

# Tarea 4

## IWI131 - Programación

Lunes 4 de Mayo de 2020, v1.1

### I. INTRODUCCIÓN

Un Casino de juegos ha detectado que a sus clientes les gusta jugar al “cara y sello”, es decir, lanzar monedas y ver cuales salen cara y cuales salen sello. El problema es que les gusta lanzar de a muchas de monedas, lo cual es complicado desde un punto de vista logístico. Usted que aprobó Programación y aprendió la función **randint()** le sugiere al Casino que utilicen un computador para los lanzamientos. La idea le gustó a la administración y le contrataron para que implementara el programa, eso si, el juego era bastante más complicado de lo que usted pensó en un inicio. El juego consiste en lo siguiente:

1. Una persona comienza con una cantidad de monedas y las lanza a la mesa
2. La persona toma todas las monedas que tienen “cara” y las retira del juego
3. Se lanza nuevamente todas las monedas que salieron “sello” y vuelve al punto anterior
4. Cuando no quedan monedas que hayan salido “sello” la persona anota la cantidad de lanzamientos que le tomó limpiar la mesa

### II. DESARROLLO

#### II-A. Lanzamiento de una moneda

Comience con una función llamada **es\_cara()** que en base a la función **randint()** retorne el valor lógico **True** (interpretado como cara) y **False** (interpretado como sello) cada vez que la función se ejecute. Por su naturaleza, esta función no requerirá parámetros.

Un ejemplo de ejecución sería el siguiente:

```
1 >>> print(es_cara())
2 True
```

#### II-B. Lanzador de monedas

Utilizando la función anterior, haga una función llamada **lanzamiento()** que reciba como parámetro una cantidad de monedas, lance las monedas y retorne el número de caras de la ronda.

Un ejemplo de ejecución sería el siguiente:

```
1 >>> monedas = 10
2 >>> print(lanzamiento(monedas))
3 5
```

#### II-C. Máquina de juego

Continúe con una función llamada **maquina()** que reciba como parámetro una cantidad de monedas a lanzar y calcule internamente las caras obtenidas y las monedas que quedan en juego para cada ronda. Al acabarse las monedas retorne el número de rondas que fue necesario para terminar el juego.

Un ejemplo de ejecución sería el siguiente:

```
1 >>> monedas = 10
2 >>> print(maquina(monedas))
3 5
4 >>> print(maquina(monedas))
5 6
```

#### II-D. Estimación de rondas para el Casino

El Casino entregará un premio a los jugadores que limpien la mesa luego de una cantidad de intentos igual o menor a una cantidad determinada por el Casino. Como al Casino no le interesa perder dinero, le ha pedido que haga una función llamada **intentos()** que calcule, en promedio, cuantas rondas se requiere para terminar el juego dependiendo de la cantidad de monedas. Para este calculo deberá proveer tanto el número de monedas como la cantidad de intentos a promediar.

Un ejemplo de ejecución sería el siguiente:

```
1 >>> monedas = 100
2 >>> pruebas = 1000
3 >>> print(intentos(monedas, pruebas))
4 8.057
5 >>> print(intentos(monedas, pruebas))
6 8.019
7 >>> monedas = 1000
8 >>> print(intentos(monedas, pruebas))
9 11.236
```

Junto con su programa entregue la cantidad de intentos promedio que se requiere para limpiar la mesa cuando se juega con 10, 100, 1000 y 10000 monedas.

### III. ENTREGA

Usted deberá entregar un único archivo “.py”. Al inicio del archivo debe colocar únicamente las 4 funciones solicitadas, respetando los nombres indicados para cada función e indicando a que sección de la tarea corresponden (II-A, II-B, etc.). Luego deberá colocar a lo menos un uso de cada una de las funciones para demostrar su funcionamiento, luego incluya las líneas que fueron necesarias para realizar los cálculos de la última parte de II-D. Al final de su archivo incluya un comentario informando la cantidad de intentos promedio que usted ha calculado con su programa de acuerdo a lo solicitado al final de II-D.