Estructuras de Datos
Alexandra Lecuna
Andres Imery

Proyecto Amazon

Clases

AddRoute: Interfaz que permite agregar rutas nuevas
AgregarAlmacen: Interfaz que permite agregar Almacenes
AgregarProductos: Interfaz que permite agregar Productos
AmazonProyecto: Clase Main, desde aquí corre todo el programa.

Archivo: Clase que permite la lectura y el guardado del archivo txt.

Métodos:
Abrir: Abre el archivo txt y retorna su contenido
Guardar: Recibe la información y la guarda en un archivo txt.

Arista: Clase que crea el objeto de las aristas para un grafo en forma del dibujo.

Sus atributos son:
x1: Guarda las coordenadas x de la primera arista.
y1: Guarda las coordenadas y de la primera arista.
x2: Guarda las coordenadas x de la segunda arista.
y2: Guarda las coordenadas y de la segunda arista.
Route: Guarda la ruta en String.
Distance: Guarda la distancia de la ruta.

Métodos:
paintLine: Dibuja una línea representando la arista de ambos Almacenes.
smallest: retorna la arista más pequeña de los dos dadas.
largest: retorna la arista más grande de las dos dadas.

Buy: Interfaz que muestra los productos disponibles y permite añadirlos para comprarlos, luego muestra un resumen de la compra.

Dijkstra: Clase que permite al recibir el grafo, el origen y el destino para calcular la ruta más corta entre los Almacenes.

Sus atributos son:

Peso: Matriz de destino del grafo.
Ultimo: Arreglo de todos los Almacenes.
D: Arreglo del tamaño de todos los almacenes
F: Arreglo del tamaño de todos los almacenes en Boolean.
Destination: int distancia hacia el destino
g: grafo.
m: Map
route: rutas arreglo de las rutas.

Metodos:

shortestRoute: calcula la ruta más corta.
Minumum: Calcula la ruta más mínima de las próximas.

FileChooser: Interfaz que permite elegir el archivo txt.

Grafo: Clase del grafo y permite al recibir la información del Map organizarla en forma de grafo.

Sus atributos son:

Warehouse: guarda los warehouse en un arreglo.
matrizA: La matriz para todos los Almacenes.
matrizD: Segunda matriz de todos los almacenes.
Vertice: vértice del grafo.
Map: Map
Route: arreglo de rutas.
numRoutes: número de rutas.
Routes: arreglo de las rutas

Metodos:

isEmpty: retorna si el grafo esta vacío.
addWarehouse: permite agregar un Warehouse al grafo.
addRoute: agrega una ruta al grafo.
CalcularRutasDeNodos: Permite calcular todas las rutas posibles entre Nodos.
searchWarehouse: permite buscar el Warehouse.
searchVertice: permite buscar el vértice.
listAllProducts: Hace un string con todos los productos.

listAllWarehouses: Hace un arreglo con los nombres de todos los Warehouses.
maxStockWarehouse: Busca el warehouse con mayor stock.

ItemCompra: Clase de la compra, almacena el resumen de toda la información de la compra.

Sus atributos son:
Warehouse: guarda el almacén.
Product: guarda el producto.
Quantity: Guarda la cantidad de productos.
Distance: distancia que recorrer en rutas para los productos.

Lista: Clase de la Lista.

Sus atributos son:
pFIrst: primer producto de la lista.
pLast: ultimo producto de la lista.
Size: Tamaño de la lista.
Metodos:
isEmpty: devuelve un booleano si la lista esta vacia.
findProduct: consigue el producto en la lista.
insertFinal: inserta un producto nuevo al final de la lista.
printList: imprime la lista en consola.
pickProducts: elige los productos de un warehouse y los reserva en una lista.
buyProducts: permite comprar los productos y remover del stock y warehouse.

Map: Crea la clase map, con algunas modificaciones para manejar la información luego en el grafo.

Principal: Interfaz principal de la aplicación.

Product: Clase que almacena información de los productos. Funciona como Nodo.

Name: String y nombre del producto.
Stock: int y cantidad de producto disponible.
pNext: próximo producto de la lista

Metodos:
reserveProducts: remueve los productos de un warehouse y los retorna en una lista.
Buy: permite comprar el producto si está disponible en la lista.

Route: Guarda la información de la ruta.

Sus atributos son:
Origin: Warehouse (Almacén) de donde empieza la ruta.
Destination: Warehouse (Almacen) de donde termina la ruta.
Distance: int, distancia de la ruta.

Stock: Interfaz que permite buscar en un almacén, varios o un producto.

Vertice: Crea el dibujo del grafo en forma de los almacenes y las rutas.

Sus atributos son:
X: posición x del vértice
Y: posición y del vértice.
Name: Nombre del vértice.
Color: color del vértice.
Metodos:
warehouseDibujo: Agrega el vértice al dibujo.

Warehouse: Clase para crear los Almacenes.

Sus atributos son:
Name: String nombre del Almacén.
CodigoC:
Products: Lista de los productos que tiene el almacén.
Metodos:
findProduct: busca el producto en el Almacen y lo devuelve.