**Object Oriented Programming**

**Vs**

**Procedural Programming**

by A. A. B. C. B. Adhikari

UOW number: 16280809

IIT number: 2015138

Module code: 4COSC008C

Module Name: Computer Science Practice

Email:Buddhi.2015138@iit.ac.lk

Tutorial Number: Tutorial 03

Date: 11/11/16

**Introduction**

I will discuss about the differences of Object Oriented Programming and Procedural Programming Paradigm. Moreover, I will discuss about each programming paradigm separately and I will give a brief explanation for each one of them.

There are several programming paradigms but two of the most important methods are Object Oriented Programming and Procedural Programming Paradigm. Consequently, this report will compare Object Oriented Programming and Procedural Programming Paradigm.

**Object Oriented Programming (OOP)**

Object Oriented Programming or OOP is a programming paradigm that uses the concept of classes and objects to construct models based on the real world surrounding. An **object** is a constituent of a program that recognizes how to execute certain actions and how to interrelate with other elements of the program (study.com, 2003). Objects are the foundation of object-oriented programming. An object-oriented program uses a set of objects, which will communicate by sending and receiving messages to request services or information. A **class** is a collection of objects with similar properties and behaviours (aka methods). A **method** (behaviours) in OO (Object-Oriented) language is like a procedure in procedural language. Finally, an **object** or a collection of objects (**class**) attempts to complete its goals (goals such as displaying ‘hello world’ on to the screen) by communicating by swapping messages. In fact, displaying ‘Hello World’ is a **method**.

Some examples for Object-Oriented Programming languages include Java, C#.NET, C++, Python and Perl.

**Procedural Programming**

Procedural Programming languages follow a sequence of instructions and conveys it to the computer. Procedural programming depends on procedures. As procedural programming language follows a method of solving problems from the top of the code to the bottom of the code, if a change is required to the program, the developer has to change every line of code that links to the main or the original code.

If the user wants to code a program, they would have to follow a sequence of instructions and thereby enter the instructions. In addition, we can say that when a problem is need to be fix using procedural programming, the developer will start with the problem (procedure) and then he logically fragment the problem down into sub problems (Sub-Procedures). Subsequently, this process will continue until a sub-procedure is simple enough to be solved by itself. Examples for procedural programming languages include C, COBOL, FORTRAN and VB.

**Comparing OOP And Procedural Programming**

When we consider, what are the differences between Object-Oriented Programming and Procedural Programming it is obvious that OOP is based on objects and classes while Procedural Programming is based on procedures. Using objects in OOP rather than procedures as in procedural programming allow the developers to reuse a single code anywhere as needed. Thus, allowing coding methods that are more complicated with ease and using less code.

When we consider about the security of the data when using either of the programming paradigms, OOP provides more security as it has a more improved data concealing mechanism rather than procedural programming languages.

Procedural programming uses global data for sharing data within functions therefore data can be accessed from function to function without any access limits. However, OOP does not allow global data but instead the developer has the ability to set the functions to private or public so developers can control the access rights for data.

In procedural programming, it is quite difficult to add new data or functions to the program but OOP offers an easy approach to add new data and functions. Additionally, in procedural programming data cannot be moved liberally from function to function but OOP allow objects to move and communicate with each other via member functions.

**Conclusion**

The purpose of this report was to compare Object Oriented Programming and Procedural Programming with adequate descriptions for each programming paradigm.

Firstly, the report looked into what is OOP and the report defined what OOP is. Moreover, the report briefly explained what objects, classes and methods are and how they cooperate with each other to make OOP work. Furthermore, we considered few examples for OOP.

Thereafter, the report show what is Procedural Programming and the report defined what Procedural Programming is. Moreover, the report briefly explained what procedures are, how procedures are used and how procedures work. Furthermore, we considered few examples for Procedural Programming.

Finally, we differentiated OOP and Procedural Programming Paradigms and we discussed few main differences between them. Which concluded us that OOP is more easy to use, more secure and efficient than procedural programming paradigm [1].

**References**

Dhanvani, P. (2012) *Difference between Procedure Oriented Programming and Object Oriented Programming | Procedure Oriented Programming vs. Object Oriented Programming*. Available at: http://freefeast.info/general-it-articles/difference-between-procedure-oriented-programming-and-object-oriented-programming-procedure-oriented-programming-vs-object-oriented-programming/ (Accessed: 11 November 2016).

Eliason, K. (2013) *Difference Between Object-oriented Programming and Procedural Programming Languages (Part 1)*. Available at: https://neonbrand.com/website-design/procedural-programming-vs-object-oriented-programming-a-review/ (Accessed: 11 November 2016).

Rouse, M. (2013) *What is object-oriented programming (OOP)? - definition from WhatIs.Com*. Available at: http://searchsoa.techtarget.com/definition/object-oriented-programming (Accessed: 11 November 2016).

Russell (no date) *Object-oriented programming vs procedural programming*. Available at: http://www.ctp.bilkent.edu.tr/~russell/java/LectureNotes/1\_OOConcepts.htm (Accessed: 11 November 2016).

Study.com (2003) *Object oriented programming vs. Procedural programming - video & lesson transcript*. Available at: http://study.com/academy/lesson/object-oriented-programming-vs-procedural-programming.html (Accessed: 9 November 2016).

Zahid, H.B. (2015) *Procedural vs. Object oriented programming*. Available at: http://www.slideshare.net/HarisBinZahid/procedural-vs-object-oriented-programming (Accessed: 11 November 2016).