Algorithm Discovery

Topics:

Iterative Operations
Algorithmic Problem Solving

Prof. Dr. Slim Abdennadher

22.11.2007

Warm-Up

Multiplication of two integers N and M via addition

• **Example 1:** N=3 and M=4

```
N M result
3 4 0
3 3 3
3 2 6
3 1 9
3 0 12
```

• Algorithm:

```
get N, M
set result to 0
while (M>0) {
   set result to result + N
   set M to M - 1
}
print result
```

Lists: Sequential Search

Problem: Find the phone number of a given Name in an (unsorted) list of names and their phone numbers

N1 T1

N2 T2

. . .

N1000 T1000

©Slim Abdennadher

3

Sequential Search: First Attempt

Problem: Find the phone number of a given Name in an (unsorted) list of names and their phone numbers

```
get values for Name, N1, ...,N1000, T1, ...,T1000
if Name = N1 then print the value of T1
endif
if Name = N2 then print the value of T2
endif
...
if Name = N999 then print the value of T999
endif
if Name = N1000 then print the value of T1000
endif
```

Sequential Search: Using a Loop

Problem: Find the phone number of a given Name in an (unsorted) list of names and their phone numbers

```
get values for Name, N1, ..., N1000, T1, ..., T1000
set i to 1
set Found to NO
while (found = No or i<=1000)
    if (Name = Ni) then
       print the value of Ti
       set Found to YES
    else
       set i to i+1
    endif
}
if Found = NO then
 print 'sorry, name is not in directory'
endif
```

• **Problem:** Given a list of values A1, ..., An, find the largest value and its (first) location

• Example:

```
A1 A2 A3 A4 A5 A6 A7

Value 5 2 8 4 8 6 4
```

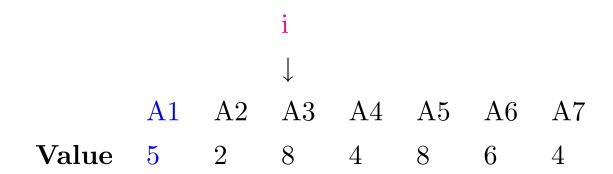
The largest is 8 at location 3

• Idea: Go through the entire list, at each iteration find the largest-so-far and record its location

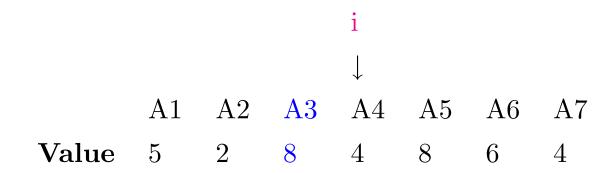
```
i \downarrow A1 \quad A2 \quad A3 \quad A4 \quad A5 \quad A6 \quad A7 Value 5 \quad 2 \quad 8 \quad 4 \quad 8 \quad 6 \quad 4
```

To begin with,

set largest-so-far to the value of A1 set location to 1 set i to 2



Compare A1 and A2 largest-so-far still holds the value A1 set i to i+1



Compare A1 and A3
largest-so-far now holds the value A3
location is 3
set i to i+1

A1 A2 A3 A4 A5 A6 A7

Value 5 2 8 4 8 6 4

Continue the similar process until i=8

```
get n
get A1, A2, ..., An
set largest-so-far to A1
set location to 1
set i to 2
while (i = < n)
{
      if Ai > largest-so-far then
         set largest-so-far to Ai
         set location to i
      endif
      set i to i + 1
print largest-so-far
print location
```