**THE UNIVERSITY OF DODOMA**

**COLLEGE OF INFORMATICS AND VITUAL EDUCATION**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**INDIVIDUAL ASSIGNMENT**

**STUDENT NAME:**ANUARI IDDI ISSA

**COURSE**: BSc-SE

**REG NO**: T/UDOM/2020/00345

**COURSE NAME**: OBJECT- ORIENTED SYSTEMS DESIGN

**COURSE CODE**: CP 223

**Instructor:** Mr. Justine Woiso

**1: THE HISTORY OF OBJECT ORIENTED DESIGN IN RELATION TO HISTORY OF PROGRAMMING FROM 1990S TO DATE.**

Object oriented programming developed as the dominant programming methodology in the early and mid 1990s when programming languages supporting the techniques became widely available. These included Visual FoxPro 3.0, C++ and Delphi. Its dominance was further enhanced by the rising popularity techniques. An example of a closely related dynamic GUI library and OOP language can be found in cocoa frameworks on Mac OS X, written in objective-C an object-oriented, dynamic messaging extension to C based on Smalltalk. OOP toolkit also enhanced the popularity of event driven programming ( although this concept is not limited to OOP).

At ETH Zurich, Nicklaus Wirth and his colleagues had also been investigating such topics as data abstraction and modular programming ( although this had been in common use in the 1960s or earlier) Modula-2 (1978) included both, and their succeeding design, Oberon, included a distinctive approach to object orientation, classes and such.

Object- oriented features have been added to many previously existing languages including **Ada, basic, Fortran, pascal and cobol.** Adding these features to languages that were not initially designed for them often led to problems with compatibility and maintainability of code.

To date, number of languages have emerged that are primarily object oriented but that are also compatible with procedural methodology. Two such languages are python and ruby. Probably the most commercially- important recent object-oriented languages are Visual Basic.NET (VB.NET) and C#, both designed for Microsoft's .NET platform and Java which are developed by sun Micro systems. Each of these two frameworks shows in its own way, the benefit of using OOP by creating an abstraction from implementation. VB.NET and C# support cross-language inheritance, allowing classes defined in one language to subclass classes defined in the other language. Developers usually compile java to byte code , allowing java to run on any operating system for which a java virtual machine is available.

**REFERENCE:**

* Johnson R.E and Foote B. design reusable classes
* Journal of object-oriented programming, 1(2), pp. 22-35, June/July 1988. [2] Micallef J. Encapsulation, Re usability and Extensibility in object-oriented programming languages,

**2:Differences between object- oriented design and object-oriented programming.**

**OOD** (Design) is where you break up the problem (or problem space) into objects, or perhaps it could be thought of as building up your solution conceptually into objects.

**OOP** (Programming) is the act of doing the programming. This is where you need to know the programming language. Sort of like mapping the design onto software concepts.