Muhammad Ali
muhammadallee@gmail.com

Why Object Oriented Programming?

Shortcomings of Structured Programming

Structured programming faced two short-comes, which can really simplify the programming and reusability of the software.

- ### Each function could access the global data

  At first it may seem like a benefit but actually it isn't. Suppose we want to make an Accounts Software for a university. Now, the accounts software needs read only information about students' registration information, for example the address.

  Now let's map this problem into programming, we know there's some information, which is shared among the functions that handle accounts and the functions that handle students' registration. And there might be a set of functions that don't need this shared information at all.

  To have shared data, we can use the global data which is available to *all* the functions. And here's where the problem lies. As the global data is available for all the functions, any function can access the global data and change it. So if there's an error in the value of a global data, we cannot guarantee which function caused it as all have access to it. The only solution is to hit and try every function and see which function actually caused the erroneous value. While this might be an easy task when you have a small number of functions, but usually software have hundreds of and may be thousands of functions in it. Now while having such huge number of functions at hand, figuring out which function caused the wrong value in the global data is not an easy task to do.

  So we see that we cannot map the real world problem exactly into structured programming and also debugging and maintenance became difficult.

- ### A function cannot save multiple values of data

  Whenever a function call is terminated, all of its local variables get destroyed. It cannot retain their values.

  Consider a situation where you might be drawing circles on the screen. You'll be having a function called Circle which takes center point and radius as parameters and then draws the circle at the given point with given radius. But as soon as the call terminates there's no track of the circle's center point and radius.
  One way is to have static variables in the function to track the previous state of the function.

  Note
  Static variables when declared in functions retain their values on next function call.

However, they don't fully solve the problem as they can only retain one value (i.e. last call's value) and cannot save multiple values against multiple function calls.

We must write our own code in order to save the multiple states, which unfortunately comes with many pitfalls. There must be some other way out!

Object Oriented Programming

Shift of Focus

Object oriented programming paradigm shifts the focus from 'verbs' to 'nouns'. Instead of looking into a problem and breaking it down into steps (to get the 'verbs' against which functions will be made), you look for 'nouns' and gather the 'information' and 'actions' that the particular 'noun' performs (or holds).

Object Oriented Programming deals with the shortcomings of the Structured Programming paradigm. It combines both the data and the functions that require access to particular data, and form an entity (called as class).

Solving the Shortcomings of Structured Programming

Solving Shortcoming 1

Object Oriented Programming discourages the direct access to the data and restricts to call a function which operates on the data. The function must be accessed if the data access is required. Now the function takes care of the legal and illegal requests and only operates on the data if the access is legal.

Solving Shortcoming 2

In Object Oriented Programming, classes are similar as primitive data types. There's only one difference that they are created by the programmer himself. This allows to create multiple variables (called as objects) from a class. This is identical to the fact that 'int' is a type and you can create multiple variables of it.

Similarly, you can make a class called Circle and then you can create multiple variables (objects) of it which all will behave identically.

Real World Programming

Object Oriented Programming is called Real World Programming as it models the real world more clearly as compared to Structured Programming. In real world, we see persons interact with each other similarly in OOP objects interact with each other to perform the task.

What Benefits it Offers?

- Maintainability
- Reusability
- Extensibility
- …

For What Type of Software OOP IS Suitable?

For almost all type of software, Object Oriented Programming is suitable.

For What Type of Software OOP Is NOT Suitable?

Whenever efficiency is a key consideration, usage of OOP is not advisable. As by OOP the overall efficiency of the software becomes less. Following are example of such type of software where OOP should not be used.

- Real-time Software.
- Operating System.
- Device Drivers.

Examples of some languages supporting OO paradigm

Following are some languages the support Object Oriented Programming Paradigm,

- C++
- Java
- C#
- Objective-C
- Small Talk
- …