

## **GE8151 PROBLEM SOLVING AND PYTHON**

### **PROGRAMMING**

#### **UNIT-I**

#### **ALGORITHMIC PROBLEM SOLVING**

#### **PART-A**

**1. What is an algorithm?**

Algorithm is a step-by-step procedure/description for solving a task or a problem. The steps must be ordered, unambiguous and finite in number. Algorithm is an ordered sequence of finite, well defined unambiguous instructions for completing a task it is an English-like representation of the logic which is used to solve the problem.

**2. Write an algorithm to accept two numbers compute the sum and print the result.**

**Algorithm:**

**Step 1:** Start

**Step 2:** Read the two numbers A, B.

**Step 3:** Compute the sum,  $\text{sum} = A + B$

**Step 4:** Print the value of sum

**Step 5:** Stop

**3. Write an algorithm to find the minimum number in a given list of numbers.**

**Algorithm:**

**Step 1:** Start

**Step 2:** Read the limit of list as 'n'

**Step 3:** Read the elements of list A

**Step 4:** Assign  $\text{min} = A[0]$

**Step 5:** for  $i = 1$  to  $n$  repeat step 6 and step 7

**Step 6:** Check if  $A[i] < \text{min}$

**Step 7:** Then assign  $\text{min} = A[i]$

**Step 8:** Print the value of min

**Step 9:** Stop

**4. Distinguish between algorithm and program**

Algorithm	Program
Algorithm is finite.	Program need not be finite.
It is a step-by-step procedure to solve a problem.	It is an exact code to solve a problem.
It is written using natural language or algorithmic language.	It is written using a specific programming language.

**5. List the building blocks of algorithm.**

The building blocks of an algorithm are

- Statements
- State
- Control flow
  - Sequence
  - Selection or Conditional
  - Repetition or Iteration
- Functions

**6. Define statement. List its types.**

Each and every line in an algorithm is called statement. A statement is a segment of code that represents a command.

Two types:

- Simple Statement
- Compound Statement

**7. How does flow of control work?**

Control flow (or flow of control) is the order in which individual statements, instructions or function calls of a program are executed or evaluated. A control flow statement is a statement which execution results in a choice being made as to which of two or more path to follow.

**8. Define control flow statement with an eg.**

A Control flow statement is a statement which execution results in a choice being made as to which of two or more paths to follow.

Example: Algorithm to find biggest of two numbers

**Step 1:** Start

**Step 2:** Read the two numbers A, B

**Step 3:** Check if(A>B) then //selection control flow statement

**Step 4:** Print A is big

**Step 5:** Else print B is big

**Step 6:** Stop

**9. What is a function?**

A function is a set of related instructions that are used to perform a specified task which repeatedly occurs in the main program. Functions usually “take in” data, process it, and “return” a result.

**10. Define a pseudocode?**

Pseudo means limitation or false and code refers to the instructions written in any programming language. It is also called Program Design Language (PDL). It is somewhat halfway in between English and a Programming Language.

**11. Give the rules of writing Pseudocode.**

- Write one statement per line.
- Keywords must be capitalized
- Steps must be understandable
- It must be concise
- Statements must be language independent.

**12. Write the pseudocode to calculate the sum and product of two numbers and display it.**

```
BEGIN
    READ TWO NUMBERS num1, num2
    COMPUTE sum=num1+num2
    COMPUTE product=num1*num2
    PRINT the value of sum and product
END
```






**13. Give the difference between flowchart and pseudocode (or) Distinguish between pseudocode and flowchart.**

Flowchart	Pseudocode
A flowchart is a diagrammatic representation of an algorithm.	Pseudo means imitation or false and code refers to the instructions written in any programming language.
It follows a standard format.	Unlike a flowchart, it uses a written format.
It uses different shapes of boxes to represent the processes involved in the problem	It can be written in ordinary English, and with some keywords.

**14. Define a flowchart.**

A flowchart is a diagrammatic representation of an algorithm. In which the steps are represented in the form of different shapes of boxes and logical flow is indicated by interconnecting arrows. The purpose of drawing the flowchart is to help the programmer in understanding the logic the program.

**15. List the symbols used in drawing the flowchart?**

Symbols	Name
	Terminal
	Input/Output
	Processing
	Decision
	Connector

**16. List the categories of Programming Languages. (or) Point out any 5-programming language.**

Programming Language are divided into the following categories:

- Interpreted Programming Languages
- Functional Programming Languages
- Compiled Programming Languages
- Procedural Programming Languages
- Scripting Programming Languages
- Markup Programming Languages
- Logic-Based Programming Languages
- Concurrent Programming Languages
- Object-Oriented Programming Languages

**17. Compare machine language, assembly language and high-level language.**

Machine Language	Assembly Language	High-Level Language
The language of 0s and 1s is called as machine language.	It is low level programming language in which the sequence of 0s and 1s are	High level languages are English like statements and programs.

	replaced by mnemonic (ni-monic) codes.	
The machine language is system independent because there are different set of binary instruction for different types of computer systems.	Typical instruction for addition and subtraction are add and sub.	Written in these languages are needed to be translated into machine language before to their execution using a system software compiler.

**18. List out the simple strategies to develop an algorithm (or) Discover the steps of simple strategies for developing algorithms.**

Algorithm development process consists of five major steps.

**Step 1:** Obtain a description of the problem

**Step 2:** Analyze the problem

**Step 3:** Develop a high-level algorithm

**Step 4:** Refine the algorithm by adding more details

**Step 5:** Review the algorithm

**19. Give the difference between recursion and iteration**

Recursion	Iteration/Repetition
Function calls itself again and again until the base condition is reached.	Executing the processes until the condition fails.
In recursive function only base condition (terminate condition) is specified.	Iterative approach involves four steps: initialization, condition, execution and updation.
Recursion keeps your code short and simple.	Iterative approach makes your code longer.
Recursion is slower than iteration due to overhead of maintaining stack.	Iterative is faster.
Recursion takes more memory.	Iteration takes less memory.