 Proyecto CUPi2	ISIS-1205 Algorítmica y Programación 2 Descripción
Ejercicio:	
Autor:	
Fecha:	

Enunciado

Se desea desarrollar un juego en el cual ha ocurrido un crimen y es necesario que el usuario persiga al ladrón por varias ciudades del mundo, basándose en pistas que se encuentran en diferentes lugares de las ciudades visitadas.

Para capturar al ladrón, se tiene que tener en cuenta el tiempo. Hay un límite de tiempo dado en días y en caso que el tiempo especificado en el juego se acabe, el ladrón habrá escapado. Así, cada vez que se consulte un lugar de la ciudad, va a costar tiempo (especificado para cada lugar). Adicionalmente, cada vez que se quiere viajar de una ciudad a otra también va a costar tiempo.

Se tiene una lista de sospechosos, de los cuales sólo uno de ellos es el ladrón. Para poder atrapar al ladrón es necesario hacer dos cosas:

1. Encontrar la ciudad donde está el sospechoso siguiendo las pistas dadas.
2. Generar la orden de captura del sospechoso de acuerdo a las pistas que se encuentran en los lugares, antes de haber encontrado la ciudad.

En caso que el tiempo dado acabe, o se llegue a una ciudad de la cual no se puede viajar a otra, o se encuentre el sospechoso pero no se genere la orden de captura, el juego termina y el usuario pierde.

Para ganar en el juego, el usuario debe haber generado la orden de captura correctamente y encontrar la ciudad donde se encuentra el ladrón.

En caso que se genere una orden de captura para un sospechoso incorrecto el juego termina y el usuario pierde.

Ya que se piensa utilizar un árbol para estructurar la aplicación, se pretende conocer cierta información acerca de él:

- Peso
- Altura
- Recorrido Preorden
- Recorrido Postorden
- Recorrido Inorden

Interfaz

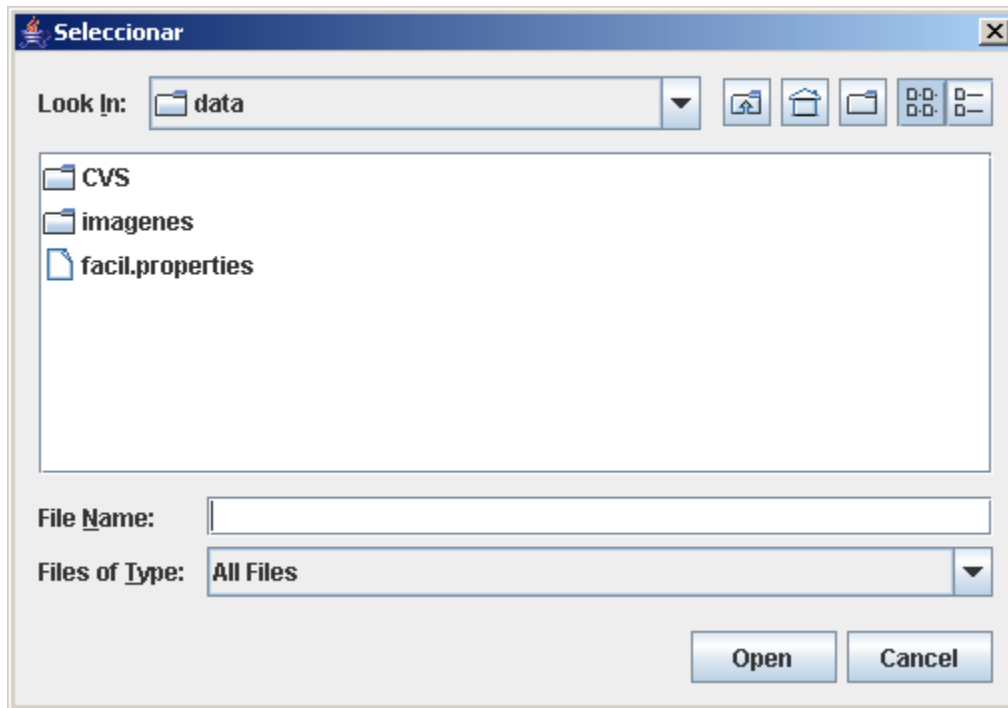


Figura 1 - Es posible seleccionar un archivo de juego



Figura 2 - Se muestra la ciudad donde todo comienza.



Figura 3 - Se muestran los lugares donde se pueden buscar pistas



Figura 4 - Se muestran los posibles destinos para viajar

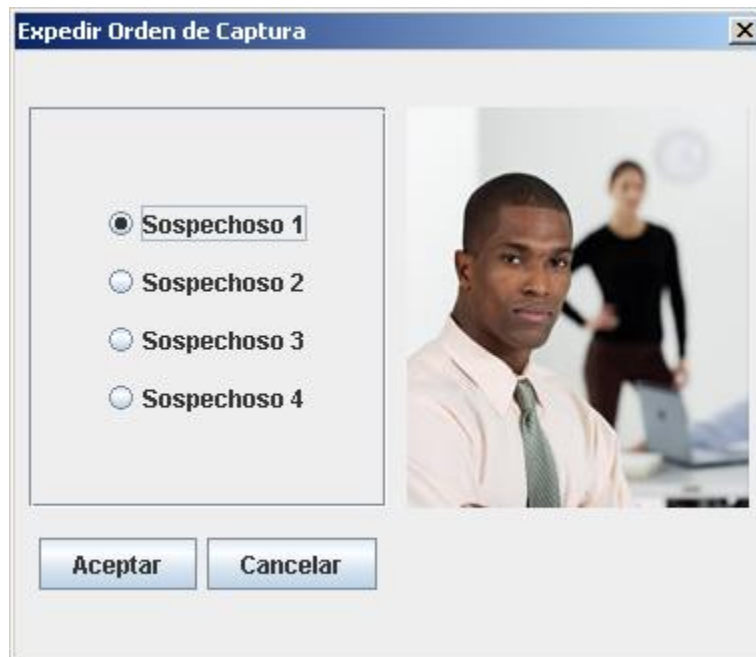


Figura 5- Se tiene una lista de sospechosos posibles

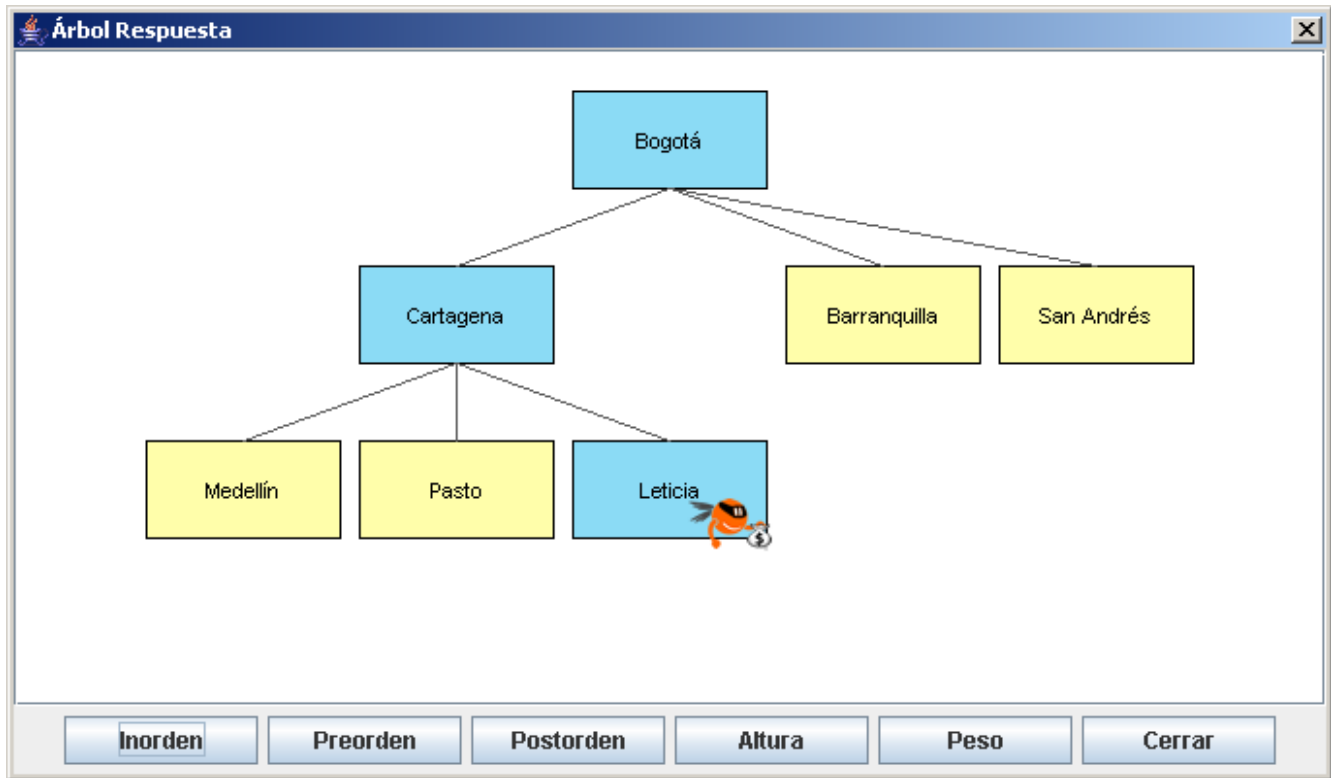


Figura 6 – Imagen del árbol de juego que se ve al finalizar



Figura 7 - Se muestra el recorrido en inorden



Figura 8 - Se muestra el recorrido en preorden



Figura 9- Se muestra el recorrido en postorden

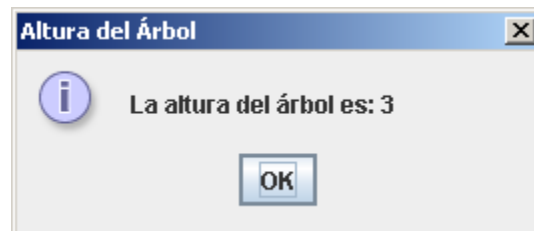


Figura 10 - Se muestra la altura del árbol

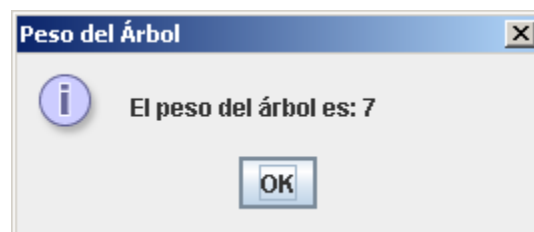


Figura 11 - Se muestra el peso del árbol

Descripción del Problema

La aplicación comienza con una interfaz vacía. El usuario debe seleccionar un archivo de propiedades que contenga un juego válido.

Una vez el usuario ha seleccionado un juego, se muestra el caso en el que se piensa trabajar. Cada juego tiene un tiempo máximo en el cual se puede desarrollar. Cuando ese tiempo se acaba, el juego termina.

El usuario puede seleccionar ver las pistas del juego en cierta cantidad de lugares que tiene la ciudad. Cada vez que el usuario decida ver una de estas pistas se le reducirá el tiempo de juego.

El usuario puede seleccionar viajar a otra ciudad persiguiendo al sospechoso. El viaje también tiene un costo en el tiempo. El usuario debe viajar a la ciudad donde se encuentra el sospechoso.

Para poder ganar en el juego, el usuario debe encontrar la ciudad donde se encuentra el sospechoso y debe generarse previamente una orden de captura.

Una orden de captura debe ser generada cuando se descubra cuál es el sospechoso de acuerdo a las pistas que se leen en cada lugar de la ciudad.

Diseño

La manera como se decidió modelar el problema fue utilizando un árbol 3-ario, donde en sus nodos se encuentran las ciudades donde el usuario puede viajar y donde puede o no encontrarse el sospechoso. Cada nodo debe tener asociado una lista de lugares donde se colocan las pistas.

Estructura de la Solución

Descripción de las Clases

Nombre:	Interpol
Descripción:	Representa la clase que tendrá el estado del juego.
Asociaciones	
Ciudad	Tendrá siempre la raíz del árbol, la ciudad que está en juego.
Sospechoso	Tendrá la lista de los sospechosos
Responsabilidades:	Mantener el estado del juego. Debe servir de puente entre la interfaz gráfica y el resto del mundo.



Nombre:	Ciudad
Descripción:	Clase que representa un nodo del árbol
Asociaciones	
Ciudad	Las ciudades donde se puede viajar en busca del sospechoso (3)
Lugar	Una lista de los lugares en la ciudad donde se pueden averiguar pistas.
Responsabilidades:	<p>Cuando la clase representa un nodo:</p> <ul style="list-style-type: none"> ○ Debe contener las diferentes ciudades donde puede viajar el usuario. <p>Cuando la clase represente una hoja:</p> <ul style="list-style-type: none"> ○ Debe saber si el ladrón se encuentra en la ciudad o no

Nombre:	Lugar
Descripción:	Clase que representa un lugar en una ciudad donde hay una pista
Asociaciones	
Responsabilidades:	Mantener la descripción de un lugar, el nombre de la pista y el tiempo que gasta ver el lugar

Nombre:	Sospechoso
Descripción:	Clase que representa un sospechoso
Asociaciones	
Responsabilidades:	Mantener el nombre del sospechoso y saber si él es ladrón o no