

LATIHAN 5.1

1. *State the definition of program and programming language.*

PROGRAM

A computer program is a series of organised instructions that directs a computer to perform tasks. Without programs, computers are useless.

PROGRAMMING LANGUAGE

A programming language is a set of words, symbols and codes that enables humans to communicate with computers. It is a language used for writing computer programs, that direct a computer to perform computation and to organise the flow of control between mechanical devices.

2. *Identify the generations of low-level and high-level programming languages with examples.*

A low-level programming language is a programming language that provides little or no abstraction from computer's microprocessor.

FIRST GENERATION OF PROGRAMMING LANGUAGE

The first generation of programming language, or 1GL, is machine language. Machine language is a set of instructions and data that a computer's central processing unit can execute directly.

SECOND GENERATION PROGRAMMING LANGUAGE

The second generation programming language, or 2GL, is assembly language. Assembly language is the human-readable notation for the machine language used to control specific computer operations

A high-level programming language is a programming language that is more abstract, easier to use, and more portable across platforms.

THIRD GENERATION PROGRAMMING LANGUAGE

The third generation of programming language, 3GL, or procedural language uses a series of English-like words, that are closer to human language, to write instructions.

FOURTH GENERATION PROGRAMMING LANGUAGE

The fourth generation programming language or non-procedural language, often abbreviated as 4GL, enables users to access data in a database.

FIFTH GENERATION PROGRAMMING LANGUAGE

The fifth generation programming language or visual programming language, is also known as natural language. Provides a visual or graphical interface, called a visual programming environment, for creating source codes

3. *Define structured approach in programming.*

Structured programming often uses a top-down design model where developers map out the overall program structure into separate subsections from top to bottom.

In the top-down design model, programs are drawn as rectangles. A top-down design means that the whole program is broken down into smaller sections that are known as modules. A program may have a module or several modules

4. *Define object-oriented approach in programming.*

Object-Oriented Approach

The object-oriented approach refers to a special type of programming approach that combines data with functions to create objects. In an object-oriented program, the objects have relationships with one another. One of the earliest OOP languages is Smalltalk. Java, Visual Basic and C++ are examples of popular OOP languages

5. *Differentiate between structured approach and object-oriented approach in programming.*

STRUCTURED APPROACH	OBJECT-ORIENTED APPROACH
Structured programming often uses a top-down design model	The object-oriented programming approach uses objects.
The programmer divides programming problem into module like function	The programmer packages the data and function into single unit and object

6. *Describe the translation method of programming using assembler, interpreter and compiler.*

Assembler

An assembler is a computer program for translating assembly language — essentially, a mnemonic representation of machine language — into machine language.

For example in intel 80836, the assembly language for the 'no operation' command is NOP and its machine code representation is 10010000. Examples of assemblers are MACRO-80 Assembler and Microsoft MASM.

Interpreter

Interpreter is used to interpret and execute program directly from its source without compiling it first. The source code of an interpreted language is interpreted and executed in real time when the user executes it.

The interpreter will read each code, convert it to machine code and execute it line by line until the end of the program. Examples of interpreter-based languages are BASIC, Logo and Smalltalk

Compiler

The source code (in text format) will be converted into machine code which is a file consisting of binary machine code that can be executed on a computer. If the compiler encounters any errors, it records them in the program-listing file.

7. Differentiate between constants and variables.

	CONSTANTS	VARIABLES
CHARACTERISTICS	Value is not changeable during the course of the program.	Value can be changed anytime during the course of the program
USAGE	Use constant when you want to declare something that won't be changed midway in your program execution	Use variable to store data that may or will change during the running of the program

8. Differentiate between the data types: Boolean, integer, double, string and date.

A data type is a type of data. Of course, that is rather circular definition, and also not very helpful. Therefore, a better definition of a data type is a data storage format that can contain a specific type or range of values

INTEGER

An integer is a whole number (not a fraction) that can be positive, negative, or zero. Therefore, the numbers 10, 0, -25, and 5,148 are all integers. Unlike floating point numbers, integers cannot have decimal places

BOOLEAN

Boolean, or boolean logic, is a subset of algebra used for creating true/false statements. Boolean expressions use the operators AND, OR, XOR, and NOT to compare values and return a true or false result.

DOUBLE

As the name implies, floating point numbers are numbers that contain floating decimal points. For example, the numbers 5.5, 0.001, and -2,345.6789 are floating point numbers. Numbers that do not have decimal places are called integers

STRING

A string is a data type used in programming, such as an integer and floating point unit, but is used to represent text rather than numbers. It is comprised of a set of characters that can also contain spaces and numbers. For example, the word "hamburger" and the phrase "I ate 3 hamburgers" are both strings. Even "12345" could be considered a string, if specified correctly. Typically, programmers must enclose strings in quotation marks for the data to be recognized as a string and not a number or variable name.

DATA TYPES	EXAMPLES
Integer	18, 79, 21
Double	41.5, 31.4
String	Kok Keong, Amira, Eden, Alex
Boolean	TRUE,FALSE

9. *Differentiate between mathematical and logical (Boolean) operators.*

Mathematical Operator

In computer programming, four basic ways are used to manipulate numbers: adding, subtracting, multiplying, and dividing. To perform a mathematical operation with two numbers (or two numbers represented by variables), use these symbols in your computer program

Logical Operator

The logical operators are often used to help create a test expression that controls program flow. This type of expression is also known as a Boolean expression because they create a Boolean answer or value when evaluated. The answers to Boolean expressions within the C++ programming language are a value of either 1 for true or 0 for false. There are three common logical operators that give a Boolean value by manipulating other Boolean operand(s). Operator symbols and/or names vary with different programming languages.

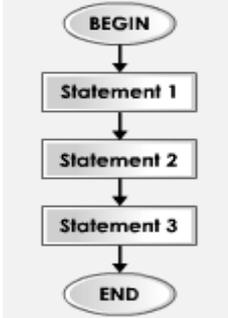
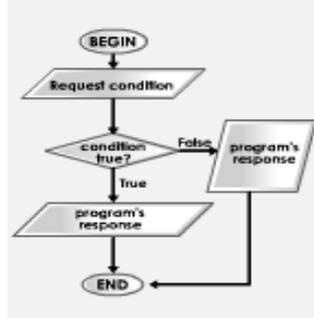
10. *Differentiate between sequence control structure and selection control structure.*

Sequence Control Structure

Sequence is the default control structure; instructions are executed one after another. They might, for example, carry out a series of arithmetic operations, assigning results to variables, to find the roots of a quadratic equation $ax^2 + bx + c = 0$. The conditional IF-THEN or IF-THEN-ELSE control structure allows a program to follow alternative paths of execution.

Selection Control Structure

The selection control structure allows one set of statements to be executed if a condition is true and another set of actions to be executed if a condition is false.

ASPECTS	SEQUENCE CONTROL	SELECTION CONTROL
Usage	<ul style="list-style-type: none"> • Use when want to execute code line by line. • Does not use the decision symbol. 	<ul style="list-style-type: none"> • Use when want to implement decision making process in the program. • Use the decision symbol.
Execution Flow	<ul style="list-style-type: none"> • Execute statement one by one in linear @ consecutive order. 	<ul style="list-style-type: none"> • Execute different statement for different conditions.
Flow Chart		

LATIHAN 5.2

1. Describe the five main phases in program development:

Problem Analysis Phase	<ul style="list-style-type: none"> • A programmer reviews and defines the problems. • Identifies the data input, process and output for the program
Program Design Phase	<ul style="list-style-type: none"> • the programmer will design a flow chart
Coding Phase	<ul style="list-style-type: none"> • Once the flow chart is confirmed, the programmer will perform coding • Coding is the process of writing the solution using the computer programming language.
Testing And Debugging Phase	<ul style="list-style-type: none"> • Program will be tested by the users • If there are any errors, the programmer will do a debugging of the program.
Documentation Phase	<ul style="list-style-type: none"> • the programmer will complete the documentation for the program; this includes the user manual, a clear layout of the input and output records and a program listing.

LATIHAN 5.3

1. *Find out the latest programming languages*

Fifth Generation Language

Fifth generation programming language (5GL) is an advance programming language which concentrates on solving problems using constraints given to the program. In fifth generation language, the programmer just need to define the problem to be solve and the program will automatically code the program based on the problem definition. Fifth generation languages are designed to make the computer solve the problem for you. These languages are mostly used in artificial intelligence research. Examples of fifth generation languages include Prolog and Mercury.

Natural Language

Natural Language programming aims to use natural language such as English to write a program. Instead of using a specific programming language syntax, natural language programming will use normal English as the input to program software. Such a technique would mean less technical programming knowledge is required to write a program. The programmer needs to define the program using normal language.

OpenGL (Graphic Library)

OpenGL is a standard specification to describe the standard Application Programming Interface (API) for 3D/2D computer graphic applications. It's specification describes a set of functions and the exact behaviours that the 3D/2D application must perform. OpenGL was developed by Silicon Graphics. OpenGL is widely used in virtual reality, scientific visualisation, flight simulation and video game development