



Taller: Grafo y árboles

Matemáticas discretas II

Carlos Andres Delgado S, Ing
`carlos.andres.delgado@correounivalle.edu.co`

Noviembre de 2021

Reglas del Taller

Importante: El no cumplimiento de alguna de las normas aquí expuestas le traerá reducción en la nota o la anulación de su taller.

1. El taller debe ser entregado antes del día **Domingo, 12 de Diciembre a las 11:59:59pm** hora de Colombia del por el enlace dispuesto en el campus virtual. Se permiten entregas tardías, pero se descuenta 0.15 en la nota por hora o fracción de retraso. Por ejemplo, si entrega a partir de las 12:00:01am se aplicará una penalización de 0.15, si lo entrega a partir de las 01:00:01am se aplicará 0.3 y así sucesivamente.
2. El taller se entrega por un enlace del campus virtual, el cual únicamente aceptará el enlace del Google Colab, asegúrese que el docente lo pueda ver sin problemas.
3. Entregue el enlace de Google Colab que tiene la implementación del proyecto.
4. No se permite copiar código de Internet ni de sus compañeros. Si se encuentra código copiado el proyecto será anulado por completo a todas las partes involucradas.
5. **Importante:** El lenguaje de programación es Python, debe entregar un enlace por Google Colab.
6. El taller puede ser realizado por grupos de 3 o 4 personas. Debe incluir sus nombres y códigos en una celda de texto al inicio del Colab, no se aceptarán reclamos sobre la omisión de esta información.
7. Debe realizarse una sola entrega por el grupo, en caso de presentarse más de una entrega sólo se valdrá una.
8. En caso de encontrar que su código parezca copiado de Internet o realizado por alguien de un semestre superior, el docente se reserva el derecho de citar al grupo para validar la autoría del código.

9. Todas las funciones principales debe ir comentadas explicando su funcionamiento con sus propias palabras, ejemplo:

```
#Descripción: Esta función realiza la suma de dos números
#Ejemplos (suma -1 2);-> 1, (suma 3 4); -> 7
def suma(a,b):
    return a+b
```

1. Conectividad

Implementar el algoritmo de Warsall para pesos mínimos en grafos dirigidos (sin bucles). Su aplicación debe permitir ingresar por un archivo que contenga la matriz de adyacencia que representa este grafo. Por ejemplo:

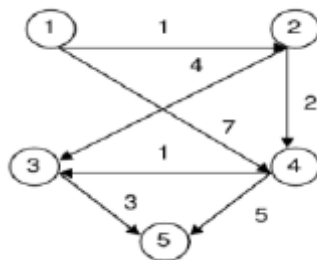


Figura 1: Ejemplo grafo

El archivo a cargar por entrada estándar sería de la siguiente forma:

```
5
1
4
0 1 0 7 0
0 0 4 2 0
0 0 0 0 3
0 0 1 0 5
0 0 0 0 0
```

- La primera línea indica el número de vértices
- La segunda línea indica el vértice inicial
- La tercera línea indica el vértices final
- Las siguientes líneas son la matriz de adyacencia

Su programa debe imprimir en pantalla (no guardar en archivo) el costo y la secuencia de pasos entre dos nodos a y b, para el ejemplo anterior:

Costo: 3 Camino: 1-2-4

2. Búsqueda por amplitud (BFS)

Diseñe una función que permita realizar una búsqueda por amplitud (BFS) en un grafo dado. Por ejemplo:

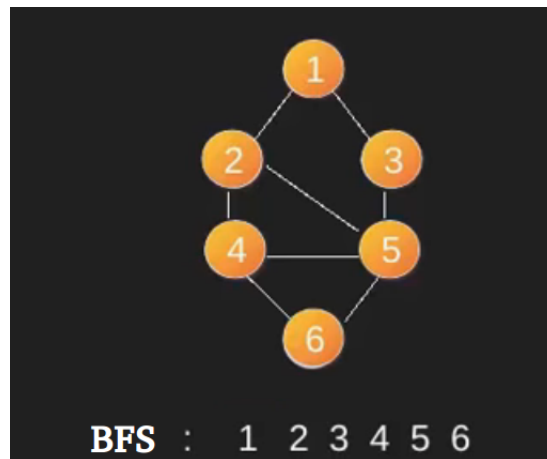


Figura 2: Ejemplo de búsqueda por amplitud

La función debe recibir un grafo y un índice numérico que indica el nodo de inicio de la búsqueda. Se aclara que se imprime la búsqueda, en la imagen se muestra una posible solución de las que existen.

El formato de la entrada es el mismo para el caso del algoritmo de Warshal pero son **grafos simples no dirigidos**. Recuerde que debe cargarse un archivo, el cual está especificado por el usuario.

La salida es el recorrido de búsqueda por amplitud, tomando la numeración de los vértices a partir del orden en que vienen en la matriz de adyacencia.

3. Rubricas de evaluación

Las rubricas que se van a utilizar para la evaluación consideran aspectos relacionados con las reglas del taller, calidad del informe y de realización de las implementaciones. En cada una se explica la asignación de puntos.

3.1. Sobre las reglas del taller

Criterio	Nivel 0 (0 pts)	Nivel 1 (3 pts)	Nivel 2 (5 pts)
Nombres y códigos presentes al inicio del Colab	No se cumple con esta regla	Uno o más de los estudiantes no incluye su código	Se encuentra al inicio del Colab los nombres y códigos de los estudiantes.
Funciones comentadas de acuerdo al enunciado	No se realizan comentarios	No se comentan todas las funciones realizadas o los comentarios en algunas de ellas no están de acuerdo al enunciado	Se comentan todas las funciones del código de acuerdo al enunciado
Organización de la entrega	No realiza la entrega	La entrega es desorganizada y no es claro donde soluciona cada punto	La entrega es organizada y se puede identificar claramente donde va cada punto.
Enlace de Colab	No entrega el enlace	El profesor no puede acceder al Colab y debe solicitarles el permiso	Se le da permiso de acceso a los usuarios de la Universidad del Valle

Total puntos 20

3.2. Sobre el desarrollo del taller

Criterio	Nivel 0 (0 pts)	Nivel 1 (10 pts)	Nivel 2 (20 pts)
Estructuras de datos en el código*	No usa matrices de adyacencia para representar grafos	Usa matrices de adyacencia para trabajar los grafos en sólo uno de los dos problemas	Usa matrices de adyacencia para trabajar ambos problemas
Implementación de los algoritmos usando matrices de adyacencia*	No implementa los algoritmos	Implementa los algoritmos pero no usa las matrices de adyacencia para resolver el problema o bien sólo implementa uno de los problemas	Implementa los algoritmos para ambos problemas y usa las matrices de adyacencia para resolverlos

Criterio	Nivel 0 (0 pts)	Nivel 1 (10 pts)	Nivel 2 (20 pts)
Claridad de la implementación*	No implementa los algoritmos	Implementa los algoritmos pero los pasos importantes no son claros y no están soportados con comentarios de apoyo	Implementa los algoritmos y son soportados por los comentarios de apoyo
Uso de colas en búsqueda por amplitud*	No implementa el algoritmo	Las colas no son la estructura fundamental para la implementación de BFS	Las colas son la estructura fundamental para la implementación de BFS
Respuesta de los algoritmos*	No implementa los algoritmos	La respuesta de uno de los dos algoritmos no sigue las especificaciones del enunciado	La respuesta de ambos problemas sigue lo esperado en el enunciado
Ejemplos	No realiza ejemplos	Realiza menos de 3 ejemplos de grafos o estos son grafos de menos de 4 vértices y 6 aristas, con sus ejemplos de ejecución	Realiza 3 o más ejemplos de grafos y estos tienen más de 4 vértices y 6 aristas, con sus ejemplos de ejecución.

*En caso de que uno de los dos algoritmos esté mal implementado, se tomarán nivel 1 en todas las rubricas asociadas a la implementación

Total puntos 120.

3.3. Total

El total del taller son 140 puntos que equivalen a 5.0, la formula para obtener su nota es:

$$\text{Nota taller} = 5,0 * \frac{\text{puntos obtenidos}}{140}$$

Sobre rúbricas

Su entrega será valorada a partir de las rúbricas consignadas previamente; cada una contiene una lista de criterios y niveles de desempeño. Para obtener la máxima calificación posible de su proyecto, debe cumplir las especificaciones del nivel más alto para cada criterio.

La nota se calculará con la suma de los puntos asignados al nivel que usted logre alcanzar para cada caso.