Lecture

NOTE: FOR FURTHER DETAILS AND MORE COMPREHENSIVE STUDY, PLEASE SEE RECOMMENDED BOOKS OR INTERNET.

IMPORTANT:
BEFORE GOING DEEP INTO DATA STRUCTURE, YOU SHOULD HAVE A GOOD KNOWLEDGE OF PROGRAMMING EITHER IN C OR IN JAVA.

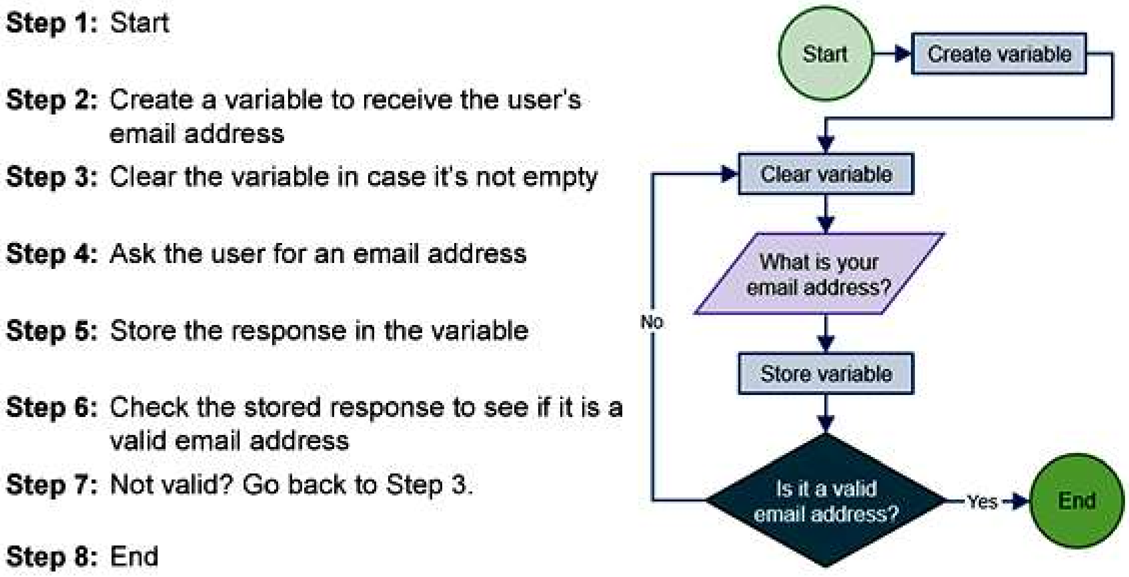
What is Algorithm?

An algorithm is a finite set of instructions or logic, written in order, to accomplish a certain predefined task. In other words, it's a representation of an idea. Algorithm is not the complete code or program; it is just the core logic (solution) of a problem, which is expressed either in simple English language or a formal high level description of pseudo code. Algorithm is not bound to the written form but can also be a pictorial representation of a flow chart.

Characteristics of Algorithm

- It must be CORRECT
- Composed of CONCRETE STEPS
- NO AMBIGUITY which step will be perform next
- FINITE number of Steps
- It must TERMINATE
- ZERO or MORE inputs
- ONE or MORE outputs
- EFFICIENT and FLEXIBLE

Example: Take an email address from user and check whether it is valid or not.



Step 1: Start

Step 2: Create a variable to receive the user's email address

Step 3: Clear the variable in case it's not empty

Step 4: Ask the user for an email address

Step 5: Store the response in the variable

Step 6: Check the stored response to see if it is a valid email address

Step 7: Not valid? Go back to Step 3.

Step 8: End

Example of Algorithm with Flowchart

Complexity

An algorithm is said to be efficient and fast, if it takes less time to execute and consumes less memory space. The performance of an algorithm is measured on the basis of following properties:

- *Space Complexity*
- *Time Complexity*

Space Complexity

Its the amount of memory space required by the algorithm, during the course of its execution. Space complexity must be taken seriously for multi-user systems and in situations where limited memory is available.
An algorithm generally requires space for following components:

- **Instruction Space:** Its the space required to store the executable version of the program. This space is fixed, but varies depending upon the number of lines of code in the program.
- **Data Space:** Its the space required to store all the constants and variables value.
- **Environment Space:** Its the space required to store the environment information needed to resume the suspended function.

Time Complexity

Time Complexity is a way to represent the amount of time needed by the program to execute. This complexity is very important type of complexity. Almost every algorithm is benchmarked based on its time complexity. Most of the time only the time complexity of an algorithm is used to compare with other algorithms that perform the same function to evaluate which one is better.

Pseudo Code

“Pseudo” means imitation or False, thus resulting Pseudo Code as False Code. Pseudo Code is a simple way of writing programming code in English so it could be implemented in any Programming Language. Pseudo Code is not an actual programming language. It uses short phrases to write code for programs before you actually create it in a specific language. Once you know what the program is about and how it will function, then you can use Pseudo Code to create statements to achieve the required results for your program.
Example: Take an email address from user and check whether it is valid or not.

```
1.  Start
2.  Initialize String email;
3.  email = “ ”;
4.  Input email;
5. IF (email != VALID)
   THEN
       Go to 4.
6. End/Terminate
```

NOTE: Definition of pseudo Code varies from Author to Author and Book to Book. Some authors consider a pseudo code as algorithm, while some say both are different.

Relationship among Data, Data Types, Data Structure, and Algorithm

The relationship among Data, Data Types, Data Structure, and Algorithm is as follows:

Data and Data Types

Selecting best and efficient data type for the data you want to store.

Data Types and Data Structure

Selecting which Data Type can be used to efficiently represent the data Structure. E.g. Implementation of list in arrays is much more efficient then implementation in Linked List.

Data Structure and Algorithms

The total efficiency of algorithm is based on the selection and implementation of the data structure. A single program can take few second to a many days to complete based on the implementation of the program.

Data → Selection of Efficient Data Type → Efficient Data Structures → Efficient Algorithms