving a problem where the solution depends on solutions to smaller instances of the same problem (as opposed to iteration).

Base Case

The base case returns a value without making any subsequent recursive calls. It does this for one or more special input values for which the function can be evaluated without recursion.

Reduction (or recursive) Step

The reduction step is the central part of a recursive function. It relates the value of the function at one (or more) input values to the value of the function at one (or more) other input values. Furthermore, the sequence of input values must converge to the base case.

Activation Record/Stack Frame

An object in our memory stack that contains the following information
1. *Local data to the callee*
2. *Return address to the caller and 3, Parameters of the callee*

Memoization - an optimization method that ensures a method doesn't run for the same inputs more than once by keeping a record of the results for the given inputs.