

2. Transformá cada uno de los algoritmos anteriores en uno equivalente que utilice **for ... to** .

(a) Inicializar cada componente del arreglo con el valor 0.

```
proc todos_0(out a: array[1..n] of nat)
  for i := 1 to n do
    a[i] := 0
  od
end proc
```

(b) Inicializar el arreglo con los primeros n números naturales positivos.

```
proc primeros_naturales(out a: array[1..n] of nat)
  for i := 1 to n do
    a[i] := i
  od
end proc
```

(c) Inicializar el arreglo con los primeros n números naturales impares.

```
proc primeros_naturales_impares(out a: array[1..n] of nat)
  for i := 1 to n do
    a[i] := i+(i-1)
  od
end proc
```

(d) Incrementar las posiciones impares del arreglo y dejar intactas las posiciones pares.

```
proc incrementar_pos_impares(in/out a: array[1..n] of nat)
  for i := 1 to n do
    if i mod 2 ≠ 0 then
      a[i] := a[i]+1
    fi
  od
end proc
```