

# Tarea 11

Algoritmos Computacionales. Grupo 3009  
Facultad de Ciencias  
Universidad Nacional Autónoma de México

Fecha de entrega:  $\infty$

**Instrucciones:** resolver todos los ejercicios de aquí mostrados dentro de un Notebook de Jupyter. La separación entre ejercicios debe de ser clara. El uso de celdas de Markdown para poner texto explicativo y los comentarios se recomiendan ampliamente.

## Ingenio

### 1. Numeros Aleatorios

1. Una tirada de un dado justo se puede entender como tomar un elemento al azar el conjunto  $\{1, 2, 3, 4, 5, 6\}$ . Define una función `doble_dado()` que simula la tirada de dos dados y nos regresa la suma de los dos valores obtenidos. Usando tu función, simula  $10^4$  tiradas de dos dados y analiza su histograma. ¿Cuál es el valor más probable?
2. **Usando solamente un número aleatorio**  $r$  entre 0 y 1 y una estructura `If ... Else`, define una función `volado()` que regrese el string “sol” con probabilidad  $\frac{1}{2}$  y el string “aguila” con probabilidad  $\frac{1}{2}$

### 2. Manipular strings

3. Cuando se hace regresión lineal con variables categóricas en R, en la tabla de coeficientes una variable categórica se representa con un string de la forma `C(variable,OTHER=referencia)`.

Utilizando la función ‘split’, Define una función `limpiar_variable(cad)` que tome el string `cad` y nos regrese una lista `[variable,referencia]`.

Por ejemplo, `limpiar_variable( "C(city,OTHER=New_York) "`) debe regresar `city,New_York` y `limpiar_variable( "C(emotion,OTHER=sadness) "`) debe regresar `[emotion,sadness]`

4. Escribe una función `mi_corrector(frase)` que tome un string `frase` y cambie todas las ocurrencias de los errores en la tabla 1 por su respectiva corrección

Error	corrección
np	mp
nb	mb
mv	nv

Cuadro 1: tabla de errores y correcciones

### 3. Archivos de texto

5. Un archivo `.tab` es un archivo que guarda una tabla tal que distintas filas se separan con una línea nueva y distintas columnas se separan con el carácter `\t`. Define una función `mi_lectura(archivo)` que lea una tabla llamada `archivo.tab` y regrese una lista de listas `resultado` tal que el  $k$ -ésimo elemento de `resultado` es la lista con los elementos de la  $k$ -ésima fila.

Por ejemplo, el resultado para el archivo `prueba.tab`, anexo a esta tarea, debe de ser: `[[0,1,2],[7,8,9]]`