4. Describe an algorithm that takes as 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.

```
procedure maxDifferent (n ∈ Z, a1 a2. . . an ∈ Z)
    begin
        if n < 2  then return error
        m := a2 − a1
        for i := 2 to n − 1 do
            if m < an+1 − an then m := an+1 − an
        end for
        return m
    end procedure
```

6. Describe an algorithm that takes as input a list of \( n \) integers and finds the number of negative integers in the list.

```
procedure count_negative(a1, a2, …, an: integers)
    count := 0
    for i := 1 to n
        if ai <0 then k := k+1
    return k
```

8. Describe an algorithm that takes as input a list of \( n \) distinct integers and finds the location of the largest even integer in the list or returns 0 if there are no even integers in the list.

```
Procedure even( a1 ,a2 ,. . .,an :distinct integers)
    begin
        L:=0
        m:= a1
        for i:=1 to n
            if ai is even and ai≥m then
                begin
                    L:=i
                    m:= ai
                end
        return L
    end
```
10. Devise an algorithm to compute \( x^n \), where \( x \) is a real number and \( n \) is an integer.

Procedure exponent (\( x: \text{real}; n: \text{integer} \))
begin
\( P:=0 \)
if \( x:=0 \) then return 0
if \( n:=0 \) then return 1
else begin
for \( i:= 1 \) to \(|n|\)
\( P:=P*x \)
if \( n>0 \) then return \( P \) else return \( 1/P \)
end
end

12. Describe an algorithm that uses only assignment statements that replaces the triple \((x, y, z)\) with \((y, z, x)\). What is the minimum number of assignment statements needed?

Procedure exchange (\(x,y,z: \text{integer} \))
begin
\( \text{tmp}:= x \)
\( x:= y \)
\( y:= z \)
\( z:= \text{tmp} \)
return \((x,y,z)\)
end

18. Describe an algorithm that locates the last occurrence of the smallest element in a finite list of integers, where the integers in the list are not necessarily distinct.

procedure find_min(\(a_0, a_1, \ldots, a_n: \text{positive integer} \))
begin
\( \text{min} := a_0 \)
\( \text{index} := 0 \)
for \( j := 1 \) to \( n \)
if \( \text{min} \geq a_j \) then
begin
\( \text{min} := a_j \)
\( \text{index} := j \)
end
return \( j \)
end