

Rapport TP Test (ILU4)

Question 0 b):

BellmanFordAlgo(g,s)

Données: graphe (classe BellmanFord) = tableau de Edges(aretes) g, source (numero de l'arete) s

Resultat: String des distances entre la source et les autres arretes

Preconditions:

s \geq 0

s < size(g)

g non nul

Postconditions:

g et s inchanges

size(t) = size(g)

Objectif: Trouver le chemin le plus court depuis un sommet donne vers tous les autres dans un graphe pondere, en tenant compte des poids negatifs mais en detectant les cycles de poids negatif.

Entrées:

Un graphe dirige represente par un nombre de sommets V et un nombre d'aretes E.

Une liste d'aretes sous forme de triplets (source,destination,poids).

Un sommet source e partir duquel calculer les plus courts chemins.

Sorties:

La distance minimale de la source e chaque sommet (ou Integer.MAX_VALUE (considéré comme infini) si le sommet est inatteignable).

Un message si un cycle de poids negatif est detecte.

Question 1:

Voir BellmanFordTest.java

Question 2:

Question 3:

Oui, il suffit d'une suite de test comprenant:

>un graphe quelconque (sans cycle negatif)

>un graphe contenant un cycle negatif

Exemple:

Test 1 : Graphe simple sans cycle négatif

V=5

E=8

0 1 -1

0 2 4

1 2 3

1 3 2

1 4 2

3 2 5

3 1 1

4 3 -3

SOURCE=0

EXPECTED=0 -1 2 -2 1

Test 2 : Graphe avec cycle négatif

V=3

E=3

0 1 1

1 2 -1

2 0 -1

SOURCE=0

EXPECTED=CYCLE_NEGATIF

Question 4: