

Programación con invariantes

Repaso - ciclos

Sintaxis de un ciclo:

```
1 while(B) {  
2   //cuerpo del ciclo  
3}
```

- El ciclo se repite continuamente mientras la **guarda** B se cumpla. Cada repetición es una **iteración**.
- El ciclo **termina** cuando no se cumpla la guarda.
- Al salir, en caso de que el ciclo terminara, el estado resultante es el mismo que el del final de la última iteración.

Ejemplo

```
1 bool hayMayorACero(vector<int> v) {  
2     int i = 0;  
3     bool encuentre = false;  
4     int n = v.size();  
5     while(i < n) {  
6         encuentre = encuentre || v[i] > 0;  
7         i = i + 1;  
8     }  
9     return encuentre;  
10 }
```



¿Son Invariantes del ciclo?

- ▶ $I \equiv i \leq n$
- ▶ $I \equiv i < n \wedge (\text{encontre} = \mathbf{true} \vee \text{encontre} = \mathbf{false})$
- ▶ $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \mathbf{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k \leq i \wedge_L v[k] > 0$
- ▶ $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \mathbf{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k < i \wedge_L v[k] > 0$

Repaso - Invariantes

- **Definición.** Un predicado I es un invariante de un ciclo si:
 1. vale antes de comenzar el ciclo, y
 2. si vale $I \wedge B$ al comenzar una iteración arbitraria, entonces sigue valiendo I al finalizar la ejecución del cuerpo del ciclo.

```
1 //vale I
2 while(B) {
3   //vale I  $\wedge$  B ← Principio de iteración
4   Cuerpo del ciclo
5   //vale I ← Final de iteración
3}
```

Ejemplo

```
1 bool hayMayorACero(vector<int> v) {  
2     int i = 0;  
3     bool encuentre = false;  
4     int n = v.size();  
5     while(i < n) {  
6         encuentre = encuentre || v[i] > 0;  
7         i = i + 1;  
8     }  
9     return encuentre;  
10 }
```



¿Son Invariantes del ciclo?

- ▶ $I \equiv i \leq n$
- ▶ $I \equiv i < n \wedge (\text{encontre} = \mathbf{true} \vee \text{encontre} = \mathbf{false})$
- ▶ $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \mathbf{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k \leq i \wedge_L v[k] > 0$
- ▶ $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \mathbf{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k < i \wedge_L v[k] > 0$

Ejemplo

```
1 bool hayMayorACero(vector<int> v) {  
2     int i = 0;  
3     bool encuentre = false;  
4     int n = v.size();  
5     while(i < n) {  
6         encuentre = encuentre || v[i] > 0;  
7         i = i + 1;  
8     }  
9     return encuentre;  
10 }
```

- Sea $v = \{-1, 2, -3\}$

Principio de iteración

Final de iteración

Principio de Iteración

Iteración	i	v[i]	encontre
1	0	-1	false
2	1	2	false
3	2	-3	true

Final de Iteración

Iteración	encontre	i	v[i]
1	false	1	2
2	true	2	-3
3	true	3	⊥

Ejemplo

Principio de Iteración

Iteración	i	v[i]	encontre
1	0	-1	false
2	1	2	false
3	2	-3	true

$v = \{-1, 2, -3\}$
 $n = v.size()$

Final de Iteración

Iteración	encontre	i	v[i]
1	false	1	2
2	true	2	-3
3	true	3	\perp

¿Son Invariantes del ciclo?

- ▶ $I \equiv i \leq n$
- ▶ $I \equiv i < n \wedge (\text{encontre} = \mathbf{true} \vee \text{encontre} = \mathbf{false})$
- ▶ $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \mathbf{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k \leq i \wedge_L v[k] > 0$
- ▶ $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \mathbf{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k < i \wedge_L v[k] > 0$

Ejemplo

Principio de Iteración

Iteración	i	v[i]	encontre
1	0	-1	false
2	1	2	false
3	2	-3	true

$v = \{-1, 2, -3\}$
 $n = v.size()$

Final de Iteración

Iteración	encontre	i	v[i]
1	false	1	2
2	true	2	-3
3	true	3	\perp

¿Son Invariantes?

▶ $I \equiv i < n$ 👍

▶ $I \equiv i \leq n \wedge (encontre = \mathbf{true} \vee encontre = \mathbf{false})$ 👍

Ejemplo

Principio de Iteración

Iteración	i	v[i]	encontre
1	0	-1	false
2	1	2	false
3	2	-3	true

$v = \{-1, 2, -3\}$
 $n = v.size()$

Final de Iteración

Iteración	encontre	i	v[i]
1	false	1	2
2	true	2	-3
3	true	3	\perp

¿Es un Invariante?

► $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \text{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k \leq i \wedge_L v[k] > 0$

Ejemplo

Principio de Iteración

Iteración	i	v[i]	encontre
1	0	-1	false
2	1	2	false
3	2	-3	true

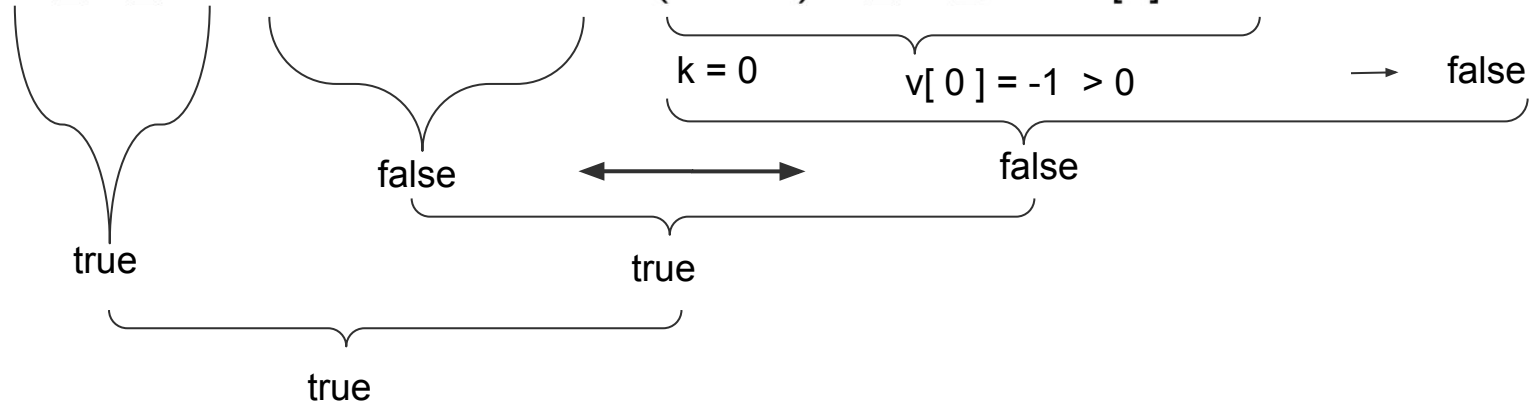
$v = \{-1, 2, -3\}$
 $n = v.size()$

Final de Iteración

Iteración	encontre	i	v[i]
1	false	1	2
2	true	2	-3
3	true	3	\perp

¿Es un Invariante?

► $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \text{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k \leq i \wedge_L v[k] > 0$



Ejemplo

Principio de Iteración

Iteración	i	v[i]	encontre
1	0	-1	false
2	1	2	false
3	2	-3	true

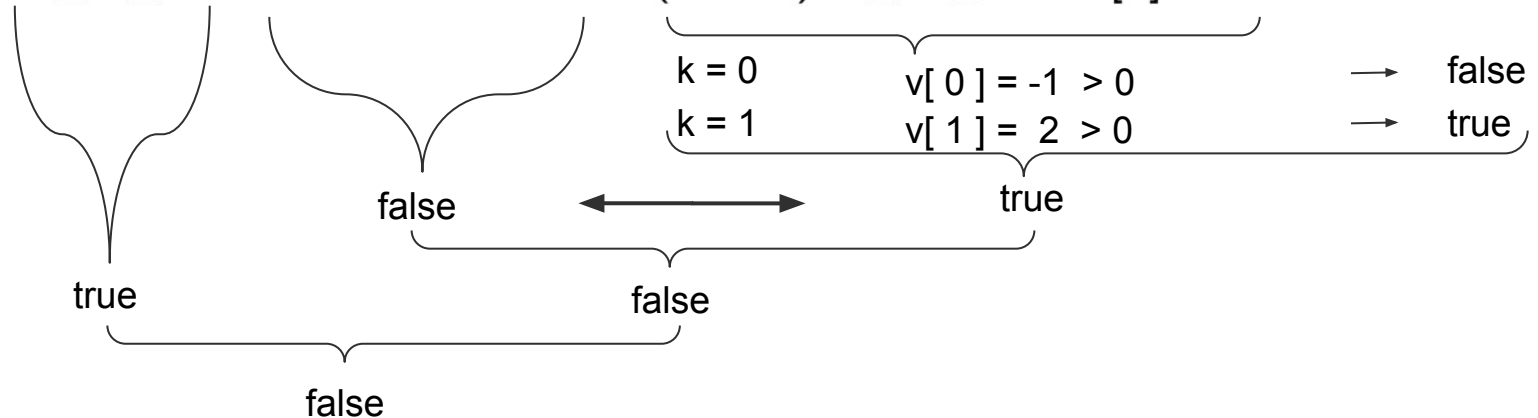
$v = \{-1, 2, -3\}$
 $n = v.size()$

Final de Iteración

Iteración	encontre	i	v[i]
1	false	1	2
2	true	2	-3
3	true	3	\perp

¿Es un Invariante?

► $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \text{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k \leq i \wedge_L v[k] > 0$



Ejemplo

Principio de Iteración




Iteración	i	v[i]	encontre
1	0	-1	false
2	1	2	false
3	2	-3	true

$v = \{-1, 2, -3\}$
 $n = v.size()$

Final de Iteración

Iteración	encontre	i	v[i]
1	false	1	2
2	true	2	-3
3	true	3	\perp

¿Son Invariantes?

- ▶ $I \equiv i < n$ 
- ▶ $I \equiv i < n \wedge (\text{encontre} = \mathbf{true} \vee \text{encontre} = \mathbf{false})$ 
- ▶ $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \mathbf{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k \leq i \wedge_L v[k] > 0$ 

Ejemplo

Principio de Iteración

Iteración	i	v[i]	encontre
1	0	-1	false
2	1	2	false
3	2	-3	true

$v = \{-1, 2, -3\}$
 $n = v.size()$

Final de Iteración

Iteración	encontre	i	v[i]
1	false	1	2
2	true	2	-3
3	true	3	\perp

¿Es un Invariante?

► $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \mathbf{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k < i \wedge_L v[k] > 0$

Ejemplo

Principio de Iteración

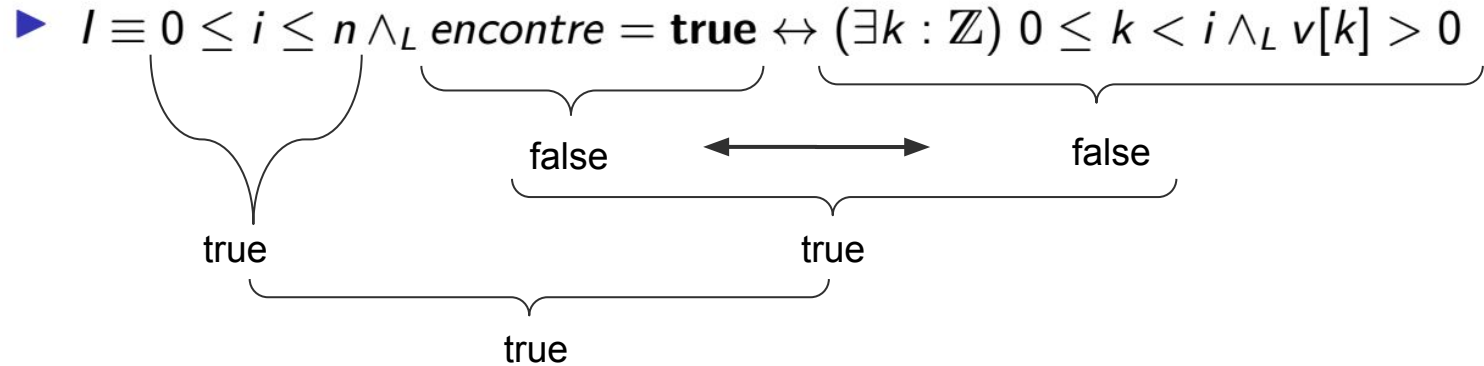
Iteración	i	v[i]	encontre
1	0	-1	false
2	1	2	false
3	2	-3	true

$v = \{-1, 2, -3\}$
 $n = v.size()$

Final de Iteración

Iteración	encontre	i	v[i]
1	false	1	2
2	true	2	-3
3	true	3	\perp

¿Es un Invariante?



Ejemplo

Principio de Iteración

Iteración	i	v[i]	encontre
1	0	-1	false
2	1	2	false
3	2	-3	true

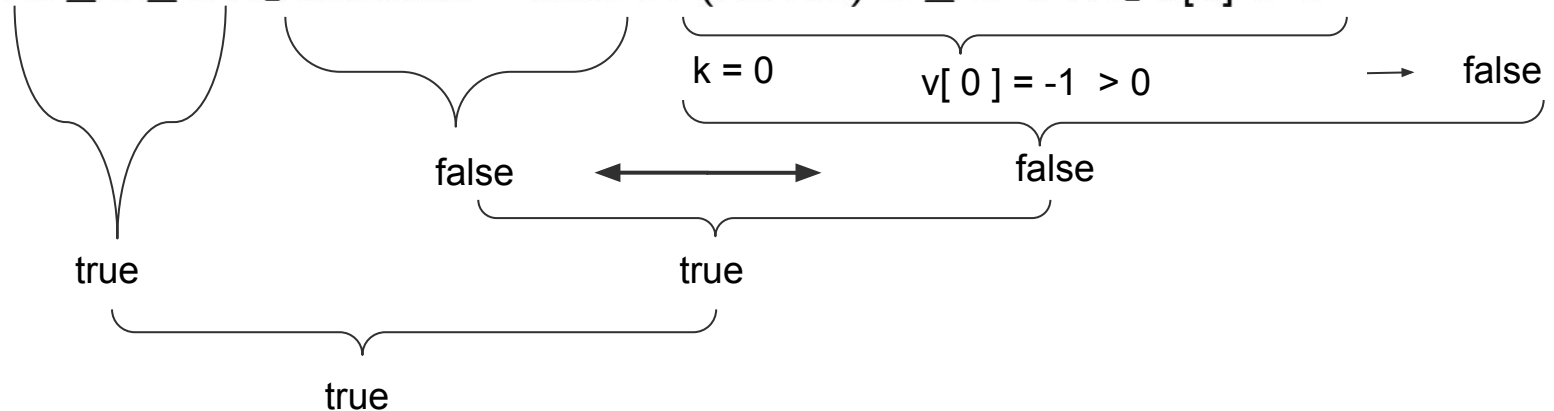
$v = \{-1, 2, -3\}$
 $n = v.size()$

Final de Iteración

Iteración	encontre	i	v[i]
1	false	1	2
2	true	2	-3
3	true	3	\perp

¿Es un Invariante?

► $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \text{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k < i \wedge_L v[k] > 0$



Ejemplo

Principio de Iteración

Iteración	i	v[i]	encontre
1	0	-1	false
2	1	2	false
3	2	-3	true

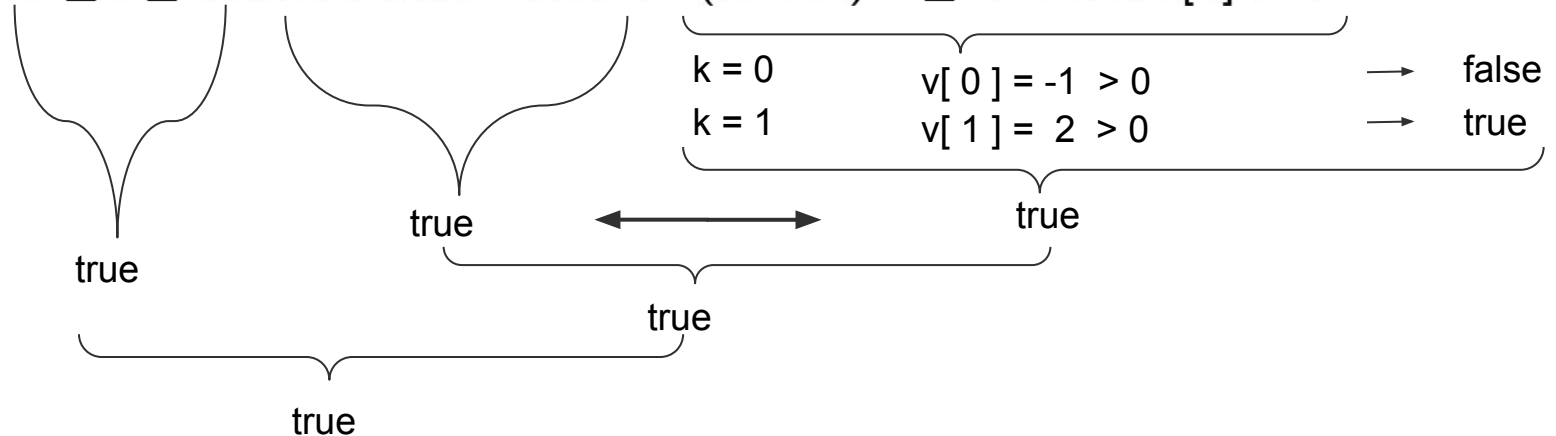
$v = \{-1, 2, -3\}$
 $n = v.size()$

Final de Iteración

Iteración	encontre	i	v[i]
1	false	1	2
2	true	2	-3
3	true	3	\perp

¿Es un Invariante?

► $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \text{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k < i \wedge_L v[k] > 0$



Ejemplo

Principio de Iteración

Iteración	i	v[i]	encontre
1	0	-1	false
2	1	2	false
3	2	-3	true

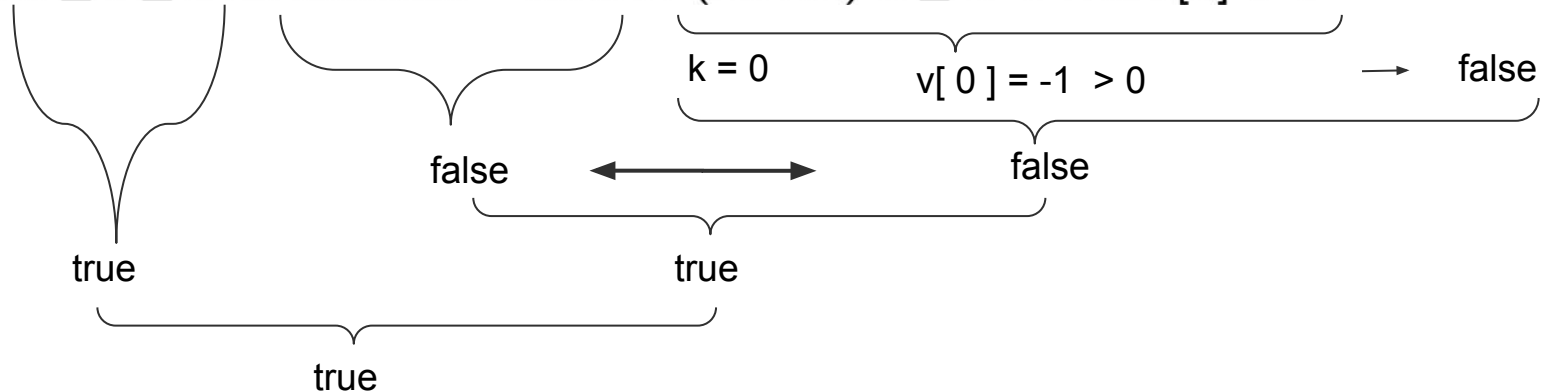
$v = \{-1, 2, -3\}$
 $n = v.size()$

Final de Iteración

Iteración	encontre	i	v[i]
1	false	1	2
2	true	2	-3
3	true	3	⊥

¿Es un Invariante?

► $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \text{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k < i \wedge_L v[k] > 0$



Ejemplo

Principio de Iteración

Iteración	i	v[i]	encontre
1	0	-1	false
2	1	2	false
3	2	-3	true

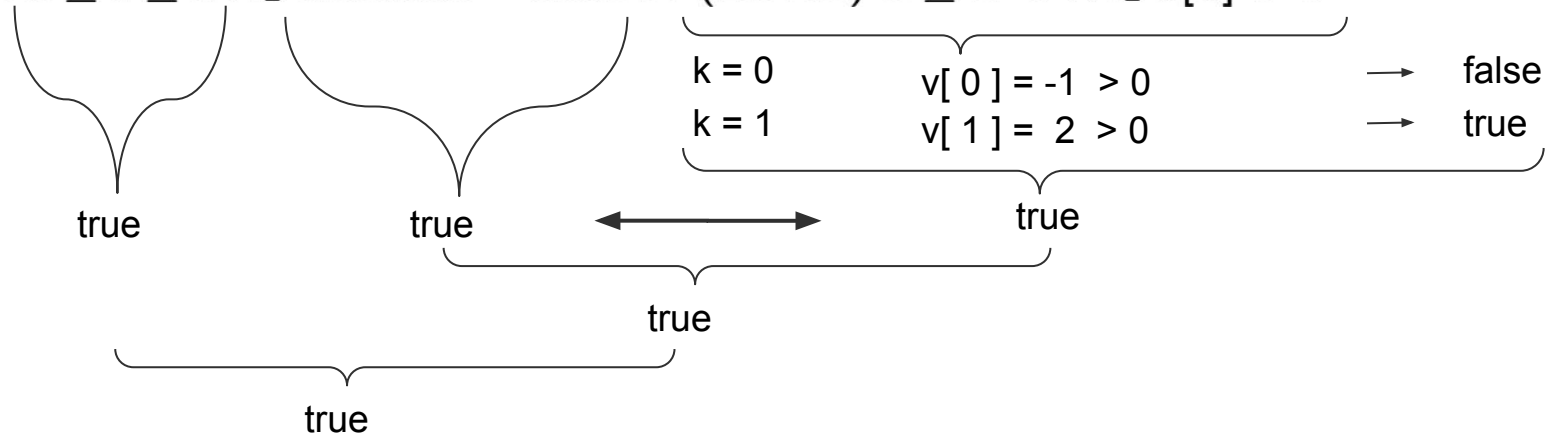
$v = \{-1, 2, -3\}$
 $n = v.size()$

Final de Iteración

Iteración	encontre	i	v[i]
1	false	1	2
2	true	2	-3
3	true	3	\perp

¿Es un Invariante?

► $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \text{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k < i \wedge_L v[k] > 0$



Ejemplo

Principio de Iteración

Iteración	i	v[i]	encontre
1	0	-1	false
2	1	2	false
3	2	-3	true

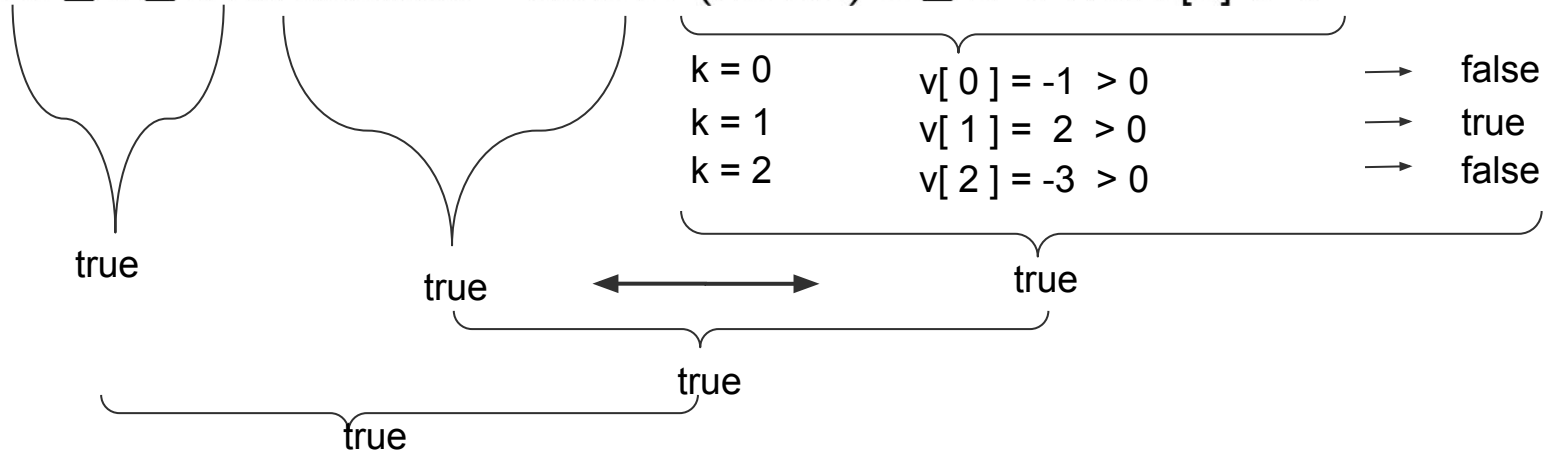
$v = \{-1, 2, -3\}$
 $n = v.size()$

Final de Iteración

Iteración	encontre	i	v[i]
1	false	1	2
2	true	2	-3
3	true	3	\perp

¿Es un Invariante?

► $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \text{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k < i \wedge_L v[k] > 0$



Ejemplo

Principio de Iteración

Iteración	i	v[i]	encontre
1	0	-1	false
2	1	2	false
3	2	-3	true

$v = \{-1, 2, -3\}$
 $n = v.size()$

Final de Iteración

Iteración	encontre	i	v[i]
1	false	1	2
2	true	2	-3
3	true	3	⊥

¿Son Invariantes del ciclo?

- ▶ $I \equiv i < n$ 👍
- ▶ $I \equiv i < n \wedge (\text{encontre} = \text{true} \vee \text{encontre} = \text{false})$ 👍
- ▶ $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \text{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k \leq i \wedge_L v[k] > 0$ 🙅
- ▶ $I \equiv 0 \leq i \leq n \wedge_L \text{encontre} = \text{true} \leftrightarrow (\exists k : \mathbb{Z}) 0 \leq k < i \wedge_L v[k] > 0$ 👍



No todos nos van a servir para poder demostrar la correctitud parcial del ciclo! (en particular $I \wedge \neg B \Rightarrow Q_c$)

Repaso - Teorema del Invariante

Sea P_C la precondition del ciclo, Q_C la postcondición, B la guarda e I un invariante del ciclo. Si se cumple:

1. $P_C \Rightarrow I$,
2. $\{I \wedge B\}$ cuerpo del ciclo $\{I\}$,
3. $I \wedge \neg B \Rightarrow Q_C$,

entonces el ciclo es parcialmente correcto (si termina, termina en Q_C).