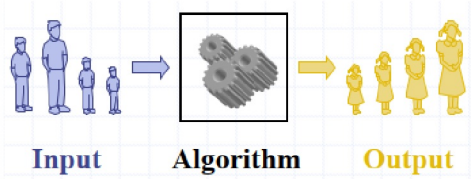
Analysis of an Algorithm

Slides based on

- T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein: Introduction to Algorithms
- R. Sedgewick and K. Wayne: Algorithms

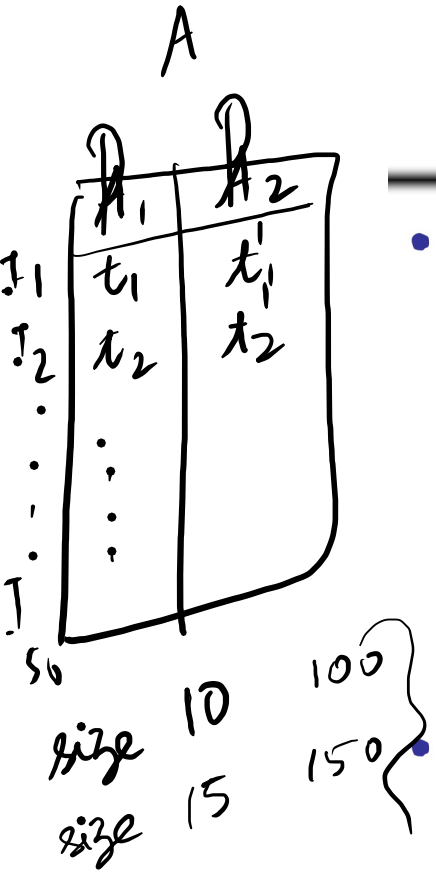
Algorithm

- What is an algorithm?
  - A step-by-step procedure for solving a problem in a finite amount of time.



- Questions of interest while executing the program/code of an algorithm:
  - How long will the program take to produce the output?
    - Will it produce an output? (Will it halt/terminate?)
    - Will the output be correct?
  - Will the execution run out of memory?

Design of Algorithm



Aim for analyzing an algorithm

- What is the goal of analysis of algorithms?
  - To measure the resources required to run an algorithm (on a computer)
  - To compare algorithms mainly in terms of **running time** but also in terms of other factors (e.g., memory requirements, programmer's effort etc.)
- What do we mean by running time analysis?
  - Determine how running time increases as the size of the problem increases.

input size

n

$T(n) = f(n) \approx 10n$

$T_1(n) \approx 10n$

$T_2(n) \approx 10n+14$

Input Size

- Input size (number of elements in the input)

  n is reserved for input size

  - size of an array
  - degree of a polynomial
  - # of elements in a matrix
  - # of bits in the binary representation of the input
  - vertices and edges in a graph

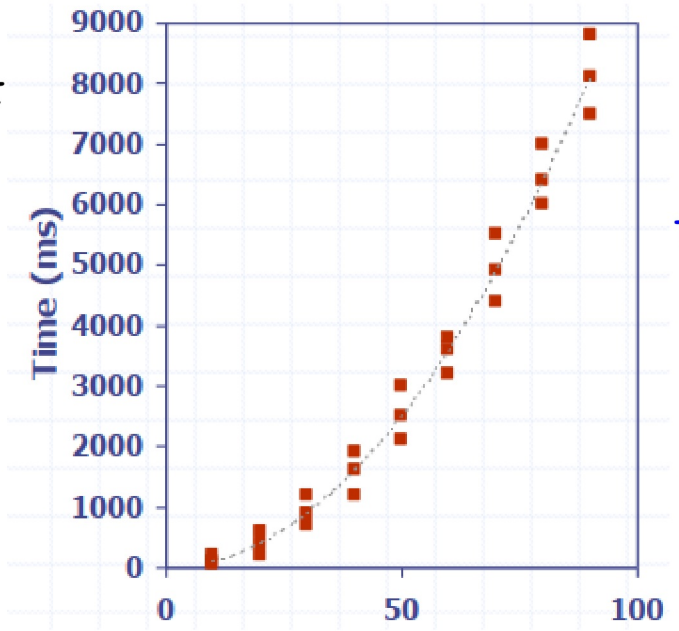
Helps come up with an objective measure to analyse an algorithm.

Experimental/Scientific Analysis

Observe -> Hypothesize -> Predict -> Verify -> Validate -> hypothesize -> predict ...

Write a program for the algorithm

- Run the program with inputs of varying size and composition
- Use a method to get an accurate measure of the actual running time
- Plot the result



$T(n) \approx 0.9n^2$

Experimental Analysis - Example

Problem: 3-SUM Given n distinct integers, how many triples sum to zero?

Brute force algorithm: Assume that the input is an array A of size n.

```
Triples ← 0
For i = 1 to n-2
    For j = i+1 to n-1
        For k = j+1 to n
            If A[i]+A[j]+A[k] = 0 then
                Triples ← Triples + 1
Print Triples
```

Instance: 30, -40, -10, -20, 40, 0, 10, 5

o/p: 4

Tuple #	Tuple	Sum
1	30, -40, -10	-20
2	30, -40, -20	-30
3	30, -40, 40	30
4	30, -40, 0	-10
5	30, -40, 10	0
6	30, -40, 5	-5
7	30, -10, -20	0
8	30, -10, 40	60
…		
	-40, 40, 0	0
…		
	-10, 0, 10	0
…		
56	0, 10, 5	15

Experimental Analysis - Example

A Java Program for the algorithm:

```
Public class ThreeSum
{
Public static int Triples(int[], A)
{
int n = A.length;
int Triples = 0;
For (int i=0; i<n-3; i++)
   For (int j=i+1; j<n-2; j++)
     For (int k=j+1; k<n-1; k++)
        If (A[i]+A[j]+A[k] == 0)
           Triples++;
Return Triples;
}
```

```
public static void
Main(string[] args)
{
In in=new In(args[0]);
int[]A=in.readAllInts();
Stdout.Println(Triples(A));
}
}
```