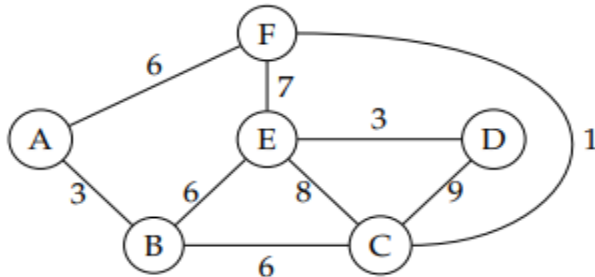


# MINI-PROVA 2024.1

## IF672 - Algoritmos e Estruturas de Dados

### Questão 1:



- a) Utilizando o grafo acima, complete o diagrama correspondente à execução do Algoritmo de **Dijkstra**.

Iter. #	Peso, Precursor					
	A	B	C	D	E	F
0	0, -	$\infty$ , ?	$\infty$ , ?	$\infty$ , ?	$\infty$ , ?	$\infty$ , ?
1	0, -	3, A	...	...	...	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮

- b) As arestas que ligam cada vértice ao seu precursor formam uma árvore geradora. A árvore geradora resultante dessa execução é uma MST? Justifique através de uma comparação com a árvore encontrada pelo Algoritmo de **Prim**.

### Questão 2:

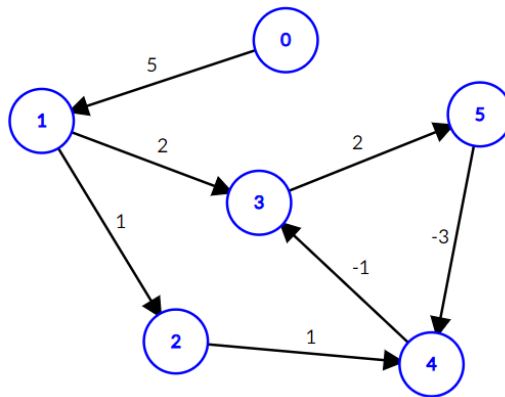
**OdnaL** tinha 8 milhões de reais disponíveis para fazer algumas melhorias em peças de seu carro. Ele pesquisou algumas informações sobre as melhorias e as organizou na seguinte tabela, considerando o custo da melhoria em cada peça e o ganho de performance no carro:

Melhorias	Asa traseira	Motor	Piso	Duto de freio	Suspensão traseira
Custo (em milhões)	3	4	2	1	2
Ganho (em %)	30	35	20	15	25

Usando o algoritmo de **knapsack**, faça passo a passo e indique qual o ganho máximo que OdnaL terá e em quais partes ele deve aplicar melhorias.

### Questão 3:

Usando o algoritmo de **Bellman-Ford**, identifique se o grafo abaixo possui ciclo(s) negativo(s), com o vértice 0 sendo o ponto inicial:

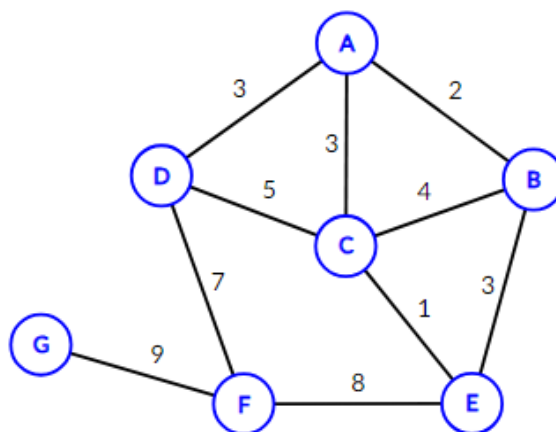


#### Questão 4:

O Algoritmo de Kruskal é um algoritmo guloso para encontrar uma árvore geradora de custo mínimo (MST) de um grafo ponderado. A execução do algoritmo segue os seguintes passos:

1. Inicie com a MST vazia.
2. Ordene as arestas do grafo em ordem crescente de peso.
3. Para cada aresta na ordem crescente, adicione-a à MST se ela não formar um ciclo com as arestas já presentes na árvore.

Considerando a execução do Algoritmo de Kruskal sobre o grafo abaixo, **determine a Árvore Geradora Mínima (MST)** e apresente a soma dos pesos das arestas selecionadas.



Extra: Qual a complexidade assintótica do Algoritmo Kruskal no pior caso? Justifique brevemente.