

Objetivos

Unidad 4: Grafos

- OE4.1. Explicar los conceptos básicos sobre la teoría de grafos.
- OE4.2. Modelar la información de un problema utilizando un grafo como estructura de datos.
- OE4.3. Aplicar los recorridos en profundidad y por niveles de los grafos en el contexto de un problema dado.
- OE4.4. Aplicar los algoritmos de Dijkstra y Floyd-Warshall para resolver problemas de búsqueda de caminos más cortos en el contexto de un problema dado.
- OE4.5. Aplicar los algoritmos de Prim y Kruskal para resolver problemas de árboles de recubrimiento mínimo en el contexto de un problema dado.
- OE4.6. Diseñar y construir un grafo representado por matrices de adyacencias y listas de adyacencias.
- OE4.7. Implementar los algoritmos de recorridos sobre grafos y búsqueda de caminos más cortos.
- OE4.8. Diseñar y construir las pruebas unitarias de cada uno de los grafos implementados.

Enunciado

Usted debe desarrollar (analizar, diseñar e implementar) un juego de uno o más jugadores, el cual pueda ser modelado utilizando grafos, con un **mínimo de 50 vértices y 50 aristas**, y tal que para su solución sea posible aplicar al menos dos (2) de los algoritmos de grafos que se estudiarán durante el curso: Recorridos sobre Grafos (BFS, DFS), Caminos de Peso Mínimo (Dijkstra, Floyd-Warshall), Árbol de Recubrimiento Mínimo -MST- (Prim, Kruskal).

El juego debe ser definido por usted y su grupo de máximo 3 personas, incluyéndole. Los requerimientos mínimos son los siguientes:

- Desarrollar 2 versiones de Grafo (su solución debe funcionar sin problema con las dos versiones, es decir, el programa debe admitir el cambio de la implementación utilizada en cualquier momento y funcionar bien indistintamente de la que se esté usando). Cada grafo debe ser desarrollado desde el TAD hasta las pruebas unitarias automáticas.
- Llevar a cabo y documentar cada una de las fases del método de la ingeniería para la solución del problema planteado.
- Documentar apropiadamente las fases de análisis y diseño con el documento de especificación de requerimientos, el diseño del TAD, diagramas de clase y objetos, y el diseño de los casos de pruebas de las pruebas unitarias automáticas.
- Su programa debe contar con una interfaz gráfica de usuario que permita utilizar las funcionalidades que respondan a los requerimientos del problema.

Entrega preliminar. Domingo 5 de noviembre de 2023. Hasta las 23:59 a través del enlace dispuesto en Intu. Poner un tag en el último commit de dicha entrega.

La estructura Grafo completamente analizada, diseñada, implementada y probada, más una explicación detallada del problema a solucionar. Su entrega debe incluir:

1. Informe del seguimiento del Método de la ingeniería sobre el problema que están abordando.
2. Especificación del TAD Grafo. Nombre, representación, invariante, operaciones y la especificación de cada una de las operaciones en términos de entrada y salida (tal como se revisó en la Unidad 2 del curso).
3. El problema a solucionar:
 - a. Enunciado suficientemente claro de **la situación problemática** que solucionarán.
 - b. Especificación de Requerimientos Funcionales del programa que darán solución al problema.



UNIVERSIDAD
ICESI

Departamento de TIC
Computación y Estructuras Discretas I
2023-2 - Tarea Integradora 2

Entrega Final. Jueves 23 de noviembre de 2023. Poner

un tag en el último commit de dicha entrega.

1. Diseño de Diagrama de Clases del TAD Grafo incluyendo las dos implementaciones y las mejores prácticas de diseño (no olvidar desacoplamiento y generics).
2. Diseño de pruebas unitarias:
 - a. De las operaciones estructurales del grafo (agregar, eliminar y consultar).
 - b. De los algoritmos vistos en clase.
3. Implementación completa de:
 - a. El código que da solución al problema que escogieron.
 - b. Los grafos incluyendo los algoritmos vistos en clase.
 - c. Las pruebas unitarias automáticas diseñadas sobre los grafos.

Esta es la [rúbrica](#) de evaluación de la entrega final de la TI2.