PROGRAMMING LANGUAGE SURVEY

PROGRAMMING LANGUAGE SURVEY - SIMULA

Presented by Avi Nair
CB.EN.U4CYS21002
TIFAC-CORE in Cyber Security
Amrita Vishwa Vidyapeetham, Coimbatore Campus

Feb 24, 2023

AMRITA
VISHWA VIDYAPEETHAM

Outline

Introduction

Simula is a programming language that was developed in the 1960s by Ole-Johan Dahl and Kristen Nygaard at the Norwegian Computing Center in Oslo, Norway. Simula is often considered the first object-oriented programming language, as it introduced many of the concepts that are now fundamental to object-oriented programming.

Simula was designed to help programmers model complex systems, such as simulations of real-world systems. It was also designed to support the development of large, complex software systems, with features like classes, inheritance, and encapsulation. Simula's object-oriented programming concepts have been widely adopted by other programming languages, including C++, Java, and Python.

Simula was also influential in the development of the programming language Smalltalk, which is another important object-oriented programming language. While Simula is not widely used today, its impact on the development of modern programming languages is still felt.

Why was SIMULA created?

The purpose behind the creation of Simula was to provide a programming language that could support the development of simulations of real-world systems. Ole-Johan Dahl and Kristen Nygaard, the creators of Simula, were interested in developing a programming language that could model complex systems, such as those found in science, engineering, and economics.

Simula was designed to allow programmers to define and manipulate abstract data types, which could be used to model real-world objects and systems. This was achieved through the introduction of key concepts of object-oriented programming, such as classes, inheritance, and encapsulation. By using Simula, programmers could more easily develop simulations of real-world systems, and test out various scenarios in a virtual environment before implementing them in the real world.

In addition to its use in simulations, Simula was also designed to support the development of large, complex software systems. The object-oriented programming concepts introduced in Simula have since been widely adopted by other programming languages, making Simula an important precursor to many modern programming languages.This slide shows the example for Blocks.

Sample Code

Here is a simple Simula program that demonstrates some of its unique features compared to other programming languages of its time:

CODE

```
Class Person;
. Integer age;
. Text name;

. Procedure printInfo;
. OutText(name); OutText(": "); OutInt(age,0); OutImage;
. End printInfo;

End Person;

Begin
. Person p;
. p.name := "Alice";
. p.age := 25;
. p.printInfo;
End.
```

Code - Explanation

This Simula program defines a class called Person, which has two attributes - age and name. It also defines a method called printInfo, which prints out the name and age of the person.

What makes this program unique compared to other programming languages of its time is its use of object-oriented programming concepts, such as classes and methods. The Person class is defined as an abstract data type, which can be used to model real-world objects and systems.

Simula also introduced the concept of inheritance, which allows new classes to be derived from existing ones. This feature is not used in the above program, but it allows for greater flexibility and code reusability in more complex programs.

Furthermore, Simula was one of the first programming languages to support the creation of graphical user interfaces, making it ideal for developing simulations and other visual applications.

Overall, this simple Simula program demonstrates some of the unique features that s[illegible] apart from other programming languages of its time, and paved the way for the development of modern object-oriented programming languages.

Current Usage

Simula is not widely used today as a programming language, but its impact on the development of modern programming languages is still felt. The key features of Simula, such as classes, inheritance, and encapsulation, have been widely adopted by other programming languages, including C++, Java, and Python.

Simula was primarily developed as a language for modeling complex systems and simulations, and its use in these areas has decreased over time due to the development of more specialized tools and languages. However, Simula is still used in some niche areas, such as in the development of some specific types of simulations.

In addition, Simula's influence on modern programming languages has been significant. Many of the concepts introduced in Simula have become standard in modern object-oriented programming languages, and the development of Simula played an important role in the evolution of computer programming as a whole.

Overall, while Simula is not widely used as a programming language today, its contributions to the field of computer programming and its impact on the development of modern programming languages cannot be overstated.