UNIVERSIDAD DE INVESTIGACIÓN TECNOLÓGICA YACHAY TECH
PRIMER SEMESTRE "C"

PROYECTO DE ALGORITMOS

Grupo:

- Keila Franchesca Gómez Acosta
- Liceth Patricia Minaya Gonzalez
- Airina Valentina Córdova Torres
- Lesley Sofia Tulcanazo Nuñez
- Lourdes Aracely Granja Alvear
- Emil Darío Vega Gualán

OBJETIVO

Resolver un problema matemático utilizando las herramientas y la teoría vista en clase de Algoritmos y Programación, siguiendo cada uno de los pasos para la construcción de una función en el software Matlab de tal forma que satisfaga al interesado.

INTRODUCCIÓN TEÓRICA:

Algoritmo

Un **algoritmo** es una secuencia de pasos para resolver un problema. Los pasos deben estar muy bien definidos, y tienen que describir sin ambigüedades cómo llegar desde el inicio hasta el final.

Componentes de un algoritmo

Conceptualmente, un algoritmo tiene tres componentes:

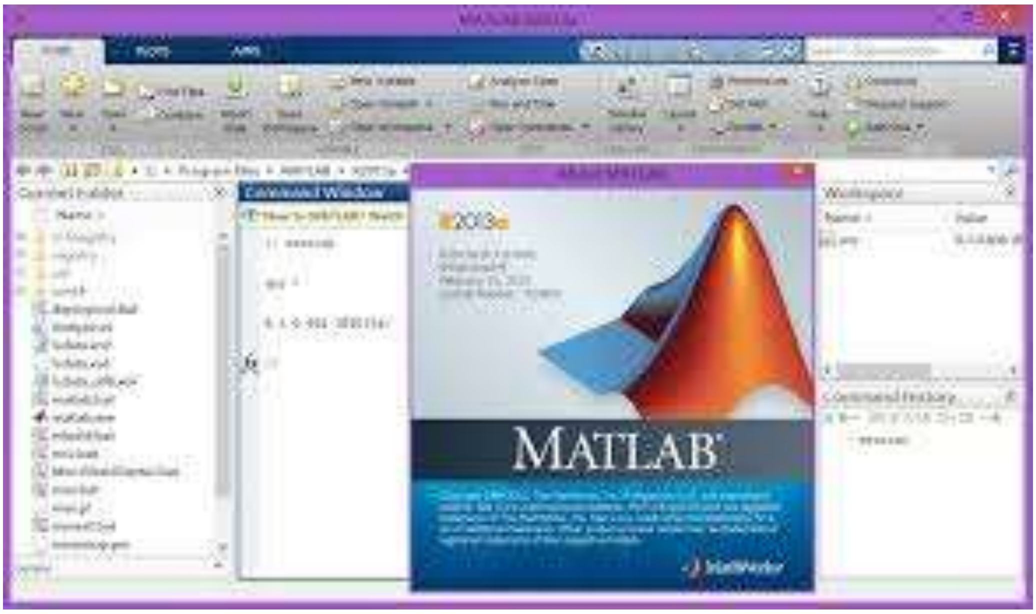
1. la **entrada**: son los datos sobre los que el algoritmo opera;
2. el **proceso**: son los pasos que hay que seguir, utilizando la entrada;
3. la **salida**: es el resultado que entrega el algoritmo.

El proceso es una secuencia de **sentencias**, que debe ser realizada en orden. El proceso también puede tener **ciclos** (grupos de sentencias que son ejecutadas varias veces) y **condicionales** (grupos de sentencias que sólo son ejecutadas bajo ciertas condiciones).

Matlab

|El nombre MATLAB viene de "matrix laboratory" (laboratorio matricial). MATLAB es un entorno de computación y desarrollo de aplicaciones totalmente integrado orientado para llevar a cabo proyectos en donde se encuentren implicados elevados cálculos matemáticos y la visualización gráfica de los mismos. MATLAB integra análisis numérico, cálculo matricial, proceso de señal y visualización gráfica en un entorno completo donde los problemas y sus soluciones son expresados del mismo modo en que se escribirían tradicionalmente, sin necesidad de hacer uso de la programación tradicional.

MATLAB nace como una solución a la necesidad de mejores y más poderosas herramientas de cálculo para resolver problemas de cálculo complejos en los que es necesario aprovecharlas las amplias capacidades de proceso de datos de grandes computadores, fue originalmente escrito para proveer acceso fácil al software matricial desarrollado por los proyectos LINPACK y EISPACK, que juntos representan el estado del arte e software para computación matricial. Hoy MATLAB es usado en una variedad de áreas de aplicación incluyendo procesamiento de señales e imágenes, diseño de sistemas de control, ingeniería financiera e investigación médica. La arquitectura abierta facilita usar MATLAB y los productos que lo acompañan para explorar datos y crear herramientas personalizadas que proveen visiones profundas tempranas y ventajas competitivas.



DESARROLLO

1. Planteamiento del problema

Considerar una cuadrícula 50x50

(a) Describir todos y cada uno de los posibles triángulos rectángulos sin incluir repeticiones, que se pueden formar con la condición de que: el punto (0; 0) sea uno de sus vértices, y los otros dos vértices tengan coordenadas enteras en [0; 50] _ [0; 50].

(b) Determinar cuántos y cuáles triángulos satisfacen que: la suma de los cuadrados de sus lados sean iguales. Representarlo gráficamente. (Cuales vs Cuantos).

2. Compresión del problema

Se deben dar ciertos pasos para llegar a la resolución del problema, estos se enlistan a continuación:

- Se debe comprender que es un triángulo rectángulo, mediante las distintas definiciones matemáticas que se han dado a través de la historia. Como lo es la definición de área, catetos, hipotenusa, ángulos entre otros. Principalmente el Teorema de Pitágoras.

- Entender el problema con cuadrículas más simples que una 50x50, como lo son la 1x1; 2x2 y 3x3 que tienen dimensiones más pequeñas y se encuentra dentro del alcance de cada estudiante.

- Comprender como se relaciona cada cuadrícula para generar un bucle de repetición que nos genere automáticamente todas las coordenadas de los posibles triángulos.

- Aplicar el concepto que anteriormente se adquirió, sobre el cálculo de la distancia entre dos puntos, que en este caso está dado en coordenadas 2D. Siendo esta una herramienta que ayuda en la comparación de distancia, arrojándonos la distancia mayor, que para un triángulo, como es conocido es la hipotenusa.

- Según la anterior acotación, se determina que la suma del cuadrado de los catetos debe ser igual a la hipotenusa, para poder definir que es un triángulo rectángulo.

- Con este cálculo, quedan totalmente descartados las líneas rectas que se forman o triángulos que no son rectángulos, por no cumplir la condición. Permitiendo trabajar con los que deseamos.

- Luego de poner las condiciones principales: 1). Debe ser un triángulo y uno de sus puntos obligatoriamente debe ser la coordenada (0,0); 2). Este triángulo debe ser triangulo rectángulo, es decir sus catetos deben formar un ángulo de 90°. Se puede pasar al punto ‘b’ del problema.

- Para resolver el literal ‘b’, es necesario que el literal ‘a’ del planteamiento del problema esté claramente desarrollado. Teniendo en cuenta, que el propósito es identificar cuantas veces se repite la longitud de los triángulos rectángulos que se van formando en la cuadricula de 50x50 y en que posiciones se encuentran.

- Desarrollando un poco más nuestra comprensión, será entonces que con la fórmula de la distancia se calcula la longitud de cada triangulo rectángulo que se forme, de tal manera que en la función de Matlab, se archiven dichas longitudes en un vector. **Así se responde ¿Cuántos triángulos cumplen que la suma de los cuadrados de sus lados sean iguales?**

- Después de obtener el vector con el número de triángulos rectángulos, se puede obtener una clasificación según ***cuales*** cumplen esa condición y así obtener la posición en la que se encuentran dichos triángulos que tienen la misma longitud. Por ejemplo: si al utilizar la fórmula de la distancia el algoritmo encuentra que hay triángulos rectángulos con la misma longitud = 4, entonces dará como una de sus respuestas las posiciones del vector en las que se repite el número 4, posición: 1,2, 10.

- Finalmente el algoritmo generará una matriz de ‘n’ filas y 6 columnas. Además un vector tamaño n x 1. Se imprimirán los triángulos rectángulos más representativos según nuestro criterio debido a la cifra es demasiado grande.

3. Análisis del problema

- Se tiene una cuadricula de 50 x 50 en la cual se van a dibujar los triángulos rectángulos en el cual uno de sus punto será el origen, es decir la coordenada (0,0).

- Se realizan pruebas de cuantos triángulos rectángulos se forman en una cuadrícula de 2x2 consiguiendo como resultados: 25 triángulos y 3 líneas rectas; de estos 25 triángulos dibujados a mano, solo 14 cumplieron la condición de ser *triángulos rectángulos.* Los restantes entonces deben ser descartados junto con las líneas rectas.

- La definición de un triángulo es *«polígono que tiene tres lados y tres ángulos »* (Romero) según la clasificación por sus ángulos, existe el triángulo rectángulo para el cual se cumple que *«uno de sus ángulos mide 90°»* (Romero).

- Teniendo la anterior definición, entonces debe de crearse una condición que permita ingresar los datos que se buscan. Así la utilización de la **fórmula 1**: $\sqrt{(Xf - Xi)^2 + (Yf - Yi)^2}$ para calcular la distancia entre puntos es útil.

- Se debe ingresar tres entradas ya que son tres los puntos que forman un triángulo, cada uno con coordenadas (X,Y), de la siguiente manera c1= (0,0), c2=(X2,Y2) y c3=(X3,Y3); siempre la coordenada c_0 será igual a (0,0) el origen.

- Se definen tres distancias como d1, d2 y d3. La d1 que se calcula con la fórmula 1, será la distancia entre c0 y c2; la d2 será la distancia entre c2 y c3; finalmente d3 será la distancia entre c3 y c0.

- Posteriormente se debe comparar las distancias y quedarnos con la mayor de las tres, ya que esa probablemente será la hipotenusa del triángulo rectángulo que según la definición es el lado más largo esta figura.

- Según el teorema de Pitágoras: el cuadrado de la hipotenusa es igual a la suma de los cuadrados de los otros dos lados (catetos). Por lo tanto para comprobar que es un **triángulo rectángulo** se debe obtener que la mayor de las tres distancias al cuadrado es obligatoriamente igual, a la suma de los cuadrados de las otras dos distancias. Por ejemplo: si d3 = 25cm d2= 16 y d1= 9; tendríamos que d3 es la distancia mayor de las tres, así que para que se cumpla el teorema de Pitágoras tendría que cumplirse: $3^2 + 4^2 = 5^2$. Sin embargo al hacer esta comparación, debido a que Matlab le cuesta mucho trabajo elevar al cuadrado una raíz, este genera un resultado con muchos decimales tanto para la hipotenusa como para la suma de sus catetos al cuadrado, por lo tanto debemos poner una tolerancia a la hora de hacer la comparación

- La tolerancia a considerar inicialmente será 0,01 y se probará para la cuadricula de 50x50, si los resultados aún no son los esperados, se disminuirá la tolerancia, para mejor la exactitud. Con esto podríamos concluir que el triángulo de este ejemplo si es rectángulo.

- Finalmente al aplicar este procedimiento con cada una las coordenadas que forman triángulos o líneas en la cuadricula de 50x50 podemos concluir cuales exactamente forman triángulos rectángulos, por ende descartar las figuras que no lo cumplan. Así se finaliza el literal 'a'.

- Para resolver el literal 'b', debemos calcular la suma de sus lados al cuadrado de cada triangulo rectángulo que se forma de la matriz de 50x50. Del teorema de Pitágoras se obtiene la hipotenusa, que sería el tercer lado del triángulo y los otros dos serían las otras dos distancias calculadas, con esto la expresión quedaría dada por la **fórmula 2:** Perímetro= $d1^2+d2^2+d3^2$

- Cada suma se tiene que almacenar en un vector de 'n' filas, para luego proceder con la segunda parte del aparto 'b', que consiste en identificar qué sumas son iguales y en qué posición se encuentran.

- Posteriormente identificar cuantas veces se repite el valor de la suma de sus lados al cuadrado de un triángulo. De esta manera conocer cuáles y las posiciones en donde se encuentran cada uno de estos.

4. Desarrollo de una solución

- En primer lugar se debe realizar una función que me permita generar la matriz resultado, ingresando el tamaño de la cuadricula que podemos llamar como variable "n". La función debe realizar:

    - Crear una serie de bucles de repetición mediante el cual irá recorriendo cada columna aumentando de uno en uno hasta "n". Así inicialmente se iría generando todas las posibles combinaciones de coordenadas.

    - Cada vez que generemos una coordenada debemos preguntar si los puntos de cada coordenada sean diferentes, si lo son se debe preguntar si es un triángulo rectángulo. Para lo cual creamos la función triangulo rectángulo:

        - Esta función calcula las distancias de cada punto, busca quien es la mayor para los puntos que resten sean los catetos.
        - Luego elevamos el mayor al cuadrado que será nuestra hipotenusa, y sumamos los catetos al cuadrado.
        - Para poder comparar restamos la hipotenusa de la suma de los catetos en valor absoluto y preguntamos si este es menor a nuestra tolerancia
        - Si es menor entonces decimos que si es un triángulo rectángulo.

    - Si resulta que es un triángulo rectángulo lo guardamos en nuestra matriz resultado.

- En segundo lugar creamos una función para obtener cuantos, cuáles y donde están los triángulos de la misma suma de sus lados al cuadrado. Inicialmente necesitamos un vector con las sumas respectivas, por lo tanto creamos otra función que me genere las sumas al cuadrado de cada lado:

    - Primero obtenemos el tamaño de la matriz resultado para guardar cuántas filas tiene y crear el vector de la misma dimensión.

    - Mediante un bucle de repetición recorro todas las filas de la matriz resultado calculando la distancia de sus puntos, sumándolas y guardándolas en un vector.

- Una vez obtenido el vector de las sumas de sus lados al cuadrado, obtengo el tamaño del vector para poder hacer el recorrido del mismo comparando cuáles y cuántas son iguales y donde están.

- Se crea bucle de repetición para capturar un dato del vector y mediante otro bucle anidado recorro hacia atrás del vector para tomar en cuenta si ya no lo conté antes, en caso de no ser así, continuo hacia delante con otro bucle comparando y guardando cuál se repite, cuántas veces se repite y las posiciones donde se encuentran estos números.

- Ordenamos este vector de manera ascendente tomando solo la columna de cuales se repiten con su respectiva cantidad de veces que se repite. Para esto utilizamos otra función:

    - Esta función recibe el vector a ordenar inicialmente, saca el tamaño del vector para poder hacer el recorrido, coloca un mayor y un menor y mediante un bucle recorremos el vector comprando quien es mayor y menor y reasignamos las salidas de mayor y menor.

- Por último se crea una función la cual será nuestra principal en donde llamaremos progresivamente las demás funciones según las necesitamos, también iremos generando la gráfica mediante un par de bucles.

BIBLIOGRAFÍA

Romero, J. C. (s.f.). Triángulos. *Triángulos*. Hidalgo, Mexico: Universidad Autónoma del Estado de Hidalgo.

Universidad de Jaen. (2006). *Área de Ingeniería de Sistemas y Automática.* Obtenido de http://www4.ujaen.es/~satorres/practicas/practica1.pdf