Analysis and Design of algorithms

Books

- **Fundamentals of Computer algorithms**

  Horowitz , Sahani and Rajasekaran

- **Introduction to Algorithms**

  Coremen , Leiserson

- **The Design and Analysis of Computer Algorithms**

  Aho, Hopcroft and Ullman

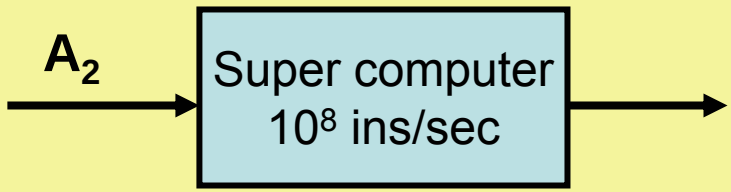
ALGORITHM

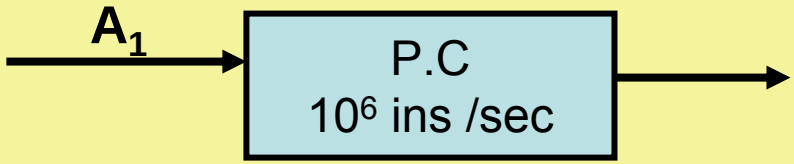
A finite set of instructions which if followed accomplish a particular task.

In addition every algorithm must satisfy following criteria:

1. Input: zero or more quantities externally supplied
2. Output: at least one quantity is produced
3. Definiteness: Each instruction must be clear and unambiguous.
4. Finiteness: In all cases algorithm must terminate after finite number of steps.
5. Effectiveness: each instruction must be sufficiently basic.

- Two algorithm on two systems
- Algorithm A_1 $50\ n \lg n$
- Algorithm A_2 $2\ n^2$





For $n = 10^6$

Time taken by Super Computer

$= 2.(10^6)^2 / 10^8 = 20{,}000$ sec

Time taken by P .C.

$= 50 \cdot 10^6 \lg 10^6 / 10^6 = 1{,}000$ sec

Thus by using a fast algorithm , the personal computer gives results

20 times faster than the result given by super computer using a slow algorithm.

Thus a good algorithm is like a sharp knife, it does exactly what it is supposed to do with a minimum amount of effort.

Complexity

Some questions to answer:

- How fast can we solve a problem?
- There may be many algorithms for a given problem. Which algorithm to use?
- What are the classical algorithm design techniques ?
- Are there problems inherently difficult to solve?

How do we express the complexity of algorithm?

Resources : Time and Space

Complexity lower bounds for problems.

Complexity classes P, NP etc.

Pseudocode

- Pseudocode is an English language like representation of the code required for an algorithm.
- It is partly English, partly structured code.
- The English part provides a relaxed syntax that is easy to read.
- The code part consists of an extended version of the basic algorithmic constructs-sequence, selection and iteration.

Sequence, selection, loop

- A sequence is a series of statements that do not alter the execution path within an algorithm.
- Statements such as assign and add are sequence statements.
- A call to another algorithm is also considered a sequence statement.
- Selection statements evaluate one or more alternatives. Paths are followed based on its result.

- The typical selection statement is the two way selection
- if (condition) action 1 else action 2.
- The part of the loop are identified by indentation.
- Loop iterates a block of code. It closely resembles the while loop. It is a pretest loop.

Example

Algorithm deviation

It finds deviations from average.

Pre: nothing

Post: numbers are read and deviation from average printed

```
1     i= 0
2     Loop(all data are read)
                  1       I = I + 1
                  2       read numbers into array[i]
                  3       sum = sum + number
3     Average = sum / I
4     Print (average)
5     J = 0
6     Loop (j < i)
                  1       j = j+ 1
                  2       dev = array[j] – average
                  3       print (array [ j] . Dev)
7     Return
8     End deviation
```

Asymptotic notations(O , Ω , Θ)