

Nom	
Prénom	
Groupe	

Note	
------	--

Algorithmique - Info-SPE

Partiel n° 2

D.S. 312354.45 BW (10 mai 2011 - 09 :00)

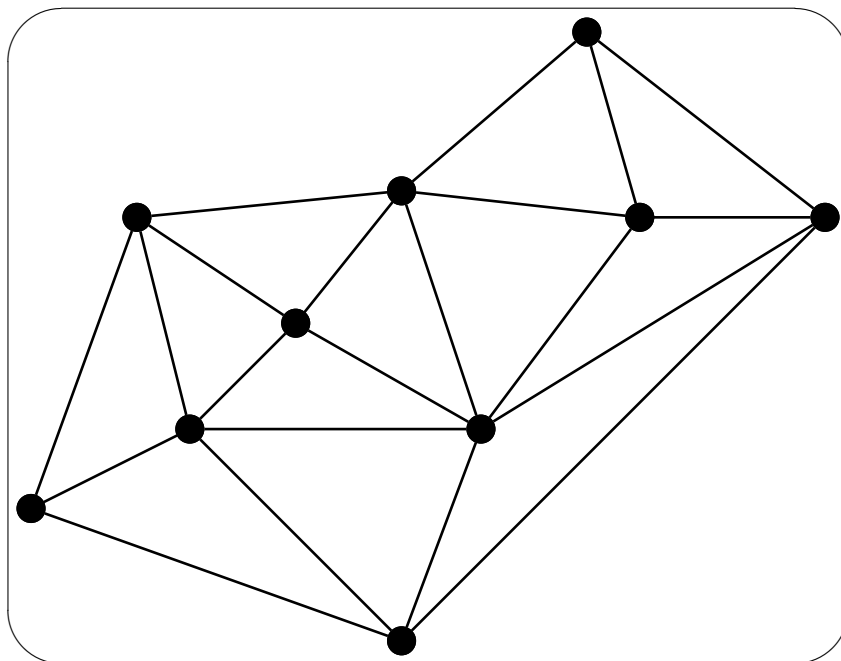
Feuilles de réponses

Réponses 1 (Gisement épuisant... – 5 points)

1. La solution correspond à :

2. Dans le cas de la figure 1, combien faut-il sécuriser de galeries ?

3. Proposer une solution graphique (Surligner les galeries que vous vous proposez de sécuriser).



4. Pour un réseau de N points d'extraction, il faut sécuriser

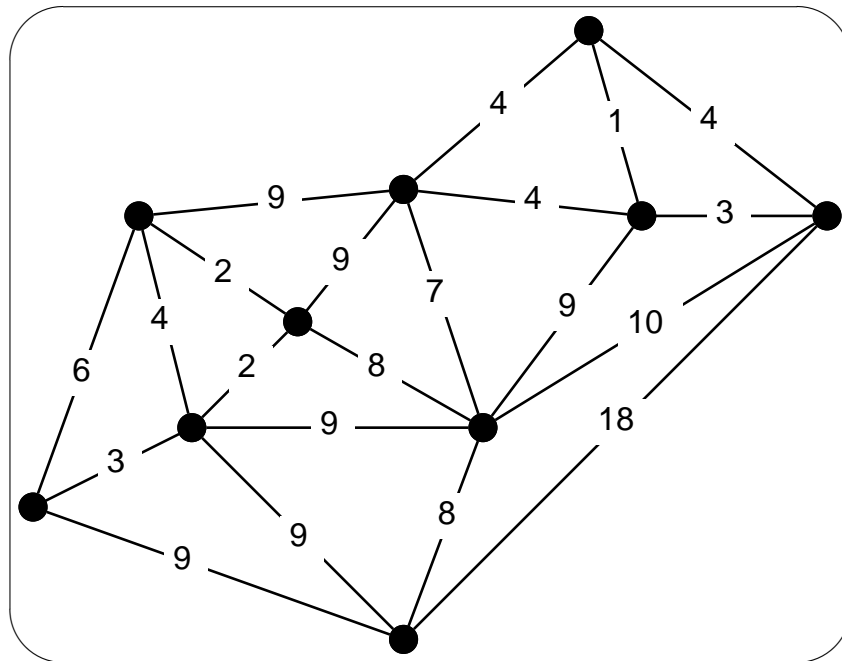
galeries.

5. Justification :

On affine l'analyse du problème : pour chaque galerie, nous avons évalué le coût des travaux de sécurisation (voir figure 2).

6. Comment dans ce cas sécuriser l'accès à toutes les grottes au moindre coût ?

7. Proposer une solution graphique (Tracer en gras les galeries qu'il faut sécuriser).

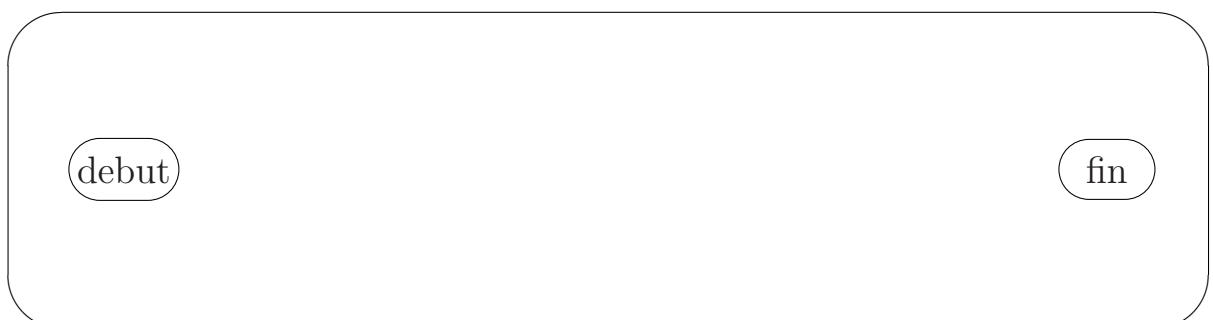


8. La solution est-elle unique ? OUI – NON

9. Justification : _____

Réponses 2 (Mangez des crêpes – 16 points)

1. Graphe représentant le projet :




```

algorithme procedure tri_topo
  parametres locaux
    t_graphe_d    G
  parametres globaux
    t_pile    tri    /* contient des t_listsom */
  variables

```

debut

fin algorithme procedure tri_topo

3. (a) Comment calculer les dates au plus tôt de chaque tâche à partir de ce type de graphe ?

Dates au plus tôt pour la recette :

<i>debut</i>	A	B	C	D	E	F	G	H	I	J	<i>fin</i>

(b) Durée minimale avant de pouvoir déguster la crêpe :

Comment obtenir la durée minimale du projet ?

(c) *Comment calculer les dates au plus tard de chaque tâche ?*

Dates au plus tard pour la recette :

<i>debut</i>	A	B	C	D	E	F	G	H	I	J	<i>fin</i>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

(d) *Comment obtient-on les tâches critiques dans ce type de projet ?*

Tâches critiques pour le présent projet : une croix (X) pour chaque tâche critique !

<i>debut</i>	A	B	C	D	E	F	G	H	I	J	<i>fin</i>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

(e) **Spécifications :** La fonction `duree_minimale` (G , *source*, *finale*) calcule la longueur du plus long chemin (la durée minimale du projet) dans le graphe G , entre les tâches *source* et *finale*.

```
fin algorithme fonction duree_minimale
```

Réponses 3 Construire un ARPM par suppression – 9 points

1. Principe du test de connexité entre deux sommets par un parcours profondeur :

2. **Spécification** : la fonction `lie_rec(dst,ps,T,M)` teste (elle renvoie donc un booléen) s'il existe un chemin depuis le sommet pointé par `ps` jusqu'au sommet de numéro `dst`. La fonction utilise le principe de la question précédente. La matrice `T` décrit les arêtes qui ont été supprimées du graphe et le vecteur de booléens `M` sert de vecteur de marques pour le parcours profond.

```

algorithme fonction lie_rec : booléen
parametres locaux
    entier          dst
    t_listsom       ps
    t_mat_entiers   T
parametres globaux
    t_vect_booléens M
variables

```

debut

[illegible]

```
fin algorithme fonction lie_rec
```

3. Condition d'arrêt de l'algorithme de suppression des arêtes :

4. **Spécification :** la procédure `revdel(g,E,T)` applique l'algorithme de construction de l'ARPM par suppression des arêtes dans le graphe g , à partir de l'ensemble d'arête E et indique dans la matrice T les arêtes supprimées.

algorithme procedure revdel

parametres locaux

t_graphe_d	g
------------	---

parametres globaux

ensemble E

t_mat_entiers T

variables

debut

[illegible]

```
fin algorithme procedure revdel
```


5. Surligner les arêtes supprimées par l'algorithme

