EA4 – Éléments d'algorithmique
TD n^{o} 6

Exercice 1 : déroulement du tri rapide

On considère le tableau `T` suivant :

Décrire le déroulement des variantes suivantes du tri rapide sur le tableau `T` :
- variante avec mémoire auxiliaire, en prenant le premier élément comme pivot ;
- *idem*, mais en choisissant toujours le pivot de manière optimale ;
- variante « en place », le pivot étant toujours le premier élément.

Exercice 2 : des vis et des écrous

Un charpentier dispose d'un tas d'écrous et de vis et souhaite associer chaque écrou à la vis correspondante. Chaque écrou correspond à une seule et unique vis (et réciproquement). En essayant de comparer une vis à un écrou, le charpentier peut voir lequel est le plus grand, mais ne peut pas comparer directement deux vis ou deux écrous. Comment peut-il alors associer efficacement chaque vis à son écrou ?

Exercice 3 : des tris linéaires ?

En cours, nous avons montré que les algorithmes de tri ne pouvaient pas avoir une complexité meilleure que $(n \quad n)$ *en moyenne. Dans cet exercice, nous allons pourtant étudier des tris de complexité linéaire.*

1. Expliquer en quoi ce n'est pas une contradiction avec la borne $(n \quad n)$.

2. On considère des tableaux ne contenant que des et des . Décrire un algorithme de complexité linéaire pour trier un tel tableau.

3. Plus généralement, on considère des tableaux contenant des valeurs choisies dans un ensemble fini, par exemple des entiers dans l'intervalle $[\ ,m]$. Donner un algorithme de complexité linéaire en `len(T)` permettant de trier un tel tableau. Quelle est la complexité en espace de votre algorithme ?

4. Le problème du *drapeau hollandais* est le suivant : le tableau à trier contient trois types d'éléments, les bleus, les blancs et les rouges et on souhaite les trier par couleur. Tous les éléments d'une même couleur ne sont pas identiques, il ne suffit donc pas de les compter. Proposer un algorithme linéaire pour résoudre ce problème.

Exercice 4 : complexité du tri rapide dans le meilleur cas

On a vu en cours que le nombre de comparaisons e ectuées dans le meilleur des cas par le tri rapide sur un tableau de longueur n *est en* $O(n \quad n)$*. Ceci n'exclut pas* a priori *que la complexité soit meilleure, linéaire par exemple. Le but de cet exercice est de montrer que ce n'est pas le cas.*

Soit n un entier de la forme $\ ^{k} -$ et `T` un tableau de longueur n. Montrer que le tri rapide du tableau `T` nécessite exactement $(k - \ ) \ ^{k} +$ comparaisons si le choix du pivot est toujours fait de manière optimale. Conclure.

Exercice 5 : une amélioration du tri rapide : le tri « médiane de 3 »

L'e cacité du tri rapide repose essentiellement sur le choix d'un bon pivot : idéalement, il faudrait utiliser comme pivot l'élément médian *du tableau,* i.e. *tel que la partition produise deux sous-listes de même taille (à 1 près, naturellement, si le tableau est de longueur paire). Cependant, le calcul de cet élément est coûteux et ne permet pas d'améliorer la complexité du tri rapide en moyenne. On se contente donc d'augmenter la* probabilité *de choisir la médiane comme pivot.*

1. Dans le tri rapide classique d'un tableau de n éléments, quelle est cette probabilité ?

On va considérer la variante suivante : si le tableau est de longueur au plus 3, le trier directement. Sinon, choisir d'abord un échantillon de trois éléments (par exemple les trois premiers), puis prendre comme pivot la médiane de cet échantillon.

2. Appliquer cette variante au tableau `T` de l'exercice 1.

3. Donner un algorithme pour calculer la médiane de trois éléments. Montrer qu'il est de complexité optimale (en nombre exact de comparaisons).

Soit $p_i(n)$ la probabilité de choisir comme pivot le i^{e} plus petit élément du tableau (l'entier i est appelé *rang* de l'élément).

4. Combien valent $p_1(n)$ et $p_n(n)$?

5. En déduire le gain concernant la complexité dans le pire des cas.

6. Donner une formule exacte pour $p_i(n)$.

7. Quel est (asymptotiquement) le gain pour la probabilité de choisir la médiane comme pivot ?

8. (*) Si on considère qu'un pivot est « bon » si son rang i est compris entre $n/$ et $n/$, de combien améliore-t-on (asymptotiquement) la probabilité de choisir un bon pivot ?

La probabilité de choisir un bon pivot (ou exactement la médiane) augmente encore si ce pivot est choisi comme médiane de $k +$ *éléments avec* $k >$ *. Cependant, le calcul de la médiane de ces éléments devient alors plus coûteux... Une meilleure idée est en fait la variante dite « médiane 3-3 », qui repose sur le calcul de la médiane des médianes de 3 échantillons de 3 éléments.*

9. (**) Donner un algorithme pour déterminer la médiane de 5 éléments en 6 comparaisons.

Exercice 6 : hauteur de pile du tri rapide

Comme tout algorithme récursif, le tri rapide est sujet à des débordements de la pile si le nombre d'appels récursifs devient trop grand.

1. Quelle hauteur atteint la pile dans le pire cas ? dans le meilleur cas ? (**) en moyenne ?

2. Proposer une solution permettant de garantir une hauteur de pile maximale de $O(\quad n)$ dans tous les cas.

3. Quel est alors le meilleur cas en ce qui concerne la hauteur de pile ?