

# Recherche Operationnelle

Vincent Gripon

19 avril 2011

Dans ce TD, nous allons voir comment construire la table de routage associée à la recherche de plus courts chemins dans l'algorithme de Roy-Warshall.

## Question 1)

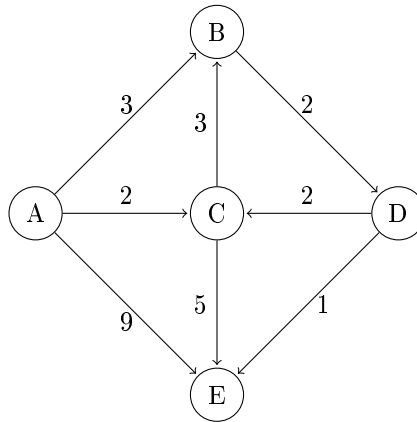
On propose de changer l'algorithme de Roy-Warshall de la façon suivante :

```
sp = matrice(n,n)
routage = matrice(n,n)

pour i de 1 à n
  pour j de 1 à n
    sp(i,j) <- delta(i,j)

pour k de 1 à n
  pour i de 1 à n
    pour j de 1 à n
      si sp(i,k)+sp(k,j) < sp(i,j)
        alors
          sp(i,j) <- sp(i,k)+sp(k,j)
          routage(i,j) <- k
```

Trouver les matrices `sp` et `routage` associées au graphe suivant :



**Question 2)**

Pour retrouver le chemin le plus court menant d'un noeud à un autre, on utilise le principe suivant :

- Si  $roulage(i, j)$  n'a pas été modifié alors le plus court chemin menant de  $i$  à  $j$  est l'arrête les reliant,
- Sinon  $k = roulage(i, j)$  désigne un noeud par lequel un des plus court chemin passe, il reste donc à trouver un plus court chemin de  $i$  à  $k$  et de  $k$  à  $j$ .

Proposer un algorithme récursif qui permette de trouver le plus court chemin entre deux noeuds du graphe en utilisant la matrice *roulage*. L'appliquer pour trouver le plus court chemin reliant  $A$  à  $E$ .

**Question 3)**

Quelle est la complexité de votre algorithme? Est-ce gênant vis à vis de Roy-Warshall?

**Question 4)**

Quelle est la complexité du calcul des chemins entre tous noeuds du graphe?

**Question 5)**

On se propose maintenant de construire un tableau de routage pour l'algorithme de Dijkstra. Proposer une modification de l'algorithme qui permette de remplir ce tableau.

**Question Bonus**

Pour l'algorithme de Dijkstra, on peut légèrement améliorer la fonction récursive qui permet de reconstituer un chemin le plus court. Expliquer en quoi elle est différente de celle pour Roy-Warshall.