

Ejercicios de preparación



Miniproyecto 2

Se recomienda realizar estos ejercicios **antes** de desarrollar el miniproyecto 1, ya que les facilitará enormemente la tarea. Se sugiere además hacer los 3 ejercicios en el orden propuesto (1, 2 y 3).

En el miniproyecto van a utilizar lo aprendido con estos ejercicios.

Ejercicio 1

Escribe un programa que pida los nombres de dos personas (separados por un espacio en blanco) e imprima las primeras 3 letras de cada nombre en mayúsculas. Además, si resultan iguales agregarle un número 2 al segundo.

Ejemplo:

```
Ingrese los nombres de las personas: Matías Fernando
```

```
>> MAT FER
```

```
Ingrese los nombres de las personas: Fernando Fernanda
```

```
>> FER FER2
```

Ayuda

- a) La función `split()` permite separar un string de varias palabras dejándolo como un array de palabras. Es posible especificar el separador (en este caso es un espacio en blanco). Por ejemplo, si `s1 = "este es un ejemplo"`.

Entonces `s1.split(" ")` sería `['este', 'es', 'un', 'ejemplo']`.

- b) Dado un string de caracteres es posible extraer un segmento de él, especificando los subíndices de inicio y de final. Por ejemplo, si `x = 'ejemplo'`

Entonces `x[0:3]` sería `'eje'` y `x[1:4]` sería `'jem'`.

- c) La función `upper()` permite transformar un string en solo mayúsculas, entonces si `x = "pedro"`. Entonces `x.upper()` será `PEDRO`.

d) Para agregar caracteres al final de un string se puede usar el operador de concatenación +

Por ejemplo, si `s1 = 'Java'` y `s2 = 'Script'`, entonces `s1 + s2` es el string `'JavaScript'`.

Ejercicio 2

Escribe un programa que pida el resultado de un lanzamiento de dardos y genere el puntaje asociado al lanzamiento. El resultado ingresado puede ser uno de los siguientes:

- “SINGLE BULL”
- “DOUBLE BULL”
- “NULL”
- (número 1) (número 2)

Si se trata de "SINGLE BULL" el puntaje es 25, en "DOUBLE BULL" el puntaje es 50, mientras que "NULL" tiene puntaje 0. En el caso en que se ingresen dos números, el puntaje se calcula multiplicando el primer número por el segundo.

Ejemplo:

```
Ingrese jugada:15 4
>> 60
```

```
Ingrese jugada: SINGLE BULL
>> 25
```

```
Ingrese jugada: NULL
>> 0
```

Ayuda

- a) Puedes suponer que el input es correcto y que solo vienen los 4 formatos indicados, sin necesidad de validar.
- b) Recuerda que la función `split()` te permite separar los dos números en el cuarto caso. No olvides pasarlos por `int()` para convertirlos a números (ver ejercicios de miniproyecto 1).

Ejercicio 3

Escribe un programa que genere un número aleatorio entre 1 y 20 (ver ejercicios de miniproyecto 1) y que luego le pida al usuario un número que tiene el objetivo de reducirlo a cero. Por ejemplo, si el número generado es 15 y el usuario indica 10, el programa va a responder 5 ($15 - 10$). Pero ojo, si el número a restar lo deja en negativo se toma el valor absoluto. Es decir, si el número generado era 15 y el usuario ingresa 20, el programa responderá igualmente 5 ($\text{abs}(15 - 20) = 5$)

Ejemplo, si el número aleatorio inicial es 16.

```
Ingrese valor a restar: 10
>> Quedan 6
Ingrese valor a restar: 8
>> Quedan 2
Ingrese valor a restar: 2
>> Quedan 0
```

Ayuda

- a) Recuerda que la función que permite obtener el valor absoluto de un número es `abs()`. Si $x = -30$ entonces `abs(x)` será 30.
- b) Después de generar el número random, debes incluir toda la secuencia que sigue en un ciclo (loop). Este ciclo (por ejemplo `while`) incluye la captura del número que ingresa el usuario, el restarlo a lo que queda del número inicial e imprimir lo que queda. Este loop se termina cuando lo que queda es cero.