

II Prueba de Ejecución Estructuras de Datos EIF-207

Escuela de Informática
Facultad de Ciencias Exactas y Naturales
Universidad Nacional

La prueba está diseñada para comprobar la capacidad del estudiante de adaptar los conceptos aprendidos para generar una solución a los problemas nuevos planteados en los ejercicios que conforman el examen.

Usted debe trabajar de forma ordenada, y crear 1 carpeta/folder por cada pregunta con un nombre descriptivo. Una vez que haya finalizado su trabajo debe hacer entrega de este en el aula virtual en el espacio establecido para tales efectos. No se aceptan entregas tardías ni por otros medios. Es responsabilidad del estudiante hacer entrega del resultado final de su trabajo. Recuerde que el aula virtual tiene un máximo tamaño para los archivos a ser entregados por lo que si su solución excede este tamaño el proceso de entrega falla.

Puntos totales 100 de la prueba

La prueba cuenta por un 20% de la nota final del curso.

Enunciado del Problema Planteado

La empresa **NetPro** especializada en telecomunicaciones, está diseñando la red de computadoras de uno de sus principales clientes para lo cual hizo una visita reciente a sus oficinas y encontró que desea conectar las computadoras etiquetados como: A, B, C, D, E, F, G, H, I

Es importante entender que las computadoras están distribuidas a lo largo de un edificio de múltiples pisos, y en cada piso puede haber separaciones de paredes de concreto, de gypsum y algunas otras limitaciones adicionales. Dadas las limitaciones que plantea cada uno de los ítems antes mencionados, el arquitecto de la red ha definidos la siguiente tabla dando los pesos a cada uno de ellos:

ITEM	PESO
CONCRETO	2,5
GYPSUM	1,5
PISO	2
OTROS	0,5

La siguiente es la distribución de las limitaciones que tiene cada uno de los posibles puntos de conexión entre las computadoras que se desea agregar a la red. Vea el siguiente ejemplo de conectar las computadoras G e I, en este caso se debe pasar una pared de Gypsum y además esa conexión tiene una limitación adicional por lo que el peso total de esa conexión es de 2, dado que $1,5 + 0,5 = 2$.

Como parte del trabajo a realizar usted debe de generar los pesos finales de cada una de las conexiones entre máquinas lo que definirá finalmente cuánto cuesta el camino de conexión entre dos computadoras.

MAQUINA 1	MAQUINA 2	CONCRETO	GYP SUM	CAMBIO DE PISO	ADICIONAL
A	B	2	1	1	3
A	C	3	1	1	2
B	C	1	0	2	5
B	D	2	0	1	2
C	E	0	1	0	3
C	F	0	0	0	2
D	E	2	1	0	1
D	F	3	1	0	4
D	H	2	0	0	0
D	G	0	4	1	0
E	F	0	2	0	0
F	H	1	2	0	1
G	H	3	0	0	3
G	I	0	1	0	1
H	I	1	3	2	0

Usted debe Generar con la combinación de las dos tablas anteriores un archivo de texto con los datos del vértice1, vértice2 y el peso de dicha conexión, todo separado por comas, ya que el sistema debe permitir leer dicho archivo y cargar los datos correspondientes para que sea simple la construcción del grafo en las estructuras de datos internas.

Con esta información usted ha sido comisionado para crear una aplicación simple que pueda cargar los datos antes mencionados, lo cual representa un grafo ponderado, a partir del cual usted debe crear un menú simple como el siguiente para responder a las necesidades que ha planteado el arquitecto de red, con lo cual puede tomar una decisión adecuada de cómo hacer las conexiones.

Cree una aplicación para manejar el siguiente menú

1. **Cargar datos automáticos:** esta opción del menú carga los datos del archivo de texto separado por comas.
2. **Dijkstra:** Esta opción muestra el camino más corto definido a partir del nodo seleccionado como inicio.
3. **Kruskal:** Esta opción muestra el árbol de expansión de costo mínimo generado mediante el algoritmo de Kruskal.

Junto con la solución planteada usted debe entregar un dibujo que es la representación gráfica de cada 1 de los algoritmos desarrollados

Rúbrica de calificación

ITEM	PUNTUACIÓN
CÁLCULOS PARA GENERAR EL ARCHIVO	10
CARGA DE DATOS DEL CSV	10
ESTRUCTURA BÁSICA NECESARIA	10
ALGORITMO DE DIJKSTRA	25
ALGORITMO DE KRUSKAL	25
LIMPIEZA APROPIADA DE LA MEMORIA (DESTRUCTOR)	10
DIBUJO DEL RESULTADO DE CADA ALGORITMO	10 (5 C/U)
TOTAL	100

Hash Tables - Ejercicio Optativo 10 PUNTOS ADICIONALES

Construya una solución para aplicar hash tables donde la tabla maneje valores genéricos mediante el uso de templates. Esta solución debe estar capacitada para recibir tipos distintos de datos de las primitivas convencionales como int, float, double y string.

Asegúrese que la función hashCode sea modificada acorde con el tipo de dato que se selecciona. Recuerde que hay métodos distintos para definir el hash code en valores como float, y double, para considerar apropiadamente los valores que se salen de rango o que son parte del punto flotante. Del mismo modo hay un algoritmo visto en clase para generar ese mismo hash code para valores de tipo string. Para solucionar esto necesita hacer una especialización de los templates para asegurarse que los métodos correspondientes se apliquen apropiadamente. Para el manejo de colisiones debe implementar la fórmula para la Exploración Cuadrática.

Debe tener un menú como el siguiente:

1. **Definir el tipo de dato:** esta opción me permite seleccionar si quiero almacenar valores int, float, double o string
2. **Generar valores aleatorios:** esta opción del menú genera 20 valores aleatorios del tipo de dato seleccionados y los introduce a la tabla hash
3. **Mostrar datos:** esta opción del menú permite mostrar todos los datos insertados en la opción 2.
4. **Salir:** esta opción se asegura que se limpia la memoria y cierra el sistema.