

**UNIVERSIDAD NACIONAL DE SAN AGUSTIN DE  
AREQUIPA**  
**FACULTAD DE INGENIERÍA DE PRODUCCIÓN Y SERVICIOS**  
**ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**



**LABORATORIO 9**

**GRAFOS**

**ALUMNO: VLADIMIR EDSON SOLÓRZANO HUAMANI**

**DOCENTE: EDITH PAMELA RIVERO TUPAC**

**Arequipa-Perú**

**2021**

1. Implementar el código de Grafo cuya representación sea realizada mediante LISTA DE ADYACENCIA. (3 puntos)

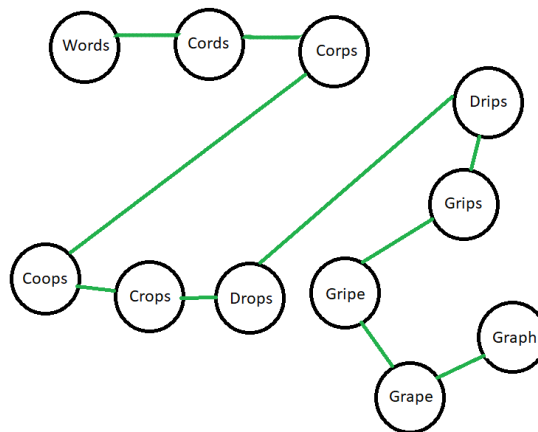
Se incluyo listas de adyacencia por medio de Listas enlazadas de vértices y aristas

2. Implementar BSF, DFS y Dijkstra con sus respectivos casos de prueba. (5 puntos)

Solo esta implementado DFS, aunque no funciona.

3. Solucionar el siguiente ejercicio: (5 puntos) El grafo de palabras se define de la siguiente manera: cada vértice es una palabra en el idioma inglés y dos palabras son adyacentes si difieren exactamente en una posición. Por ejemplo, las cords y los corps son adyacentes, mientras que los corps y crops no lo son.

a) Dibuje el grafo definido por las siguientes palabras: words cords corps coops crops drops drips grips gripe grape graph



b) Mostrar la lista de adyacencia del grafo.

```
Graph →Grape,
Grape →Graph, Gripe,
Gripe →Grape, Grips,
Grips →Gripe, Drips,
Drips →Grips, Drops,
Drops →Drips, Crops,
Crops →Drops, Coops,
Coops →Crops, Corps,
Corps →Coops, Cords,
Cords →Corps, Words,
Words →Cords,
```

4. Realizar un método en la clase Grafo. Este método permitirá saber si un grafo está incluido en otro. Los parámetros de entrada son 2 grafos y la salida del método es true si hay inclusión y false el caso contrario. (4 puntos).

No se implemento

1. ¿Cuántas variantes del algoritmo de Dijkstra hay y cuál es la diferencia entre ellas? (1 puntos)

Existen muchas variantes del algoritmo Dijkstra, como el uso de colas de prioridad, listas enlazadas y de variantes especializadas como colas especializadas. alguna de sus variantes se diferencia en la forma de aplicación del algoritmo ya sea para ganar velocidad de ejecución o solucionar un caso específico al cual un algoritmo común Dijkstra no podría solucionar por ejemplo el caso de tener vértices con un peso negativo.

2. Investigue sobre los ALGORITMOS DE CAMINOS MINIMOS e indique, ¿Qué similitudes encuentra, qué diferencias, en qué casos utilizar y por qué? (2 puntos)

Los algoritmos mas conocidos para la solución de caminos mínimos

	Diferencias	Casos a utilizar
Dijkstra	Resuelve desde un único vértice origen hasta todos los otros vértices del grafo	Si solo se desea hallar una distancia específica
Bellman - Ford	Resuelve el problema desde un origen si la ponderación de las aristas es negativa	Cuando haya aristas con peso negativo
Floyd	Resuelve toda la distancia de todos los vértices	Si se desea hallar todas las distancias
Búsqueda A*	Resuelve entre un par de vértices usando la heurística para agilizar la búsqueda.	Si se desea resolver el problema rápidamente.
Johnson	Resuelve el problema de todos los vértices al igual Floyd, pero es más rápido en grafos de baja densidad.	Cuando se tenga grafos de baja intensidad y se requiera velocidad.
Virterbi	Resuelve el problema del camino estocástico más corto con un peso probabilístico adicional en cada vértice	Cuando se quiera reducir a probabilidad de transmisión errónea.