The Random-Access Machine

Christophe Rhodes

Motivation

- model for real computers
- simple enough to reason about

Definition

A random-access machine is a computer with:

- an unbounded amount of memory
  - addressable by integers
  - each memory access takes a constant time step
- a program made up of simple instructions
  - executed one-at-a-time
  - each simple instruction takes a constant time step
- program combinations using functions, loops, conditionals
  - the combination itself takes a constant time step
  - the result of combination takes longer

Time and Space

Running time

The number of constant time steps taken

- memory access
- simple instructions executed
- combinations executed

Space used

The number of memory locations used

- in addition to the space used by the input: “additional space used”

Example

```
function EXERCISE1($v$)
    $a \leftarrow 0$; $b \leftarrow 0$                              ▷ 2
    for $0 \le i < \text{LENGTH}(v)$ do                ▷ $n = \text{LENGTH}(v)$
        if $v[i] > b$ then                                  ▷ $n$
            if $v[i] > a$ then                              ▷ up to $n$
                $b \leftarrow a$                            ▷ up to $n$
                $a \leftarrow v[i]$                         ▷ up to $n$
            else
                $b \leftarrow v[i]$
            end if
        end if
    end for
    return $b$                                              ▷ 1
end function
```

The point of all of this

In software design and implementation, we often want to:

- minimize the time the program takes to run
- minimize the resources (*e.g.* memory, disk space) the program consumes

The Random-Access Machine model

- simple enough to compute answers, at least approximately
- realistic enough to be a guide to real computers

Work

1. Reading
    - CLRS, section 2.2
2. Exercises from CLRS: 2.2-1, 2.2-4