

## **Solution**    **Section 2.2 – Algorithms**

### **Exercise**

List all the steps used by the Algorithm 1 to find the maximum of the list

1, 8, 12, 9, 11, 2, 14, 5, 10, 4.

### **Solution**

The **for** loop then begins, with  $i$  set equal from 2 to  $n = 10$  (number of the sequence).

The statement of the loop is executed since  $2 < 10$ . This is an **if ... then** statement.

$max := 1$

**for**  $i := 2$  to 10

**if**  $max < a_i$  **then**  $max := a_i$

$a_i = a_2 = 8$ , since  $1 < 8$ , then  $max := 8$

$a_i = a_3 = 12$ , since  $8 < 12$ , then  $max := 12$

$a_i = a_4 = 9$ , since  $12 < 9$  **is not true**, then  $max := 12$

$a_i = a_5 = 11$ , since  $12 < 11$  **is not true**, then  $max := 12$

$a_i = a_6 = 2$ , since  $12 < 2$  **is not true**, then  $max := 12$

$a_i = a_7 = 14$ , since  $12 < 14$ , then  $max := 14$

$a_i = a_8 = 5$ , since  $14 < 5$  **is not true**, then  $max := 14$

$a_i = a_9 = 10$ , since  $14 < 10$  **is not true**, then  $max := 14$

$a_i = a_{10} = 4$ , since  $14 < 4$  **is not true**, then  $max := 14$

Therefore **max** has the value 14

### **Exercise**

Devise an algorithm that finds the sum of all the integers in a list.

### **Solution**

**Procedure**  $sum \{a_1, a_2, \dots, a_n : integers\}$

$sum := a_1$

**for**  $i := 2$  to  $n$

$sum := sum + a_i$

**return**  $sum$  { is the sum of all the elements in the list }

### ***Exercise***

Describe an algorithm that takes as an input a list of  $n$  integers and produces as output the largest difference obtained by subtracting an integer in the list from the one following it.

### **Solution**

For  $i$  going from 1 through  $n - 1$ , compute the value of the  $(i + 1)^{st}$  element in the list minus the  $i^{st}$  element in the list. If this is larger than the answer, reset the answer to be this value.

### ***Exercise***

Describe an algorithm that takes as an input a list of  $n$  integers in non-decreasing order and produces the list of all values that occur more than once.

### **Solution**

**Procedure** negatives  $\{a_1, a_2, \dots, a_n : \text{integers}\}$   
 $k := 0$   
**for**  $i := 1$  to  $n$   
    **if**  $a_i < 0$  **then**  $k := k + 1$   
**return**  $k$  { the number of negative integers in the list }

### ***Exercise***

Describe an algorithm that takes as an input a list of  $n$  integers and finds the location of the last even integer in the list or returns 0 if there are no even integers in the list.

### **Solution**

**Procedure** last even loction  $\{a_1, a_2, \dots, a_n : \text{integers}\}$   
 $k := 0$   
**for**  $i := 1$  to  $n$   
    **if**  $a_i$  is even **then**  $k := i$   
**return**  $k$  { is the desired location (or 0 if there are no evens) }

### ***Exercise***

Describe an algorithm that interchanges the values of the variables  $x$  and  $y$ , using only assignments. What is the minimum number of assignment statements needed to do this?

### **Solution**

We cannot simply write  $x := y$  followed by  $y := x$ .

$temp := x$

$x := y$

$y := temp$

### Exercise

List all the steps used to search for 9 in the sequence 1, 3, 4, 5, 6, 7, 9, 11 using

a) a linear search

b) a binary search

### Solution

a) Note that  $n = 8$  and  $x = 9$ .

**procedure** linear\_search ( $x$ : integer;  $a_1, a_2, \dots, a_n$  : integers)

$i := 1$

**while** ( $i \leq 8$  and  $(i \leq 8 \text{ and } 9 \neq a_i)$ )

$i := i + 1$

The **while** loop is executed as long as  $i \leq 8$  and the  $i^{st}$  element is not equal to 9.

$i = 1, a_1 = 1; 9 \neq 1$

$i = 2, a_2 = 3; 9 \neq 3$

$i = 3, a_3 = 4; 9 \neq 4$

$i = 4, a_4 = 5; 9 \neq 5$

$i = 5, a_5 = 6; 9 \neq 6$

$i = 6, a_6 = 7; 9 \neq 7$

$i = 7, a_7 = 7; 9 = 9$

Therefore the body of the loop is not executed (so  $i$  is still equal to 7), and control passes beyond the loop.

**if**  $i \leq n$  **then**  $location := i$

**else**  $location := 0$

The else clause is not executed. This completes the procedure, so  $location$  has the correct value, namely 7, which indicates the location of the element  $x$  in the list: 9 is the seventh element.

b) **procedure** linear\_search ( $x$ : integer;  $a_1, a_2, \dots, a_n$  : increasing integers)

$i := 1$

$j := 8$

**while**  $i < j$

The while step is executed, first  $m = \frac{1+8}{2} = 4$

Then since  $x (= 9)$  is greater than  $a_4 (= 5)$ , the statement  $i := m + 1$  is executed, so  $i$  has the value 5.

$$i = 4 + 1 = 5, \quad m = \frac{5+8}{2} = 6 \quad x(=9) > a_6(=6)$$

$$i = 6 + 1 = 7, \quad m = \frac{7+8}{2} = 7 \quad x(=9) > a_7(=9) \text{ fails thus } j := m, \text{ so } j := 7$$

At this point  $i \not< j$ , the condition  $x = a_i$  is true, location is set to 7, as it should be, and the algorithm is finished.

### **Exercise**

Describe an algorithm that inserts an integer  $x$  in the appropriate position into the list  $a_1, a_2, \dots, a_n$  of integers that are in increasing order.

### **Solution**

**procedure** insert  $(x, a_1, a_2, \dots, a_n : \text{integers})$

$$a_{n+1} := x + 1$$

$$i := 1$$

**while**  $x > a_i$

$$i := i + 1 \quad \{\text{The loop ends when } i \text{ is the index for } x\}$$

**for**  $j := 0 \text{ to } n - i$   $\{\text{Shove the rest of the list to the right}\}$

$$a_{n-j+1} := a_{n-j}$$

$$a_i := x$$

$\{x \text{ has been inserted into the correct spot in the list, now of length } n + 1\}$