

# Especificación y Verificación Formal de Sistemas Distribuidos con TLA+

## 03. Máquinas de estados

---

Axel Suárez Polo

November 3, 2022

BUAP

Modelando un programa de C

Programar 1: Programa en C

Programar 2: Programa en C (revisado)

# Modelando un programa de C

---

¿Cómo modelar el siguiente programa?

```
// Retorna un número entre 0 y 1000  
int obtenerNumero();  
  
int i = 0;  
int main() {  
    i = obtenerNumero();  
    i = i + 1;  
}
```

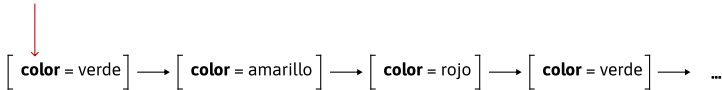
Listing 1: Programa en C

TLA+ utiliza el *modelo de comportamientos*.

- La ejecución de un programa es representada por un **comportamiento**.
- Un **comportamiento** es una secuencia ordenada de **estados**, ya sea finita o infinita.
- Un **estado** es una asignación de valores a variables.
- Un **programa** es modelado por un conjunto de **comportamientos**.

# El modelo de comportamientos

① *variable*



② *estado*

③ *comportamiento*

Figure 1: El comportamiento de un semáforo

# Pasos para modelar un programa

Para modelar un programa o sistema con TLA+ tenemos que saber 3 cosas:

- **Variables** del sistema
- **Estado** inicial
- Relación entre el **estado** actual y el **estado** siguiente

¿Cuáles son las **variables**, **estado inicial** y **relación entre estados** del programa?

```
// Retorna un número entre 0 y 1000  
int obtenerNumero();  
  
int i = 0;  
int main() {  
    i = obtenerNumero();  
    i = i + 1;  
}
```

Listing 2: Programa en C



Una forma de determinar esto es pensando en los comportamientos del programa.

$$[i:0] \rightarrow [i:10] \rightarrow [i:11]$$

# Pensando en comportamientos

Una forma de determinar esto es pensando en los comportamientos del programa.

$$[i:0] \rightarrow [i:10] \rightarrow [i:11]$$

$$[i:0] \rightarrow [i:99] \rightarrow [i:100]$$

**Figure 2:** Posibles comportamientos del programa

## Programar 1: Programa en C

---

- Un *comportamiento* **siempre es lineal**.
- El modelo de comportamientos requiere pensar sobre las relaciones entre los estados.
- El **conjunto de comportamientos** es la especificación del programa.

# Pensando en comportamientos

Tenemos que usar una fórmula lógica para representar la relación entre estados, lo que es imposible con las variables actuales.

$$[i:0] \rightarrow [i:10] \rightarrow [i:11]$$

$$[i:0] \rightarrow [i:11] \rightarrow [i:12]$$

Figure 3: Posibles comportamientos del programa

Los lenguajes de programación convencionales ocultan estado en diversas formas: **variables**, **apuntador de instrucción**, **pila de llamadas**, **memoria dinámica**, etc.

```
1  int i = 0;           // pc = "start"
2  int main() {
3      i = obtenerNumero(); // pc = "middle"
4      i = i + 1;         // pc = "end"
5  }
```

Listing 3: El programa en C con anotaciones

# Pensando en comportamientos

TLA+ requiere que pensemos en todo el estado de un sistema para la especificación.

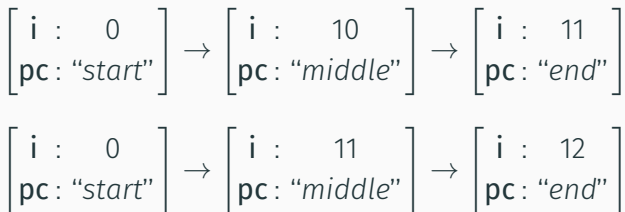


Figure 4: Posibles comportamientos del programa

## Programar 2: Programa en C (revisado)

---