

Fundamentos de Programación / Estructuras Computacional

Taller de Repaso Nro 2 - Algoritmos Básicos - 2024-1

Escribir un programa en Python o un diagrama en Flowgorithm para cada uno de los siguientes enunciados:

1. Construir un programa que lea un número entero mayor que 2 y devuelva como resultado el número primo de valor más cercano, en este caso menor o igual, al número leído.
2. Construir un programa que lea un número entero positivo n y después lea, uno a uno, n valores. El resultado que entregara el programa es la media de los números pares de entre los leídos, es decir, la suma de todos los valores pares dividida por n . (nota: sin usar listas)
3. La formula de Leibniz para calcular π :

$$\sum_{i=0}^{\infty} \frac{(-1)^i}{(2i+1)} = \frac{\pi}{4}$$

Con esta formula, que es una serie infinita, se puede hallar el valor de π . El verdadero potencial de ella está en poder lograr una aproximación bastante buena del valor de π usando solo operaciones matemáticas básicas y repeticiones. Para realizar esta aproximación se substituye el valor de infinito en la sumatoria por un valor n . Construir un programa que lea n y después genere los términos desde $i=0$ hasta $i < n$ y los almacene en un vector. Cuando ya esté generado, sumar los elementos y multiplicar por 4 para hallar el valor aproximado de π . Imprimir el resultado. (nota: se notará si quedó bien al ver el valor de π entregado, que debe acercarse más al valor de π entre más términos se usen)

4. Una aplicación sencilla de la multiplicación de matrices es el uso de transformaciones lineales, por ejemplo la rotación.

$$A = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}$$

donde θ es el ángulo de rotación en radianes.

o el estiramiento:

$$B = \begin{pmatrix} k_x & 0 \\ 0 & k_y \end{pmatrix}$$

Donde k_x y k_y son los factores de estiramiento en x y y respectivamente. Ambos factores deben ser mayores que 1.

Construir un programa que lea 4 parejas \mathbf{x}, \mathbf{y} (que representan los cuatros vertices de un cuadrilatero), así como valores para θ , k_x y k_y y calcule los valores

resultantes (\mathbf{x}' , \mathbf{y}') de aplicar la transformación compuesta a los puntos dados. Graficar los puntos iniciales y los punto finales.

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = (AB) \begin{pmatrix} x \\ y \end{pmatrix}$$

5. Construir un programa que halle los números enteros pares hasta un valor dado por el usuario n . Ese valor debe ser leído mediante una entrada.

6. Modificar el programa anterior para que muestre un pequeño menú de opciones, permitiendo elegir entre números pares o impares. Después de recibir la opción, hallar los números correspondientes, pares o impares, hasta el valor n .

7. Un número es un *cuadrado perfecto* si su raíz cuadrada es un número exacto (sin decimales). Por ejemplo, el 4 es un cuadrado perfecto (2^2), al igual que lo son el 36 (6^2) y el 3.500.641 (1871^2).

Todos los números que *no* son cuadrados perfectos pueden multiplicarse por otros para conseguir serlo. Por ejemplo, el número 8 no es un cuadrado perfecto, pero al multiplicarlo por 2 se obtiene el 16, que sí lo es.

Entradas del programa: La entrada comienza con un número que indica cuántos casos de prueba tendrán que procesarse. Cada caso de prueba consiste en un número mayor que 0 y menor que 2^{31} .

Salidas: Para cada caso de prueba, el programa escribirá por la salida estándar, en una línea independiente, el número más pequeño que al ser multiplicado por el número del caso de prueba da como resultado un cuadrado perfecto.

8. Construir un programa que reciba las componentes en \mathbf{x} y \mathbf{y} de un vector y calcule una proyección del mismo sobre un par de vectores unitarios al azar. El programa de permitir recibir más de un vector, pero uno a la vez. Para cada caso graficar el punto inicial y los puntos que representan las proyecciones.

9. Construir un programa que lea un número entero mayor que 2 y devuelva como resultado el número primo de valor más cercano, en este caso menor o igual, al número leído.

Aviso de autor: Los problemas fueron tomados de diversas fuentes, incluyendo <https://acceptaelreto.com/doc/credits.php>, una iniciativa de la Universidad Complutense de Madrid.