Devoir surveillé n° 1 - Corrigé

1 Recherche du minimum dans une partie du tableau

Question 1.1. *on peut prendre comme précondition « L est une liste de nombres entiers ou flottants et k est un entier compris entre 0 et $len(L) - 1$ »et comme postcondition « $L[imin]$ est le minimum de la liste $[L[k], L[k+1], \dots, L[n-1]]$ ».*

Question 1.2. *L'algorithme ne contient qu'une boucle for, dont le nombre d'exécutions est prédéfini, donc sa terminaison ne pose pas de problème.*

Question 1.3. · **Initialisation** *Si $k = i$, la liste se réduit à $[[L[k]]$ et ne contient qu'un élément et par définition de imin, $L[imin]$ est égal à $L[k]$, qui est bien le minimum de la liste.*

Hérédité *Supposons que $L[imin]$ soit le minimum de $[L[k], L[k+1], \dots, L[i]]$ à la fin de la boucle d'indice i. Au début de la boucle suivante, i prend la valeur $i + 1$. Deux cas sont alors possibles.*

Si $L[i+1] < L[imin]$, le minimum de $[L[k], L[k+1], \dots, L[i+1]]$ est $L[i+1]$ et imin prend la valeur de $i + 1$; $L[imin]$ est bien le minimum de la nouvelle liste.

Sinon, le minimum de $[L[k], L[k+1], \dots, L[i+1]]$ et imin restent inchangés, donc $L[imin]$ est le minimum de la nouvelle liste.

Donc P est un invariant de boucle.

Question 1.4. *Lorsque l'algorithme termine, i est égal à $n - 1$ donc, d'après $P(n-1)$ (invariant de boucle), $L[imin]$ est le minimum de la liste $[L[k], L[k+1], \dots, L[n-1]]$. Donc la postcondition est vérifiée. Donc l'algorithme est correct.*

Question 1.5. *La boucle est effectuée pour un indice i variant de $k+1$ à $n-1$, avec un test à chaque passage. On effectue donc $n - 1 + 1 - (k + 1)$ tests d'où $T(n, k) = n - k - 1$.*

2 Le tri par sélection

Question 2.1. *La boucle est effectuée quatre fois de suite et L prend quatre valeurs successives.*
$[4, 3, 5, 1, 2]$ *(état initial)*
$[1, 3, 5, 4, 2]$ *(après un passage dans la boucle)*
$[1, 2, 5, 4, 3]$ *(après deux passages dans la boucle)*
$[1, 2, 3, 4, 5]$ *(après trois passages dans la boucle)*
$[1, 2, 3, 4, 5]$ *(après le dernier passage, la liste est alors inchangée)*

Question 2.2. *On peut prendre comme précondition « L est une liste de nombres entiers ou flottants »et comme postcondition « L est triée ».*

Question 2.3. *L'algorithme ne contient qu'une boucle for et on sait déjà que la fonction indicemin termine, la terminaison de TriSelection ne pose donc pas de problème.*

Question 2.4. Initialisation

Avant le premier passage dans la boucle, la liste $[L[0], L[1], \dots, L[k]]$ est vide et est donc triée. Donc $Q(-1)$ est vérifiée.

Hérédité *Supposons qu'après le passage dans la boucle d'indice k, $Q(k)$ est vrai : la liste $[L[0], L[1], \dots, L[k]]$ est triée et que $L[k]$ est plus petit que le minimum de $[L[k+1], \dots, L[n-1]]$.*

Au début de la boucle suivante, k prend la valeur $k + 1$, puis $L[k+1]$ prend la valeur du minimum de $[L[k+1], \dots, L[n-1]]$. Cet élément est plus grand que $L[k]$ d'après la seconde partie de l'invariant en début de boucle, donc $[L[0], \dots, L[k], L[k+1]]$ est triée. De plus, les éléments restant dans $[L[k+2], \dots, L[n-1]]$ sont tous plus grand que $L[k+1]$, donc la seconde partie de l'invariant est vérifiée pour $k + 1$. Donc $Q(k+1)$ est vérifié.

Donc Q est un invariant de boucle.

Question 2.5. *À la fin du passage dans la dernière boucle, on a k égal à $n - 2$, on utilise donc $Q(n-2)$. D'après la première partie de l'invariant de boucle, $[L[0], L[1], \dots, L[n-2]]$ est trié. D'autre part, d'après la deuxième partie de l'invariant, $L[n-2]$ est plus petit que $L[n-1]$ donc $[L[0], L[1], \dots, L[n-1]]$ est triée et la postcondition est vérifiée. Donc l'algorithme est correct.*

Tournez la page S.V.P.

Question 2.6. *D'après la partie 1, à chaque passage dans la boucle, on effectue* $T(n,k) = n - k - 1$ *tests, d'où*

$$U(n) = \sum_{k=0}^{n-2} (n-k-1)$$

On peut faire le changement de variable $l = n - k - 1$

$$U(n) = \sum_{k=1}^{n-1} l = \frac{(n-1)n}{2}$$

Donc asymptotiquement, $U(n) \sim \frac{n^2}{2}$ *ou* $U(n) = O(n^2)$.