TOPIC 1 – INTRODUCTION TO PROBLEM SOLVING

LEARNING OUTCOMES

By the end of this topics, you will be able to:

1. Explain the essence of computational problem solving
2. Explain what a computer algorithm is
3. Describe about program development life cycle

1.0 ALGORITHMIC PROBLEM SOLVING

Algorithms, building blocks of algorithms (statements, state, control flow, functions), notation (pseudo code, flow chart, programming language), algorithmic problem solving, simple strategies for developing algorithms (iteration, recursion). Illustrative problems: find minimum in a list, insert a card in a list of sorted cards, guess an integer number in a range, Towers of Hanoi.

1.1 Algorithms

What is algorithm?

An algorithm is a collection of well-defined, unambiguous and effectively computable instructions ,if execute it will return the proper output.

well-defined- The instructions given in an algorithm should be simple and defined well.

Unambiguous- The instructions should be clear,there should not be ambiguity .

Effectively computable- The instructions should be written step by step ,which helps computer to understand the control flow .

We often use algorithm in our day-to-day life,but when and how?

Our cooking receipe

Our daily routine as a student

When we buy something

When we go outing

Our class routine

How it helps?

How an algorithm should be?

It should be in simple English,what a programmer wants to say. It has a **start,a middle and an end**. Probably an algorithm should have,

Start

1. In the middle it should have set of tasks that computer wants to do and it should be in simple English and clear.
2. To avoid ambiguous should give no for each step.

Stop

Start
Create a variable to get the user's email address clear the variable,incase its not empty.
Ask the user for an email address.
Store the response in the variable.
Check the stored response to see if it is a valid email address Not valid?Go back
Stop

Why this Happened?

This was happened because the instructions given in an algorithm does not have numbering for each step. To avoid this ambiguity ,we should number each step while writing an algorithm.
So lets rewrite the algorithm:

Step1: Start
Step2: Create a variable to get the user's email address
Step3: Clear the variable,incase its not empty.
Step4: Ask the user for an email address.

Step5: Store the response in the variable.
Step6: Check the stored response to see if it is a valid email address
Step7: Not valid?Go back
Step8: Stop

1.2 PROGRAM DEVELOPMENT LIFE CYCLE

When we want to develop a program using any programming language, we follow a sequence of steps. These steps are called phases in program development. The program development life cycle is a set of steps or phases that are used to develop a program in any programming language.

Generally, the program development life cycle contains 6 phases, they are as follows:

- Problem Definition
- Problem Analysis
- Algorithm Development
- Coding & Documentation
- Testing & Debugging
- Maintenance

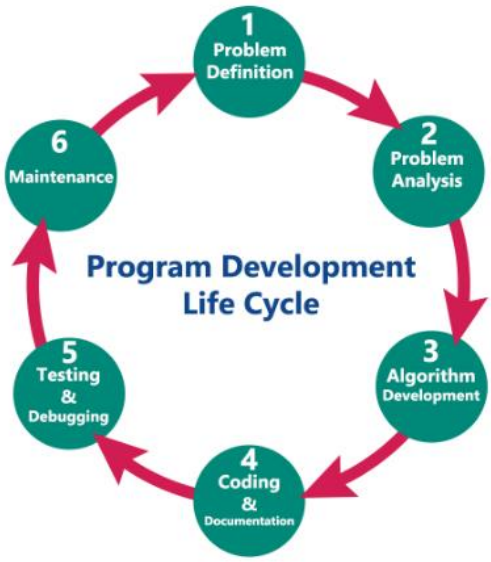

Figure 1

1. Problem Definition

In this phase, we define the problem statement and we decide the boundaries of the problem. In this phase we need to understand the problem statement, what is our requirement, what should be the output of the problem solution. These are defined in this first phase of the program development life cycle.

2. Problem Analysis

In phase 2, we determine the requirements like variables, functions, etc. to solve the problem. That means we gather the required resources to solve the problem defined in the problem definition phase. We also determine the bounds of the solution.

3. Algorithm Development

During this phase, we develop a step by step procedure to solve the problem using the specification given in the previous phase. This phase is very important for program development. That means we write the solution in step by step statements.

4. Coding & Documentation

This phase uses a programming language to write or implement the actual programming instructions for the steps defined in the previous phase. In this phase, we construct the actual program. That means we write the program to solve the given problem using programming languages like C, C++, Java, etc.,

5. Testing & Debugging

During this phase, we check whether the code written in the previous step is solving the specified problem or not. That means we test the program whether it is solving the problem for various input data values or not. We also test whether it is providing the desired output or not.

6. Maintenance

During this phase, the program is actively used by the users. If any enhancements found in this phase, all the phases are to be repeated to make the enhancements. That means in this phase, the solution (program) is used by the end-user. If the user encounters any problem or wants any

enhancement, then we need to repeat all the phases from the starting, so that the encountered problem is solved or enhancement is added.