

580: Algorithms Preparation 1

We will begin by studying, in a surprising amount of detail given its simplicity, the following computational problem. It will be helpful if you have just looked over the problem and the solution before the initial lectures.

Problem: (Sequence Search). Given an *ordered* sequence $L = \langle a_1, \dots, a_N \rangle$ of N integers, and an integer k , determine whether k is in L or not.

This problem is specified using mathematical notation so that it is independent of any particular language. The first proposed solution we will look at is as follows.

```
1: procedure SIM
2:   for  $e$  in  $L$  d
3:     if  $e == k$ 
4:       return True
5:     end if
6:   end for
7:   return False
8: end procedure
```

This solution is written in *pseudocode*. I hope it is clear to you what this solution does, meaning that you could implement it in a language of your choice if you had to, picking an appropriate data structure for the sequence.

Pseudocode has no formal syntax rules. It is often a mixture of natural language, mathematical notation and constructs that resemble common programming languages. If it is clear and unambiguous it is OK. Where there might be ambiguity, I will follow the pseudocode conventions set out in the book *Introduction to Algorithms* (Cormen *et al.* 2009). (This is the main course textbook.) They are:

- Variables are local to the given procedure.
- A variable representing an array or object is treated as a pointer.
- Parameters are passed to procedures by value. When objects are passed as parameters, the pointer is copied.

The course also assumes the same *computational model* as Cormen: a single-processor random-access machine (RAM). All instructions are executed sequentially. There are no concurrent operations.