

**Exercice 1**

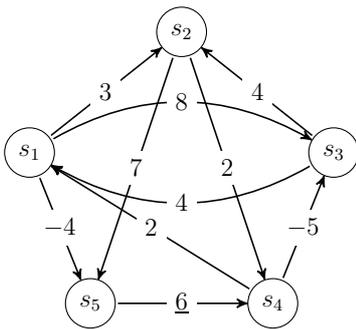
Une preuve de la validité de l'algorithme de *Floyd-Warshall* est basée sur le lemme suivant :

**Lemme 1 (Les sous-chemins des plus courts chemins sont des plus courts chemins)** *Étant donné un graphe orienté pondéré  $G = (X, A)$  et une fonction de pondération  $w : A \rightarrow \mathbb{R}$ , soit  $p = \langle v_1, v_2, \dots, v_k \rangle$  un plus court chemin d'un sommet  $v_1$  à un sommet  $v_k$  et, pour tout couple d'entiers  $i, j$  quelconques tels que  $1 \leq i \leq j \leq k$ , soit  $p_{ij} = \langle v_i, v_{i+1}, \dots, v_j \rangle$  un sous-chemin de  $p$  allant du sommet  $v_i$  au sommet  $v_j$ . Alors,  $p_{ij}$  est un plus court chemin de  $v_i$  à  $v_j$ .*

Démontrer le **Lemme 1**.

**Exercice 2**

Soit  $G = (X, A)$  le graphe orienté avec une fonction de pondération  $w : A \rightarrow \mathbb{R}$  représenté ci-dessous.



L'exécution de l'algorithme de *Floyd-Warshall* redonné en annexe page 2 a produit le résultat final suivant :

$$D^{(5)} = \begin{pmatrix} 0 & 1 & -3 & 2 & -4 \\ 1 & 0 & -3 & 2 & -3 \\ 4 & 4 & 0 & 6 & 0 \\ -1 & -1 & -5 & 0 & -5 \\ 5 & 5 & 1 & 6 & 0 \end{pmatrix} \quad \Pi^{(5)} = \begin{pmatrix} \text{NIL} & s_3 & s_4 & s_5 & s_1 \\ s_3 & \text{NIL} & s_4 & s_2 & s_1 \\ s_3 & s_3 & \text{NIL} & s_2 & s_1 \\ s_3 & s_3 & s_4 & \text{NIL} & s_1 \\ s_3 & s_3 & s_4 & s_5 & \text{NIL} \end{pmatrix}$$

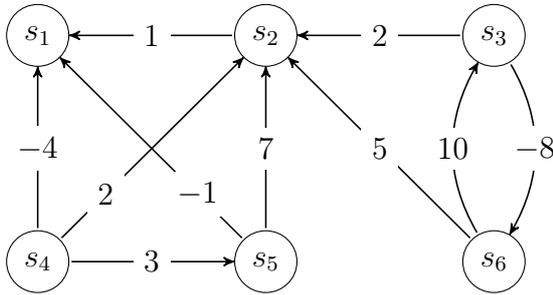
1. Expliquer comment connaître à partir de la matrice  $D^{(5)}$  la distance du plus court chemin entre les sommets  $s_5$  et  $s_4$  puis entre les sommets  $s_2$  et  $s_5$ .
2. Expliquer comment reconstruire à partir de la matrice  $\Pi^{(5)}$  le plus court chemin entre les sommets  $s_5$  et  $s_4$  puis entre les sommets  $s_2$  et  $s_5$ .
3. (\*) Le but de cet exercice est de calculer le plus court chemin entre chaque couple de sommets, c'est-à-dire de recalculer les matrices  $D^{(5)}$  et  $\Pi^{(5)}$  en appliquant l'algorithme de *Floyd-Warshall* redonné en annexe page 2.

Outre les matrices  $D^{(i)}$ , votre algorithme calculera également à chaque étape la matrice de liaison  $\Pi^{(i)}$  ; plus précisément, vous calculerez la séquence  $\Pi^{(0)}, \Pi^{(1)}, \dots, \Pi^{(n)}$ , où  $\Pi = \Pi^{(n)}$  et  $\pi_{ij}^{(k)}$  est défini comme le prédécesseur du sommet  $s_j$  sur le plus court chemin partant du sommet  $s_i$ , et dont tous les sommets intermédiaires sont dans  $\{s_1, s_2, \dots, s_k\}$ . L'initialisation de la matrice  $\Pi$  se fait de la manière suivante, pour  $k = 0$ , un plus court chemin de  $s_i$  vers  $s_j$  ne possède aucun sommet intermédiaire. Donc,

$$\pi_{ij}^{(0)} = \begin{cases} \text{NIL} & \text{si } s_i = s_j \text{ ou } w(s_i, s_j) = \infty, \\ s_i & \text{si } s_i \neq s_j \text{ et } w(s_i, s_j) < \infty. \end{cases}$$

### Exercice 3

Soit  $G = (X, A)$  le graphe orienté avec une fonction de pondération  $w : A \rightarrow \mathbb{R}$  représenté ci-dessous.



1. Calculer la fermeture transitive du graphe  $G$ .
2. Proposer un algorithme en  $\mathcal{O}(n^3)$  pour calculer la fermeture transitive d'un graphe.

## Annexe

FLOYD-WARSHALL( $W$ )

- 1  $n \leftarrow \text{nombre\_de\_lignes}(W)$
- 2  $D^{(0)} \leftarrow W$
- 3 **pour**  $k \leftarrow 1$  à  $n$
- 4     **pour**  $i \leftarrow 1$  à  $n$
- 5         **pour**  $j \leftarrow 1$  à  $n$
- 6              $d_{ij}^{(k)} \leftarrow \min(d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{(k-1)})$
- 7             **calcul**  $(\pi_{ij}^{(k)})$
- 8 **retourner**  $D^{(n)}$  et  $\Pi^{(n)}$