

# Práctica 2

## Procedimientos

Introducción a la Programación  
1<sup>er</sup> Semestre de 2017

**Nota:** deben ejecutarse **todos** los programas en PYGOBSTONES, utilizando tableros de distintos tamaños. Al realizar una consulta (en forma personal o por e-mail), debe entregar su código fuente junto con los tableros de prueba, *incluso cuando* el ejercicio estuviera incompleto. Se espera que los tableros de prueba sean significativos para el ejercicio<sup>1</sup>.

Los ejercicios que corresponden a los **contenidos mínimos** recomendados se encuentran marcados con el símbolo ⊗.

### 1. Procedimientos Simples

#### Ejercicio 1

⊗ Corresponder las siguientes oraciones que describen propósitos con sus respectivos procedimientos.

- Pone una línea azul de longitud 3 hacia el este; el cabezal queda a tres celdas hacia el este de la celda actual.
- Pone una bolita azul y mueve el cabezal a la celda lindante al este.
- Reemplaza dos bolitas rojas por dos bolitas azules, dejando el cabezal en la misma posición.

```

procedure ReemplazarRojoPorAzul() {
    Sacar(Rojo); Poner(Azul)
    Sacar(Rojo); Poner(Azul)
}

procedure PonerAzulYMoverEste() {
    Poner(Azul)
    Mover(Este)
}

procedure PonerLineaAzul3() {
    PonerAzulYMoverEste()
    PonerAzulYMoverEste()
    PonerAzulYMoverEste()
}

procedure ReemplazarRojoPorAzul() {
    Sacar(Rojo); Sacar(Rojo)
    Poner(Azul); Poner(Azul)
}

```

<sup>1</sup>Por ejemplo, algunos de los tableros tienen que satisfacer las precondiciones.

Discutir la importancia de elegir un nombre apropiado para un procedimiento.

## Ejercicio 2

⊗ Escriba un procedimiento `PonerUnaDeCada` que ponga una bolita de cada color en la celda actual. Puede reutilizar buena parte del Ejercicio ??.

## Ejercicio 3

⊗ Tómese a lo sumo 3 minutos (cronometrados) para entender qué hace el siguiente procedimiento **sin ejecutarlo**. Cuando se acabe el tiempo, escriba lo que crea que hace el procedimiento.

```
procedure EntenderEnTresMinutos() {
    Poner(Azul); Mover(Este); Poner(Azul); Mover(Este); Poner(Azul);
    Mover(Oeste); Mover(Oeste); Mover(Norte);
    Poner(Azul); Mover(Este); Poner(Azul); Mover(Este);
    Poner(Azul); Mover(Oeste); Mover(Oeste);
    Mover(Norte); Poner(Azul); Mover(Este); Poner(Azul); Mover(Este);
    Poner(Azul); Mover(Oeste); Mover(Oeste); Mover(Norte);
    Mover(Norte);
    Poner(Azul); Mover(Este); Poner(Azul);
    Mover(Este); Poner(Azul); Mover(Oeste);
    Mover(Oeste); Mover(Norte); Poner(Azul); Mover(Este); Poner(Azul);
    Mover(Este); Poner(Azul); Mover(Oeste); Mover(Oeste);
    Mover(Norte); Poner(Azul); Mover(Este); Poner(Azul); Mover(Este);
    Poner(Azul); Mover(Oeste); Mover(Oeste); Mover(Norte); Mover(Norte);
    Poner(Azul); Mover(Este); Poner(Azul); Mover(Este);
    Poner(Azul); Mover(Oeste); Mover(Oeste); Mover(Norte);
    Poner(Azul); Mover(Este); Poner(Azul); Mover(Este); Poner(Azul);
    Mover(Oeste); Mover(Oeste); Mover(Norte);
    Poner(Azul); Mover(Este); Poner(Azul); Mover(Este); Poner(Azul);
    Mover(Oeste); Mover(Oeste); Mover(Norte);
}
```

## Ejercicio 4

⊗ Tómese a lo sumo 3 minutos (cronometrados) para entender qué hace el siguiente procedimiento **sin ejecutarlo**. Cuando se acabe el tiempo, escriba lo que crea que hace el procedimiento.

```
procedure EntenderEnTresMinutosBis() {
    PonerCuadradoAzul3(); Mover(Norte);
    PonerCuadradoAzul3(); Mover(Norte);
    PonerCuadradoAzul3();
}

procedure PonerCuadradoAzul3() {
    /* Pone un cuadrado azul de 3 por 3 hacia el noreste,
       dejando el cabezal en la columna actual y la celda
       al norte del rectángulo. */
    PonerLineaAzul3(); Mover(Norte)
    PonerLineaAzul3(); Mover(Norte)
    PonerLineaAzul3(); Mover(Norte)
}
```

```

procedure PonerLineaAzul3() {
    /* Pone una línea azul de longitud 3 hacia el este,
       dejando el cabezal en la celda actual */
    Poner(Azul); Mover(Este);
    Poner(Azul); Mover(Este);
    Poner(Azul); Mover(Oeste); Mover(Oeste);
}

```

### Ejercicio 5

Ejecute los procedimientos `EntenderEnTresMinutos` y `EntenderEnTresMinutosBis` (el tablero debería estar vacío y tener al menos 12 filas y 3 columnas, y el cabezal debería comenzar en el origen) y compare lo que hacen con lo que respondió en los ejercicios anteriores. ¿Cuál le resulta más entendible? ¿Qué habría que modificar en ambos programas para que los cuadrados sean rojos? Discuta con un compañero las ventajas de documentar y de dividir en subtareas. Tome como ejes la legibilidad, la abstracción, la reutilización, y el costo de mantenimiento del código.

### Ejercicio 6

⊗ Escriba un procedimiento `MoverOeste3` que mueva el cabezal tres celdas hacia el Oeste.

### Ejercicio 7

⊗ Escriba un procedimiento `PonerLineaMulticolor4` que ponga una línea de cuatro celdas hacia el este en la que cada celda tenga una bolita de cada color. El cabezal debe quedar en la celda inicial. Para ello, **debe** reutilizar los procedimientos `PonerUnaDeCada` (Ejercicio 2) y `MoverOeste3` (Ejercicio 6).

### Ejercicio 8

⊗ Escriba un procedimiento `PonerCuadradoMulticolor4` que ponga un cuadrado de  $4 \times 4$  celdas en la que cada celda tenga una bolita de cada color. El cabezal debe quedar en la celda inicial. Para ello, **debe** reutilizar el procedimiento `PonerLineaMulticolor4` (Ejercicio 7).

### Ejercicio 9

⊗ El siguiente procedimiento pone una escalera sin hacer uso de la técnica de división en subtareas. Identifique partes de código que se repitan, y extráigalas en procedimientos auxiliares. Cada procedimiento auxiliar debe tener un nombre adecuado, y debe documentarse su propósito con un comentario. Compare su solución con la de sus compañeros; ¿cuál versión le gusta más?

```

procedure PonerEscaleraAzul4() {
    Poner(Azul); Mover(Este); Poner(Azul); Mover(Norte);
    Poner(Azul); Mover(Este); Poner(Azul); Mover(Norte);
    Poner(Azul); Mover(Este); Poner(Azul); Mover(Norte);
    Poner(Azul); Mover(Este); Poner(Azul);
}

```

### Ejercicio 10

Si no lo hizo aún, escriba los propósitos y las precondiciones de todos los procedimientos de esta sección. Compare con sus compañeros.