

# Práctica 5 : Demostración de correctitud de programas con ciclos

Algoritmos y Estructuras de Datos - Turno Vespertino

13 de de septiembre de 2023

# Ejercicio 1

Retomamos un ejemplo de la clase pasada.

```
proc productoria (in a: seq⟨ℤ⟩, out prod: ℤ) {  
  requiere {Pre : |a| mod 2 = 0}  
  asegura {Post : prod =  $\prod_{i=0}^{|a|-1} a[i]$ }  
}
```

```
i := 0;  
prod := 1;  
while (i < a.size() - 1) do  
  prod := prod*a[i]*a[i+1];  
  i := i + 2  
endwhile
```

¿Es correcto el programa? En caso de serlo, demostrar por qué.

## Ejercicio 2

Especificación del problema noHaySietes.

```
proc noHaySietes (in s: seq( $\mathbb{Z}$ ), out res: Bool) {  
  requiere {Pre : True}  
  asegura {Post : res = true  $\leftrightarrow$  ( $\forall k : \mathbb{Z}$ )( $0 \leq k < |s| \longrightarrow_L s[k] \neq 7$ )}  
}
```

## Ejercicio 2

Solución posible implementada en SmallLang. ¿Es correcta? En ambos casos demostrar por qué.

```
i := 0;
j := -1;
while (i < s.size()) do
    if (s[i] = 7)
        j := i
    else
        skip
    endif;
    i := i+1
endwhile;
if (j = -1)
    res := true
else
    res := false
endif
```

## Ejercicio 3

¿Qué hacemos en un escenario de esta pinta?

$\{Pre\}$

$S_1;$

while ( $B_1$ ) do

...

endwhile;

$S_2;$

while ( $B_2$ ) do

...

endwhile

$\{Post\}$