**Ejercicio #1**. Juego (El círculo de monedas).

**Prueba de Escritorio**

Vamos a hacer la prueba de escritorio del siguiente caso de juego:

1. Turno Jugador 1:

- Saca dos monedas posición primera moneda 4

2. Turno Jugador 2:

- Saca dos monedas posición primera moneda 12

3. Turno Jugador 1:

- Saca una moneda posición moneda 2

4. Turno Jugador 2:

- Saca dos monedas posición primera moneda 3 (error, repite el turno)

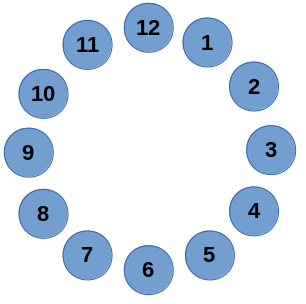
5. Turno Jugador 2:

- Saca dos monedas posición primera moneda 10

6. Turno Jugador 1:

- Saca una moneda posición moneda 8

7. Turno Jugador 2:

 - Saca dos monedas posición primera moneda 6

8. Turno Jugador 1:

- Saca una moneda posición moneda 3

9. Turno Jugador 2:

- Saca una moneda posición moneda 9

GANADOR JUGADOR 2

***Condición inicial del juego***

MONEDAS

Estado: m m m m m m m m m m m m

Posición: 1 2 3 4 5 6 7 8 9 10 11 12

Prueba de escritorio, se tomará en cuenta como variables principales: jugar, pos\_moneda, pos\_moneda\_uno, pos\_moneda\_dos, lista\_juego.

**Donde:**

jugar = true → Jugador 1

jugar = false → Jugador 2

op\_turno = 1 → Sacar una moneda

op\_turno = 2 → Sacar dos monedas (continuas)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Nro jugada** | **ope** | **jugar** | **resultado** | **op\_turno** | **Posición de la moneda** | **lista\_juego** |
| 0 | - | true | - | - | - | null |
| 0 | iniciar\_juego() | true | - | - | - |  |
| 1 | ejecutar\_turno() | true | - | 2 | pos\_moneda\_uno = 4  pos\_moneda\_dos = pos\_moneda\_uno + 1  pos\_moneda\_dos = 5  *- se verifica que en ambas posiciones estén con monedas, si cumple.*  *- se sacan ambas monedas, la lista queda →.*  *- se verifican si es/son la/s ultima/s monedas.* |  |
| 2 | ejecutar\_turno() | false | - | 2 | pos\_moneda\_uno = 12  pos\_moneda\_dos = (pos\_moneda\_uno + 1) % 12  pos\_moneda\_dos = 1  *- se verifica que en ambas posiciones estén con monedas, si cumple.*  *- se sacan ambas monedas, la lista queda →*  *- se verifican si es/son la/s ultima/s monedas.* |  |
| 3 | ejecutar\_turno() | true | - | 1 | pos\_moneda = 2  *- se verifica que en esa posición esté una moneda, si cumple.*  *- se saca la moneda, la lista queda →.*  *- se verifican si es/son la/s ultima/s monedas.* |  |
| 4 | ejecutar\_turno() | false | - | 2 | pos\_moneda\_uno = 3  pos\_moneda\_dos = pos\_moneda\_uno + 1  pos\_moneda\_dos = 4  *- se verifica que en esa posición esté una moneda, NO cumple.*  *- no se saca nada, la moneda en la posición 4 ya fue sacada.*  *- el jugador repite su turno.* |  |
| 5 | ejecutar\_turno() | false | - | 2 | pos\_moneda\_uno = 10  pos\_moneda\_dos = pos\_moneda\_uno + 1  pos\_moneda\_dos = 11  *- se verifica que en ambas posiciones estén con monedas, si cumple.*  *- se sacan ambas monedas, la lista queda →.*  *- se verifican si es/son la/s ultima/s monedas.* |  |
| 6 | ejecutar\_turno() | true | - | 1 | pos\_moneda = 8  *- se verifica que en esa posición esté una moneda, si cumple.*  *- se saca la moneda, la lista queda →.*  *- se verifican si es/son la/s ultima/s monedas.* |  |
| 7 | ejecutar\_turno() | false | - | 2 | pos\_moneda\_uno = 6  pos\_moneda\_dos = pos\_moneda\_uno + 1  pos\_moneda\_dos = 7  *- se verifica que en ambas posiciones estén con monedas, si cumple.*  *- se sacan ambas monedas, la lista queda →.*  *- se verifican si es/son la/s ultima/s monedas.* |  |
| 8 | ejecutar\_turno() | true | - | 1 | pos\_moneda = 3  *- se verifica que en esa posición esté una moneda, si cumple.*  *- se saca la moneda, la lista queda →.*  *- se verifican si es/son la/s ultima/s monedas.* |  |
| 9 | ejecutar\_turno() | false | - | 1 | pos\_moneda = 9  *- se verifica que en esa posición esté una moneda, si cumple.*  *- se saca la moneda, y es la última moneda en sacarse, la lista queda →.*  *- se verifican si es/son la/s ultima/s monedas.* |  |
| - | Fue la ultima moneda sacada entonces: | - | **El ganador es el JUGADOR 2** | - | - | - |