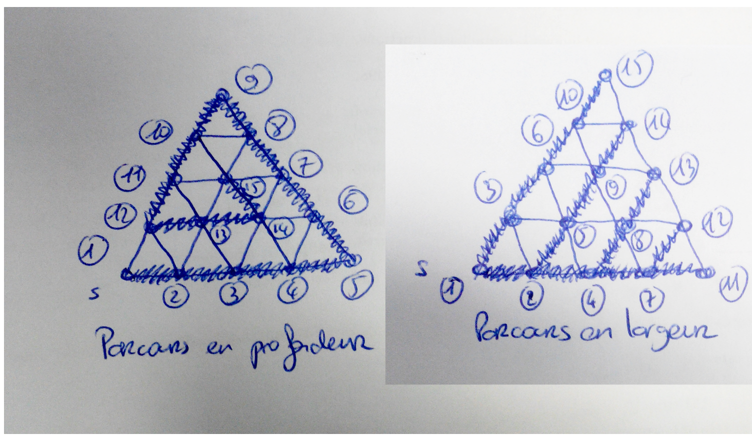
- Examen d'algos de graphes -

- Corrigé succinct -

Barême :
Exercice 1 sur 4 points : 1 - 1 - 0.5 - 1.5
Exercice 2 sur 9 points : 0.5 - 1 - 1 - 1 - 2.5 - 2 - 1
Exercice 3 sur 7 points : 1.5 - 1.5 - 1 - 1 - 2

- Exercice 1 - Parcours - 4pts -

a. et b. Justes les dessins des parcours étaient demandés, avec l'ordre de découverte :



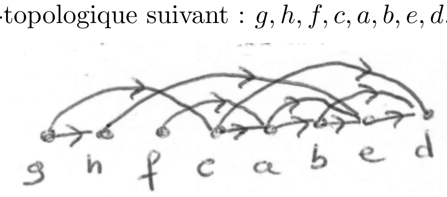
c. Pour le parcours en largeur, les diagonales penchées vers le haut à gauche sont les ensembles de sommets ayant même niveau. L'ensemble $\{s\}$ forme le sommet de niveau 0. Il y a $n-1$ autres diagonales donc le niveau maximal est $n-1$.

d. Pour le parcours en profondeur, on obtient un chemin hamiltonien. Le niveau maximal atteint est donc le nombre de sommet moins un (s étant de niveau 0). Dans T_n, il y a n sommets sur la même ligne horizontale que s puis $n-1$ au dessus, puis $n-2$... et enfin 1 sommet tout en haut. Le graphe T_n a donc $1+2+\ldots+(n-1)+n = \frac{1}{2}n(n+1)$ sommets. Le niveau maximal atteint est alors $\frac{1}{2}n(n+1)-1$.

- Exercice 2 - Plus longs chemins dans les DAGs - 9pts -

a. Un tri-topologique est un ordre $v_1, \ldots, v_n$ sur les sommets de D tel que si v_iv_j est un arc de D alors $i < j$ (les arcs vont tous de la gauche vers la droite).

b. On utilise le principe vu en cours : on trouve une source du graphe, on l'enlève et on recommence. On obtient par exemple le tri-topologique suivant : g, h, f, c, a, b, e, d.



c. En utilisant le tri-topologique précédent, on obtient : $l(g) = 0$, $l(h) = l(g) + 1$, $l(f) = 0$, $l(c) = l(g) + 1 = 1$, $l(a) = l(c) + 1 = 2$, $l(b) = l(a) + 1 = 3$, $l(e) = l(b) + 1 = 4$ et $l(d) = l(e) + 1 = 5$. Finalement le maximum des valeurs de l est 5 ici.

d. La boucle de la ligne 2 s'exécute n fois. A priori la boucle de la ligne 4 aussi, mais en fait les lignes 5 et 6 s'exécutent au plus une fois par arc du graphe. Ainsi les opérations des lignes 5 et 6 demandent un temps en $O(m)$ au total. En tout, les 2 à 9 prennent finalement un temps en $O(n + m)$. Si de plus, le tri-topologique de la ligne 1 est implémenté avec un temps d'exécution en $O(n + m)$, on obtient un temps total pour PLC en $O(n + m)$.

e. On prend l'hypothèse de récurrence suivante $\mathcal{H}_i$:'après l'étape i de la boucle de ligne 2, $l(v_i)$ contient la longueur d'un plus long chemin terminant en v_i'.
Pour $i = 1$, v_1 est une source de D et un chemin finissant en v_1 a longueur 0, $\mathcal{H}_1$ est vraie.
Supposons que $\mathcal{H}_i$ est vraie pour $i = 1, \ldots, k$ est montrons $\mathcal{H}_{k+1}$. Soit $v_{i_1}, v_{i_2}, \ldots, v_{i_l}$ un chemin de longueur maximale terminant en $v_{i_l} = v_{k+1}$. Comme $v_1, \ldots, v_n$ est un tri-topologique de D, on a $i_1 < i_2 < \ldots < i_l$. Par ailleurs, $v_{i_1}, \ldots, v_{i_{l-1}}$ est un plus long chemin terminant en $v_{i_{l-1}}$, donc par $\mathcal{H}_{i_{l-1}}$, on a $l(v_{i_{l-1}}) = l - 1$. Lors de la boucle des lignes 4 à 8, pour $i = k + 1$ on aura $l(v_{k+1}) = l(v_{i_l}) \geq l - 1 + 1 = l$. Si on avait $l(v_{i_l}) > l$, la dernière affectation de $l(v_{i_l})$ se faisant pour un indice $j < i_l = k + 1$, on aurait $l(v_j) > l - 1$ et v_j serait la fin d'un chemin de longueur $> l - 1$ par $\mathcal{H}_j$. Donc $v_{i_1}, \ldots, v_{i_l}$ ne serait pas un plus long chemin terminant en v_{k+1}, ce qui est une contradiction. Finalement $l(v_{k+1}) = l$ et vaut la longueur d'un plus long chemin terminant en $v_{i_{k+1}}$. D'où $\mathcal{H}_{k+1}$ est vraie.

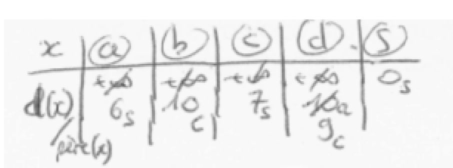
f. On introduit la variable $pas(v)$ qui contient le nom du sommet précédent v dans un plus long chemin terminant en v.
On ajoute les lignes suivantes :
L$'_3$ $pas(v_i) \longleftarrow \emptyset$;
L$'_6$ $pas(v_i) \longleftarrow v_j$;
L$'_9$ Noter v_k le sommet tel que $l(v_k) = \max\{l(v_i) \ : \ i = 1, \ldots, n\}$;
Tant que $v_k \neq \emptyset$ faire
  Ecrire v_k ;
  $v_k \longleftarrow pas(v_k)$;

g. On crée un graphe orienté D dont les sommets sont les boîtes et on ajoute l'arc $B_1 \to B_2$ dans D si la boîte B_1 s'emboîte dans la boîte B_2. Une plus longue séquence de boîtes pouvant s'emboîter correspond alors à un plus long chemin de D. Comme D est acyclique (on ne peut pas faire un cycle de boîtes s'emboîtant les unes dans les autres...), on peut utiliser l'algo de l'exercice pour calculer une telle séquence.

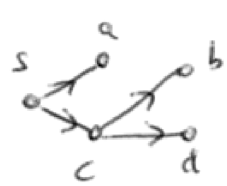
- Exercice 3 - Plus courts chemins - 7pts -

a. Voir cours.

b. On déroule l'algo de Dijkstra. On choisit les sommets selon l'ordre suivant : s, a, c, d, b, en mettant à jour dans le tableau ci-dessous les voisins sortants du sommet sélectionné à chaque étape. L'arborescence des PCC est représentée ci-dessous.



x	a	b	c	d	s
$d(x)$ / père(x)	$\cancel{+\infty}$ 6_s	$\cancel{+\infty}$ 10_c	$\cancel{+\infty}$ 7_s	$\cancel{+\infty}$ $\cancel{10_a}$ 9_c	0_s



Les PCC de s à a, c et d sont inchangés, aucun n'utilisant l'arc cb dont la longueur augmente. Pour atteindre b, si onutilise cb on aura une distance de 17, si on utilise db on aura 16. Il faut donc remplacer cb par db dans l'arborescence des PCC.

d. Les lignes suivantes permettent de calculer l, la nouvelle valeur de $d(y)$:

L_1 $l \longleftarrow d(x) + l(xy) + p\,;$
L_2 Pour tout voisin entrant z de y faire
L_3 Si $d(z) + l(zy) < l$ alors faire
L_4 $l \longleftarrow d(z) + l(zy)\,;$
L_5 $pere(y) \longleftarrow z\,;$

e. Comme y est une feuille de l'arborescence des PCC et que la longueur de xy augmente, seul le PCC de s à y est détruit par ce changement. En effet les autres chemins calculés ont toujours la même longueur et tout chemin emprûntant l'arc xy a sa longueur totale qui augmente. Notons D' le graphe identique à D sauf pour la longueur de l'arc xy qui est augmentée de p. Soit P un PCC de s à y dans D'. Notons z le sommet précédant y dans P. Le chemin $P \setminus y$ est un PCC dans D' de s à z. Comme il ne contient pas xy, c'est aussi un PCC de s à z dans D. Il a donc été précédemment bien calculé par l'algorithme de Dijkstra. Il suffit de rajouter l'arc zy pour retrouver P. L'algo proposé à la question e. teste tous les chemins de ce type là et choisit celui de longueur minimale, il calculera donc bien un PCC de s à y dans D'.