

## Tarea 1

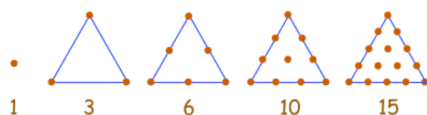
1. Haz un programa que te regrese los primeros 100 términos de la sucesión de Fibonacci (1, 1, 2, 3, 5, 8, 13, 21, 34, 55, ...)
2. Elabora un programa que te permita estimar el límite de la serie:

$$\sum_{n=1}^{\infty} \frac{1}{2^n}$$

3. Genera un programa, lo más corto posible que te regrese lo siguiente:

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
[-1.0, 0.0]
[-1.0, 0.0, 1.0]
[-1.0, 0.0, 1.0, 4.0]
[-1.0, 0.0, 1.0, 4.0, 9.0]
[-1.0, 0.0, 1.0, 4.0, 9.0, 16.0]
[-1.0, 0.0, 1.0, 4.0, 9.0, 16.0, 25.0]
[-1.0, 0.0, 1.0, 4.0, 9.0, 16.0, 25.0, 36.0]
[-1.0, 0.0, 1.0, 4.0, 9.0, 16.0, 25.0, 36.0, 49.0]
[-1.0, 0.0, 1.0, 4.0, 9.0, 16.0, 25.0, 36.0, 49.0, 64.0]
[-1.0, 0.0, 1.0, 4.0, 9.0, 16.0, 25.0, 36.0, 49.0, 64.0, 81.0]
```

4. Elabora un programa que genere una sucesión de números aleatorios en el intervalo [0.01,0.20], cuya suma sea menor que 1.50
5. La sucesión de números triangulares es 1, 3, 6, 10, 15, 21, 28, 36, 45, ... Esta sucesión es obtenida de acuerdo a la siguiente figura:



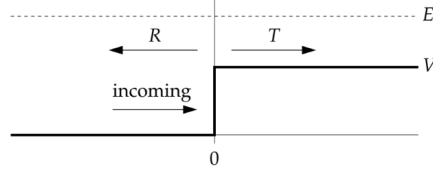
Escribe una función llamada `triangulares1`, que calcule los  $n$  primeros elementos de la sucesión. Usa esta función para calcular los primeros 10 valores.

### 6. Potencial escalón Cuántico

Un problema muy conocido en Mecánica Cuántica es el de una partícula de masa  $m$  que se encuentra con un potencial escalón unidimensional, como se ve en la figura:

La partícula tiene una energía cinética inicial  $E$  con un vector de onda  $k_1 = \sqrt{2mE}/\hbar$  que entra desde la izquierda y encuentra un potencial con energía  $V$  en la posición  $x = 0$ . Si resolvemos la ecuación de Schrödinger, se puede demostrar que hay dos opciones:

- (a) La partícula puede pasar el potencial, en cuyo caso, tiene una energía cinética más



baja de  $E - V$  en el otro lado y un vector de onda correspondiente más pequeño que  $k_2 = \sqrt{2m(E - V)}/\hbar$ .

(b) puede reflejarse, manteniendo toda su energía cinética y un vector de onda sin cambios pero en movimiento en la dirección opuesta.

Las probabilidades  $T$  y  $R$  para transmisión y reflexión son dada por

$$T = \frac{4k_1k_2}{(k_1 + k_2)^2}, \quad R = \left[ \frac{k_1 - k_2}{(k_1 + k_2)} \right]^2,$$

Supongamos que tenemos una partícula con masa igual a la masa del electrón  $m = 9.11 \times 10^{-31}$  kg y energía de 10 eV encontrando un potencial escalón de altura 9 eV. Escribe un programa para calcular e imprimir las probabilidades de transmisión y reflexión usando las fórmulas anteriores.

## 7. Algoritmo de fracciones continuas

Para encontrar la solución de ecuaciones cuadráticas con singularidades, es posible reescribir la función a evaluar en términos de una expansión Taylor. Esto se llama como fracciones continuas, que pueden verse como generalizaciones de una expansión de Taylor. Un enfoque posible de las fracciones continuas es la sustitución sucesiva. Vamos a ilustrar esto con un ejemplo simple:

$$x^2 + 4x - 1 = 0 \tag{1}$$

que puede reescribirse como:

$$x = \frac{1}{4 + x},$$

que a su vez podría representarse a través de un proceso iterativo de sustitución

$$x_{n+1} = \frac{1}{4 + x_n},$$

con  $x_0 = 0$ . Esto significa que si tenemos,

$$\begin{aligned} x_1 &= \frac{1}{4} \\ x_2 &= \frac{1}{4 + \frac{1}{4}} \\ x_3 &= \frac{1}{4 + \frac{1}{4 + \frac{1}{4}}} \end{aligned}$$

Esto puede ser reescrito de forma compacta como:

$$x_n = x_0 + \frac{a_1}{x_1 + \frac{a_2}{x_2 + \frac{a_3}{x_3 + \frac{a_4}{\dots}}}}$$

(i) Escribe un programa que resuelva la ecuación (1) con la implementación del algoritmo de fracción continua y resuelva iterativamente. La solución exacta es  $x = 0.23607$ , ¿Cuántas iteraciones necesitamos para tener 3 cifras decimales correctas?

(ii) Generaliza el código para que el usuario introduzca los coeficientes de la función y el número de precisión que desea obtener.

## 8. La fórmula semi-empírica de la masa

En física nuclear, la fórmula semi-empírica de la masa se usa para calcular la energía de enlace nuclear (binding nuclear energy <sup>1</sup>) de un átomo B con el núcleo de otro átomo con masa A y número atómico Z.

$$B = a_1 A - a_2 A^{2/3} - a_3 \frac{Z^2}{A^{1/3}} - a_4 \frac{(A - 2Z)^2}{A} + \frac{a_5}{A^{1/2}}$$

donde, las unidades están en MeV (millones de eV), las constantes son  $a_1 = 15.8$ ,  $a_2 = 18.3$ ,  $a_3 = 0.714$ ,  $a_4 = 23.2$  y

$$a_5 = \begin{cases} 0, & \text{Si } A \text{ es impar} \\ 12.0, & \text{Si } A \text{ y } Z \text{ son pares} \\ -12.0 & \text{Si } A \text{ es par y } Z \text{ es impar} \end{cases}$$

(i) Escribe un programa que le pida al usuario los valores de A y Z e imprima la energía de enlace que tendría que tener B. Usa  $A = 58$  y  $Z = 28$  que son los valores del níquel

(ii) Modifica tu programa para que también imprima la energía de enlace por núcleo. La definición de la energía de enlace por núcleo es  $B/A$ .

(iii) Ahora, modifica tu programa para que sólo tome el valor de Z y tome los valores de  $A = Z, \dots, 3Z$ , para encontrar la mayor energía de enlace por núcleo. Este es el núcleo más estable para ese número atómico. Que imprima la A que obtiene la mayor energía de enlace (el isótopo más estable).

## 9. Constante de Madelung

En Física de materia condensada se usa la constante de Madelung ( $M$ ) para obtener la energía potencial total que “siente” un átomo en un sólido. Esta constante depende de las cargas de los átomos vecinos y sus posiciones.

En este programa calcularemos  $M$  para un sólido cristalino formado por cloruro de Sodio (NaCl). El cristal tiene una estructura cristalina en forma de cubo (red cúbica) con la alternancia de los respectivos átomos. El Na tiene una carga positiva equivalente a la carga del electrón (+e), mientras el sodio tiene una carga de -e. Si etiquetamos cada posición de la red, las coordenadas, con los tres enteros (i,j,k), entonces los sodios estarán en las posiciones i+j+k par, mientras que los átomos de cloro estarán en i+j+k impar.

Sea un sodio en el origen (i=j=k=0). Si cada átomo está espaciado en la red por una distancia a, entonces la distancia al origen del átomo en posición (i,j,k) será:

$$\sqrt{(ia)^2 + (ja)^2 + (ka)^2} = a\sqrt{i^2 + j^2 + k^2} \quad (2)$$

y el potencial eléctrico creado por este átomo, evaluado en el origen sigue ley de Coulomb.

$$V(i, j, k) = \pm \frac{e}{4\pi\epsilon_0 a \sqrt{i^2 + j^2 + k^2}} \quad (3)$$

---

<sup>1</sup>La energía que se necesita para que dos núcleos se unan

El signo  $\pm$  depende del número de átomo que se trata, si es par será positivo, si es impar será negativo. El potencial total que sentirá el átomo Na central (el que está en el origen), será:

$$V_{total} = \sum_{i,j,k=-L}^L V(i,j,k) = \frac{e}{4\pi\epsilon_0 a} M \quad (4)$$

Donde  $M$  es nuestra aproximación a la constante de Madelung (la  $M$  real se obtiene cuando  $L \rightarrow \infty$ ). Escribe un programa en Python que calcule la constante  $M$ , prueba con 10, 20 y 200. ¿Cuánto tiempo se tarda tu código para cada caso?

#### 10. Tiro parabólico

Elabora un programa que calcule la altura máxima de un tiro vertical que inicia desde el suelo. La velocidad inicial es solicitada al usuario.

#### 11. Operaciones con vectores

Elabora un programa que calcule el producto  $A \cdot B \times C$  con  $A, B, C$  en  $\mathbb{R}^3$ , las componentes son solicitadas al usuario.

#### 12. Matriz aleatoria

Elabora un programa que almacene en un arreglo una matriz simétrica de  $n \times n$ . Las entradas de la matriz serán números aleatorios enteros en el intervalo  $[0,20]$ .

#### 13. Matriz de Vandermonde

Considera la matriz de Vandermonde, la cual se define como

$$A_{ij} = v_i^j \quad i = 0, 1, 2, \dots, n; \quad j = 0, 1, 2, \dots, n$$

Crea una función llamada Vandermonde(v) que reciba un vector  $\vec{v}$  y regrese la matriz  $A$  definida arriba.

#### 14. Matriz tridiagonal

Crea una función llamada tridiagonal que reciba un vector y regrese la matriz cuadrada

$$A = \begin{bmatrix} v_0 & v_1^2 & 0 & 0 & 0 & \dots & 0 \\ v_1^3 & v_1 & v_2^2 & 0 & 0 & \dots & 0 \\ 0 & v_2^3 & v_2 & v_3^2 & 0 & \dots & 0 \\ 0 & 0 & v_3^2 & v_3 & v_4^2 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & v_{n-1}^3 & v_{n-1} & v_n^2 \\ 0 & 0 & 0 & 0 & \dots & v_n^3 & v_n \end{bmatrix}$$

Prueba con varios vectores e incluye estos vectores en tu código.