

Tarea Programada #1 ***(Grupal)***

Instrucciones:

1. La tarea debe entregarse al profesor por medio del TEC Digital en la fecha y hora establecida.
2. **El nombre de la tarea debe indicar: TAREA_PROGRAMADA#1**
3. La tarea debe ser programada en Python con una interfaz gráfica.
4. Toda tarea debe ser defendida ante el profesor, de tal manera todos los estudiantes deben poder explicar la solución satisfactoriamente.
5. Cada uno de los algoritmos programados debe ser ejecutado al menos dos veces con datos de entrada diferente para ver su tiempo de ejecución de acuerdo con la entrada. Los resultados obtenidos deben ser documentados y analizados de acuerdo con la teoría de vista en clase sobre tiempos de ejecución.
6. En el caso de los algoritmos de **Dijkstra y Floyd** se deben realizar corridas de ambos programas con la **misma entrada** y analizar su comportamiento.
7. En el caso de la las Torres de Hanoi se requiere realizar corridas para N muy grandes y documentar los tiempos de ejecución
8. La tarea debe contener lo siguiente:
 - a. Fuentes. Todo el código necesario para ejecutar la tarea. El código debe estar debidamente documentado.
 - b. Documentación, incluyendo al menos:
 - i. Análisis detallado de los tiempos de ejecución en forma teórica para cada uno de los algoritmos tomando en cuenta la teoría vista en clases en cuanto la notación O, peor caso, mejor caso, tiempos polinomiales P, NP y los problemas intratables.
 - ii. Documentación en el código con una descripción de lo que hace cada programa.
 - iii. Explicación de los resultados obtenidos en cuanto a tiempos de ejecución según entradas de datos diferentes.
 - iv. Descripción técnica (algoritmos, interfaces, estructuras, etc..).

Descripción de la tarea

Este proyecto consiste en programar varios algoritmos que ilustren la técnica de Programación Dinámica tomando en cuenta la teoría vista en clases en cuanto al principio de optimalidad y los tiempos de ejecución de cada programa según los tamaños de la entrada y la técnica utilizada en el diseño del algoritmo. Debe aclarar que hay una parte del proyecto que no es de Programación Dinámica pero que son casos idóneos para analizar el tema de los tiempos de ejecución.

Los problemas a resolver mediante programación son los siguientes:

1. El algoritmo conocido como dar el vuelto con monedas, el cual consiste en que dada una cierta cantidad y un conjunto de N monedas de diferentes denominaciones se pueda dar el vuelto de esa cantidad con un número mínimo de monedas.
2. El problema de la mochila el cual consiste en dadas varias mochilas con cierta capacidad de peso y un conjunto objetos cada uno con un peso y un valor diferente, se pueda llenar la mochila con el máximo valor según los objetos que satisfagan la restricción del peso de la mochila.
3. El problema de los caminos mínimos que consiste en encontrar un camino entre dos vértices (o nodos) de tal manera que la suma de los pesos de las aristas que lo constituyen es mínima. Los algoritmos a programar son los de Floyd y Dijkstra con los cuales se debe establecer una comparación y análisis de su comportamiento y el tiempo de ejecución para cantidades grandes de nodos.
4. El algoritmo de las Torres de Hanoi en el cual se tienen tres agujas y N discos que al inicio están colocados en la aguja conocida como A apilados de tal forma que el disco más grande está en el fondo y el más pequeño en la cima sin que exista ninguna violación en este orden de acuerdo al tamaño. El problema consiste en pasar todos los discos de la aguja A a una tercera aguja llamada C utilizando la aguja B como paso intermedio de tal forma que en cada movimiento se haga con un disco a la vez y sin violar el orden de los discos (un disco menor no puede quedar por debajo de uno mayor). Se deben realizar corridas para N discos y analizar los tiempos de ejecución cuando N es grande.
5. Los algoritmos de ordenamiento Quicksort y HeapSort los cuales consisten en ordenar N elementos en orden ascendente. En este caso se deben realizar varias corridas con tamaños de N muy grandes y documentar su comportamiento. Se deben correr ambos algoritmos con una misma entrada para poder establecer una comparación de los resultados en cuanto a su tiempo de ejecución.
6. El problema de la multiplicación de matrices en cadena el cual consiste en multiplicar N matrices en cadena de tal forma que se utilice una técnica de programación dinámica para que el resultado sea el más óptimo. Se debe documentar los tiempos de ejecución para distintos tamaños de N y para matrices de tamaños muy grandes.

Preguntas.

- 1) ¿Funciona el algoritmo de Floyd en un grafo que tenga algunas aristas cuyas longitudes sean negativas pero que no contengan ningún ciclo negativo? Cuál es la relación con los problemas clase P, NP o NO completos. Justifique.
- 2) Bajo que circunstancias el algoritmo de Floyd es mejor a al de Dijkstra. Justifique.
- 3) ¿Cuáles serían los tiempos de respuesta estimados para el algoritmo de las Torres de Hanoi con $N = 50$ discos y corriendo en una máquina que procesa diez mil millones de instrucciones por segundo? Justifique.