Unit 4 Notes

Programming language

A **programming language** is a system of notation for writing computer programs.

Programming languages are described in terms of their syntax (form) and semantics (meaning), usually defined by a formal language. Languages usually provide features such as a type system, variables and mechanisms for error handling. An implementation of a programming language in the form of a compiler or interpreter allows programs to be executed, either directly or by producing an executable.

Programming tools

Programming tools, often referred to as tools in the software industry, are sets of software applications used to write, test, debug and maintain source code. They provide foundations for compiling, interpreting and editing code in various programming languages.

Compiler	**An application that translates high-level programming language into machine code.**
Text editor	Used to write and edit code in various programming languages.
Debugger	Detects and removes errors or bugs from source code.
Assembler	Converts assembly language into machine code.

Exploring Different Types of Programming Tools

When coding, you wouldn't be limited to only one type of programming tool. Various tools are available, each designed to assist a different part of your coding journey.

- IDE (Integrated Development Environment)
- Code Editor
- Library
- API (Application Programming Interface)

The Role and Function of Programming Tools in Computer Science

Programming tools help computer scientists be more productive and efficient. Those tools not only help write and design programs but also help find and fix mistakes, optimise code, and run tests to ensure functionality. Understanding the role and function of these tools is crucial in learning how to code and develop software.

Software development tools

There are many different types of software development tools and programming languages available, and the most suitable one for a particular project will depend on the specific requirements and goals of the project. Here are some common types of software development tools and languages:

1. Integrated Development Environments (IDEs): These are software applications that provide a comprehensive set of tools for software development, including a text editor, compiler, debugger, and other tools. Examples include Visual Studio and Eclipse.
2. Source Control Management (SCM) tools: These tools allow developers to manage and track changes to the source code of a software project. Examples include Git and Subversion.
3. Build Automation Tools: These tools automate the process of building and testing software, including tasks such as compiling code and running tests. Examples include Make and Ant.
4. Debugging Tools: These tools help developers find and fix errors in their code. Examples include GDB and LLDB.
5. Programming Languages: These are the languages used to write software programs. Some popular programming languages include C, C++, Java, Python, and JavaScript.

References

- Software Engineering: A Practitioner's Approach | 9th Edition By Roger S. Pressman, Bruce R. Maxim
- Software Engineering 9th Edition by Ian Sommerville.
- https://www.geeksforgeeks.org/software-engineering-introduction-to-software-engineering/
- https://www.coursera.org/learn/introduction-to-software-engineering
- https://en.wikipedia.org/wiki/Software_engineering