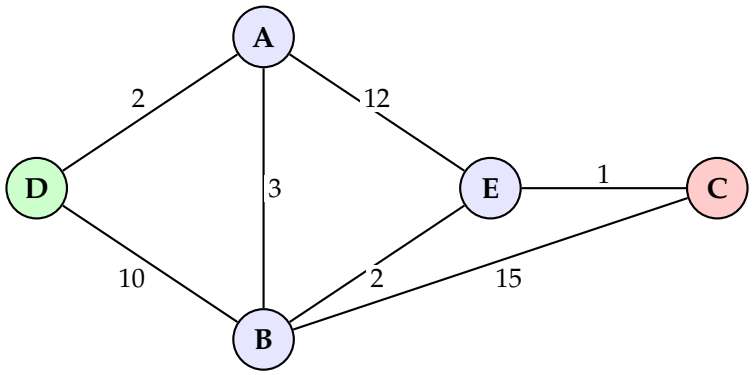
Algorithme de Dijkstra : activités de découverte

① Activité 1 : le coursier à vélo

Remarque 1 (Contexte). Karim est coursier à vélo dans une ville. Il doit livrer un colis depuis le dépôt D jusqu'au client C le plus rapidement possible. Le plan ci-dessous indique les temps de trajet en minutes entre chaque carrefour.



Exercice 1 *Comparer des chemins*

1. Lister **tous** les chemins possibles de D à C (sans passer deux fois par le même carrefour). Pour chacun, calculer le temps total.

Chemin		Temps (min)
D →	→ C	
D →	→ C	
D →	→ C	
D →	→ C	
D →	→ C	
D →	→ C	
D →	→ C	

2. Quel est le chemin le plus court **en nombre d'arêtes** ? Est-ce le plus rapide ?

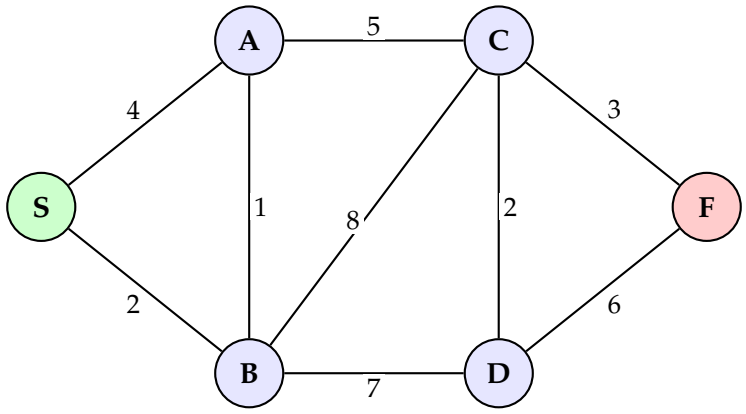
3. Quel est le chemin le plus rapide ? Combien d'arêtes comporte-t-il ?

4. Ce graphe ne comporte que cinq sommets. Si le plan de la ville en comportait 50, combien de chemins faudrait-il potentiellement examiner ? Cette méthode est-elle réaliste ?

② Activité 2 : découvrir le principe de Dijkstra

Remarque 2 (Idée clé). Au lieu d'explorer tous les chemins, Dijkstra procède par **sélection gloutonne** : à chaque étape, on traite le sommet le plus proche du départ parmi ceux qui restent. On met à jour les distances de ses voisins, puis on ne revient plus sur ce sommet. Comme tous les poids sont positifs, la distance d'un sommet traité ne peut plus diminuer.

Remarque 3 (Contexte). Un réseau de bus relie six arrêts. Les durées de trajet en minutes sont indiquées sur le schéma. On cherche le trajet le plus rapide de l'arrêt S (départ) à l'arrêt F (fin).



Exercice 2 *Dérouler l'algorithme pas à pas* On va remplir un tableau ligne par ligne. À chaque étape :

1. on **sélectionne** le sommet non traité ayant la plus petite distance provisoire ;
2. on **met à jour** les distances de ses voisins : si distance(sommet)+poids de l'arête < distance actuelle du voisin, on remplace et on note le prédécesseur ;
3. on **barre** le sommet traité (il ne sera plus modifié).

Convention : on note la distance suivie du prédécesseur en indice. Par exemple, 6_A signifie « distance 6, en venant de A ».

Étape	S	A	B	C	D	F	Sélectionné
Init	0	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	
1							
2							
3							
4							
5							

Aide pour l'étape 1. Le sommet non traité de plus petite distance est S (distance 0). On le sélectionne. Ses voisins sont A et B :

— distance de A : $\min(+\infty,\ 0+4) = 4$, prédécesseur S $\longrightarrow$ on écrit 4_S ;

— distance de B : $\min(+\infty,\ 0+2) = 2$, prédécesseur S $\longrightarrow$ on écrit 2_S.

Compléter la ligne « Étape 1 » puis continuer seul(e).

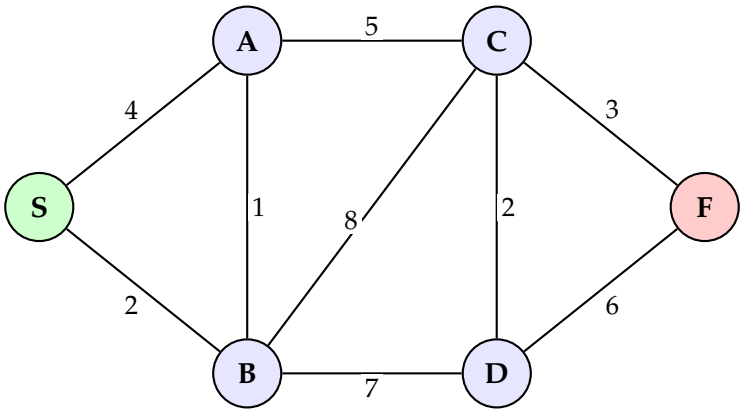
Exercice 3 *Reconstituer le chemin*

1. Dans le tableau complété, repérer le prédécesseur de F. Puis le prédécesseur de ce prédécesseur, et ainsi de suite jusqu'à S. Noter la suite des sommets obtenue.

2. Inverser cette suite pour obtenir le plus court chemin de S à F.

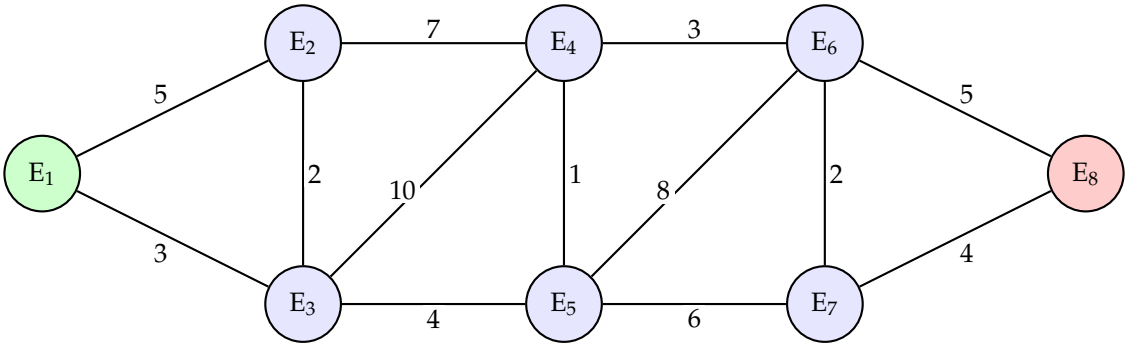
3. Quelle est la durée totale du trajet ?

4. Surligner ce chemin sur le graphe ci-dessous.



③ Activité 3 : optimiser un réseau de livraison

Remarque 4 (Contexte). Une entreprise de logistique planifie ses tournées entre huit entrepôts. Le graphe ci-dessous représente les temps de trajet en minutes entre les entrepôts. Déterminer le trajet le plus rapide de l'entrepôt E_1 à l'entrepôt E_8.



Exercice 4 *Appliquer l'algorithme de Dijkstra*

1. Compléter le tableau ci-dessous en appliquant l'algorithme de Dijkstra depuis E_1.

Étape	E_1	E_2	E_3	E_4	E_5	E_6	E_7	E_8	Sélectionné
Init	0	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	$+\infty$	
1									
2									
3									
4									
5									
6									
7									

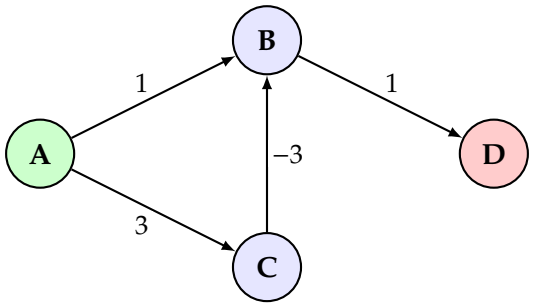
2. Reconstituer le plus court chemin de E_1 à E_8 en remontant les prédécesseurs.

3. Quelle est la durée totale de ce trajet ?

4. Le chemin le plus court en nombre d'arêtes entre E_1 et E_8 est-il le même ? Justifier.

④ Activité 4 : cas particuliers et pièges

Exercice 5 *Pourquoi les poids négatifs posent problème* Considérer le graphe orienté ci-dessous.



1. Appliquer Dijkstra depuis A. Quel sommet est traité en deuxième (après A) ? Quelle distance provisoire obtient alors D ?

2. Une fois B traité, sa distance est considérée comme définitive. Poursuivre l'algorithme jusqu'au bout. Quelle distance finale Dijkstra donne-t-il pour D ?

3. Calculer le poids du chemin A $\to$ C $\to$ B $\to$ D.

4. Dijkstra donne-t-il la bonne réponse ici ? Expliquer pourquoi le poids négatif de l'arc C $\to$ B pose problème.