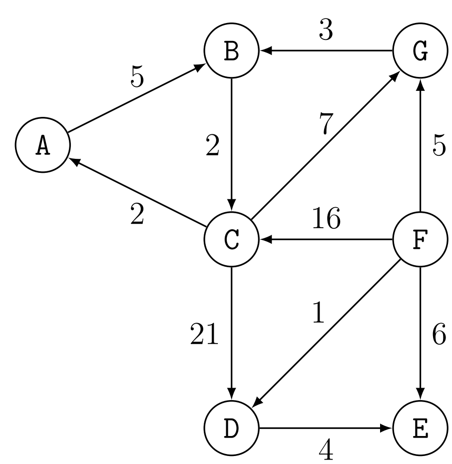
Algorithmique 1

Devoir final

Les notes de cours et de TD sont autorisées. Pour les questions demandant de dérouler un algorithme, vous veillerez à faire apparaître les étapes de calcul permettant de juger de votre bonne compréhension (le résultat final est rarement suffisant). Le barème est indiqué à titre indicatif, il n'est pas définitif.

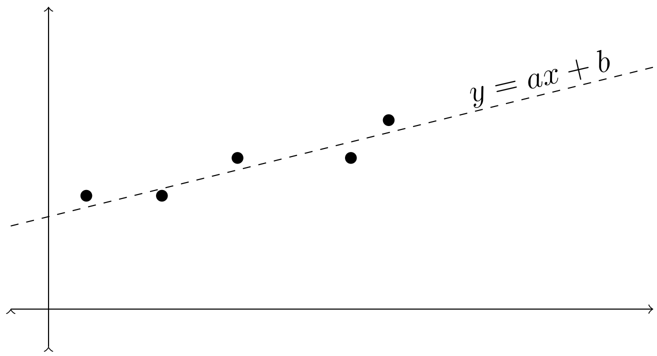
1 Algorithme de Dijkstra



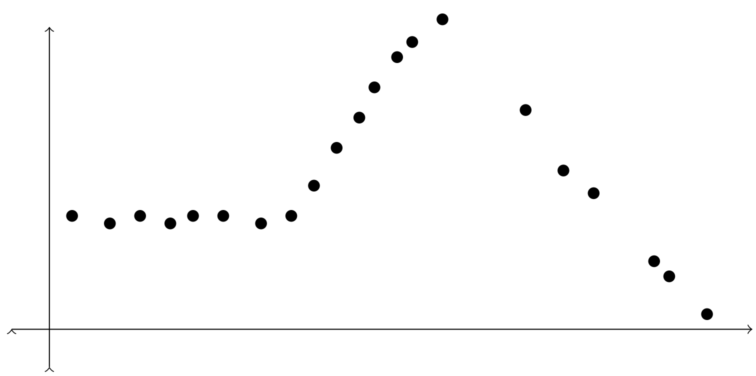
Question 1. Exécuter l'algorithme de Dijkstra sur le graphe ci-contre à partir du sommet `F`. On prendra soin de faire figurer les étapes, en montrant l'évolution des deux tableaux représentant respectivement la plus courte distance depuis `F` estimée, et le prédécesseur de chaque sommet sur un plus court chemin estimé.

2 Méthode des moindres carrés segmentée

On considère une suite P de n points $p_1 = (x_1, y_1), \ldots, p_n = (x_n, y_n) \in \mathbb{R}^2$ avec $x_1 < x_2 < \ldots < x_n$. La *méthode des moindres carrés* consiste à trouver une droite d'équation $y = ax + b$ qui minimise l'erreur $\sum_{k=1}^{n}(y_k - ax_k - b)^2$, comme le montre la figure suivante :



Dans cette exercice, nous allons améliorer cette méthode. Au lieu d'approximer P avec une seule droite, nous allons découper P en portions de points consécutifs et chaque portion est approximable par une droite. Par exemple, la figure suivante montre une suite P de points pour laquelle il semble judicieux de découper P en 3 portions :



Pour tout $i, j \in \{1, \ldots, n\}$ avec $i \leq j$, soit $e_{i,j}$ l'erreur minimale pour approximer la portion de points consécutifs $p_i, p_{i+1}, \ldots, p_j$ par une seule droite. Il s'agit du minimum de $\sum_{k=i}^{j}(y_k - ax_k - b)^2$ pour a, b dans $\mathbb{R}$. On suppose dans un premier temps que l'on a précalculé toutes les valeurs de $e_{i,j}$.

Une solution au problème est une suite d'indices $1 = i_1 < i_2 < \ldots < i_k = n$, qui représente un découpage de la suite de points P en portions : il s'agit de placer une droite approximant la première portion $p_{i_1}, p_{i_1+1}, \ldots, p_{i_2}$, une droite approximant la deuxième portion $p_{i_2}, p_{i_2+1}, \ldots, p_{i_3}$, etc. ; le nombre de portions approximées par des droites est $k - 1$. Afin de trouver la meilleure solution, le problème est de calculer le minimum de la quantité suivante :

$$(k-1)C + \sum_{\ell=1}^{k-1} e_{i_\ell, i_{\ell+1}} \qquad (1)$$

où C est une constante strictement positive.

Question 1. Expliquer l'intuition de l'expression (1).

Le nombre de solutions potentielles (c'est-à-dire, de suites finies strictement croissantes commençant par la valeur 1 et finissant par la valeur n) est exponentielle en n. Pourtant, nous allons donner un algorithme en temps polynomial en n pour résoudre notre problème. Dans un premier temps, on suppose que les valeurs $e_{i,j}$ correspondantes sont déjà précalculées.

Question 2. Procédez à l'analyse du problème en utilisant un paradigme vu en cours.

Question 3. Donner un algorithme en temps polynomial qui prend en entrée la suite de points P ainsi que les valeurs $e_{i,j}$ correspondantes déjà précalculées, et qui renvoie le minimum de l'expression (1).

Question 3. Donner la complexité temporelle et spatiale de votre algorithme.

Question 4. Expliquer comment construire une solution $1 = i_1 < i_2 < \ldots < i_k = n$ qui rende l'expression (1) minimale.

Question 5. (bonus) Expliquer comment calculer en $O(n^3)$ toutes les valeurs $e_{i,j}$.

Question 6. (bonus) Expliquer comment calculer en $O(n^2)$ toutes les valeurs $e_{i,j}$.

3 Flots

On considère le réseau de flot $\mathcal{R} = (S, A)$ de la figure suivante, où `s` joue le rôle de source et `t` le rôle de puits. On note c la capacité du réseau.

Question 1. Exécuter l'algorithme de Ford-Fulkerson sur le réseau afin de trouver un flot maximum de `s` à `t`. Indiquer les chemins améliorants successifs.

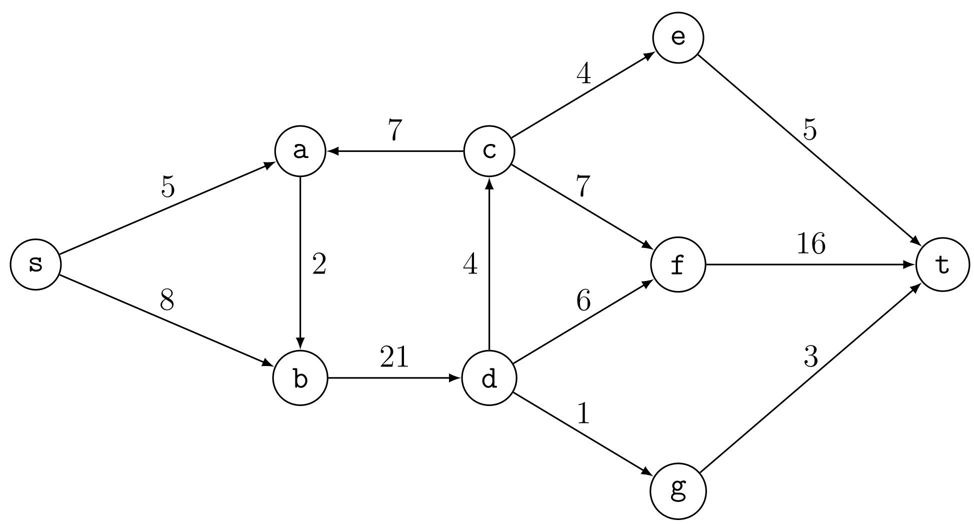
On appelle *graphe support* d'un flot f le sous-graphe constitué des arcs où le flot f est strictement positif. Un flot f est dit *acyclique* si son graphe support est lui-même acyclique.

Question 2. Proposer un flot pour le réseau $\mathcal{R}$ qui ne soit pas acyclique.

Question 3. Montrer que pour tout flot f, il existe un flot acyclique de même valeur que f.

Question 4. Soit f un flot acyclique, montrer que s'il existe un arc (u, v) tel que $f(u, v) > 0$, alors il existe un chemin de `s` à `t` dans le graphe support de f.

Question 5. Un *flot chemin* est un flot f dont le graphe support est un chemin simple de `s` à `t`. Montrer que tout flot acyclique peut être écrit comme la somme d'au plus $|A|$ flots chemins.


s
a
b
c
d
e
f
g
t
5
8
2
7
21
4
4
7
6
1
5
16
3