

Programming Applications and Programming Languages M30235

TB1&2

University of Portsmouth
BSc Computer Science
2nd Year

Contents

1	Lecture - Introduction	2
2	Lecture - Introduction to Programming Languages	3

Lecture - Introduction

14:00 22/01/24 Jiacheng Tan

• No content from TB1 is assessed as part of TB2

Lecture - Introduction to Programming Languages

14:00 22/01/24 Jiacheng Tan

Since there are many different types of application, there are also many types of programming language. The main programming domains are as follows

- Scientific (e.g. ForTran)
- Business (e.g. COBOL)
- AI (e.g. LISP)
- Systems Programming (e.g. C, C++)
- Web Software (e.g. HTML, JavaScript)

Language Categories

There are several ways to categorise programming languages, such as by uses, paradigms, abstraction level, etc

Machine Languages

- Machine languages directly run on the hardware, using the instruction set of the processor
- Machine code is usually written in hexadecimal as this is a more efficient way of displaying the binary which represent the instructions
- It is very hard for programmers to directly write machine code, as it is not easy to remember instructions and it lacks features such as jump targets, subroutines, etc

Assembly Languages

- A slight abstraction over machine languages
- Each instruction is replaced with an alphanumeric symbol which is easier for programmers to remember and understand
- They also include features such as subroutines, jump targets, etc which make it much easier to create complex programs

System Programming Languages

- More abstracted from machine languages, but you are still concerned with low-level functions such as memory management
- Used to create operating systems, and for embedded applications where low system requirements do not allow the use of high-level languages

High Level Languages

- Languages that are machine-independent (are not written directly in machine code, and are therefore portable between CPU architectures)
- Need to be compiled or otherwise translated from text to machine code before they can be run

Scripting Languages

- Used to create programs which perform a single, simple task
- Thses are used for system administration
- Usually interpreted languages
- More akin to pseudocode than other programming languages

Domain-Specific Languages

- Some languages are designed to perform a specific task much more efficiently
- The specific purpose could be just about anything, but are specific to that task and either cannot be used otherwise or are not well suited for it

Programming Paradigms

There are several different paradigms which are used in programming

- Procedural
 - Most programming languages are procedural
 - A program is made up of one or more routines which are run in a specific order
- Functional
 - Applies mathematical functions to inputs to get a result
 - Useful for data processing applications such as data analysis or big data
- Logical

There are also two major types of programming languages, which are designed for different purposes

- · Imperative Languages
 - Programs are defined as a sequence of commands for the computer to perform
 - Like a recipe for exactly how to get the desired output
- Declarative Languages
 - Programs describe the desired results without actually specifiying how the program should complete the task
 - Functional and logical programming languages are examples of this