

ICOM2021 – 1er Parcial

11 de setiembre de 2021

Notas:

1. Uso de prácticos: **se pueden utilizar los trabajos prácticos propios realizados.**

Problema 1: Juego de las coincidencias

Es un juego de cartas cuyo mazo cuenta con 55 cartas, cada una de ellas con 8 figuras diferentes. En este juego, cualquier par de cartas que se elijan del mazo, tienen un solo dibujo en común.

La idea es hacer un programa que asista a los desarrolladores y que permita validar el mazo para asegurar que tenga este comportamiento.

A partir de un conjunto de cartas cuyos dibujos están representados por caracteres alfanuméricos, por ejemplo “a2**3**4d56f” o “1**3**egh789”, se solicita, utilizando el código en el archivo `coincidencias.cpp`, implementar las funciones marcadas con `// TODO`.



Problema 2: Scanner de resonancias magnéticas

Es una técnica de diagnóstico por imagen en la que se obtiene información de las características de diferentes puntos del cuerpo a partir de su respuesta a un campo magnético variable.

El cuerpo “escaneado” puede verse como discretizado tridimensionalmente en celdas, en donde cada celda posee su valor resonante representado por un valor entero sin signo (valores entre 0 y 255).

Dentro del análisis que se realiza sobre una imagen de un cuerpo escaneado hay 2 características buscadas que son fundamentales para centrar el estudio:

- Detección de un “punto sospechoso”: dado la imagen del cuerpo y un punto a través de su coordenada x, y, z detecte si es sospechoso, considerando sospechoso aquellos puntos para los que todos los puntos adyacentes directos (propio plano y plano superior e inferior) tienen valores dentro de un intervalo de nivel de resonancia dado.
- Detección de “plano sospechoso”: Es aquel plano (en Z) que posee una proporción de puntos sospechosos mayor a un valor determinado.

Se cuenta con un diseño e implementación parcial de un UDT **ScanAnalyzer** (`Scanner.cpp`) para facilitar la detección de planos sospechosos.

Complete la implementación del tipo dado implementando los métodos marcados con `// TODO` y pruebe con el main incluido.

Problema 3: Saltarín

El juego del saltarín consta de un tablero de 10x10 casillas. En cada una de esas casillas, hay un número natural {1, 2, 3, ...}. El objetivo del juego es comenzar en la casilla {0,0} y llegar a la opuesta {9,9}. Los movimientos posibles son en las 4 direcciones {ARRIBA, ABAJO, IZQUIERDA, DERECHA}, y el número en cada casilla indica la distancia del salto. Los movimientos que caen fuera del tablero están prohibidos. Dependiendo de la combinación de números en el tablero, puede haber más de una solución posible, o ninguna.

En el archivo `saltarín.cpp` encontrará un esqueleto de la solución. Deberá completar los métodos marcados como `TODO`. Los archivos `consalida.txt` y `sinsalida.txt` contienen los saltos de dos tableros que utiliza el main para probar el algoritmo. El archivo `saltarín.jpg` es una captura de pantalla de una posible salida del programa.