Note : sur cette correction, les bonnes réponses sont cochées directement sur le sujet au lieu de la feuille à détacher.

1:1

EXAMEN INF301 2017
Algorithmique & Prog. Impérative
11 janvier 2018

Durée : 2h
Tous documents interdits
Une feuille A4 R/V manuscrite autorisée

Toutes les questions à choix ont une **unique case** à cocher. ***Les réponses aux Questions à Choix sont à donner exclusivement sur la feuille de réponse (page 3 à détacher). Les réponses aux questions ouvertes sont à donner sur votre copie d'examen.***
Cet examen comporte 23 points mais sera noté sur 20. Le barème est indicatif et pourra être modifié.

Exercice 1 (Sous-séquences croissantes)

Soit $S = \langle s_0, s_1, \dots, s_{l-1}\rangle$ une séquence d'entiers de longueur l. Une **sous-séquence** de S est une séquence $S_{ij} = \langle s_i, s_{i+1}, \dots, s_j\rangle$ avec $0 \leqslant i \leqslant j \leqslant l-1$. On dit qu'une sous-séquence S_{ij} est **croissante** si pour tout k tel que $i \leqslant k < j$, on a $s_k \leqslant s_{k+1}$. De plus, S_{ij} est **maximale** si $s_{i-1} > s_i$ (ou $i = 0$), et si $s_j > s_{j+1}$ (ou $j = l-1$) (on ne peut pas l'« aggrandir »).

Question 1 Soit $S = \langle 6, 2, 4, 5, 3, 9, 7, 1, 8, 10\rangle$, quelle sous-séquence de S est maximale et croissante ? [1 pt]

A $\langle 6\rangle$ B $\langle 3, 9\rangle$ C $\langle 1, 8, 10\rangle$ D $\langle 2, 4, 5\rangle$ E *Aucune réponse correcte.* F *Toutes les réponses sont correctes.* G *Manque de données dans l'énoncé.* H *La question est absurde.*

On considère que S est donnée sous forme de tableau avec longueur explicite. La longueur maximale est `LMAX = 15`. Soit une fonction ssmax(S, i), qui renvoie le plus grand indice j tel que S_{ij} soit une sous-séquence croissante.

Question 2 Que vaut ssmax(S, 2) pour $S = \langle 6, 2, \overset{2}{4}, \overset{3}{5}, 3, 9, 7, 1, 8, 10\rangle$? [1 pt]

A 2 B 4 C 3 D 5 E *Aucune réponse correcte.* F *Toutes les réponses sont correctes.* G *Manque de données dans l'énoncé.* H *La question est absurde.*

Question 3 Quelle propriété doit avoir i pour que, si j = ssmax(S,i), S_{ij} soit une sous-séquence maximale croissante ? [1 pt]

A $i < l-1$ et $s_i < s_{i+1}$
B $i < j$ ou $s_{i-1} < s_i$
C $i > 0$ et $s_i > s_{i+1}$
D $i = 0$ ou $s_{i-1} > s_i$
E *Aucune réponse correcte.*
F *Toutes les réponses sont correctes.*
G *Manque de données dans l'énoncé.*
H *La question est absurde.*

Question 4 Écrire l'algorithme de la fonction ssmax. [2 pts]

→ page suivante

On cherche à créer Ec, ensemble contenant toutes les sous-séquences maximales croissantes de S. On utilise pour cela ssmax dans l'algorithme ci à droite. On suppose que Ec est un ensemble sous forme de liste chainée (dans chaque cellule, le champ `valeur` est une référence vers la tête d'une sous-séquence, c'est-à-dire une cellule (les sous-séquences sont des listes chaînées d'entiers)). Pour les besoins de l'exercice, l'ordre d'ajout dans Ec est important même si c'est un ensemble.

toutes_ss(S) : ensemble de sous-séquences
  $Ec \leftarrow \varnothing$ } O(1)
  $i \leftarrow 0$
  tant que $i < S$.longueur **faire** l ×
    $j \leftarrow$ ssmax(S, i) O(l)
    ajouter_fin(Ec, S, i, j) O(l)
    $i \leftarrow j$; O(1)
  retourner Ec

La fonction ajouter_fin crée une nouvelle liste chaînée contenant $\langle s_i, s_{i+1}, \dots, s_j\rangle$, et l'ajoute comme sous-séquence à la fin de Ec.

On veut que si l'on appelle toutes_ss(S) avec $S = \langle 6, 2, 4, 5, 3, 9, 7, 1, 8, 10\rangle$, cela crée l'ensemble Ec :

$$\{ \quad \langle 6\rangle, \quad \langle 2,4,5\rangle, \quad \langle 3,9\rangle, \quad \langle 7\rangle, \quad \langle 1,8,10\rangle \quad \}$$

Cependant, pour l'instant, l'algorithme ne s'arrête pas et crée à l'infini l'ensemble suivant :

$$\{ \quad \langle 6\rangle, \quad \langle 6\rangle, \quad \langle 6\rangle, \quad \langle 6\rangle, \quad \langle 6\rangle, \quad \langle 6\rangle, \dots \quad \}$$

$O(\ell)$ ssmax (S, i)

$j \leftarrow i$

$\times \ell$ tant que $j+1 < S.$longueur et $S.tab[j] \leq S.tab[j+1]$

$O(1)$ $j \leftarrow j+1$

retourner j

Note : ⚠ à l'ordre des conditions

Ajouter_fin : principe 1) on crée une liste chaîné avec toutes les valeurs S_i à S_j en commençant par S_j et par ajouts successifs en tête

2) on rajoute cette liste comme contenu d'une cellule ajoutée en queue de Ec

Pour Ec, on utilise une structure avec un champ "tête" et un champ "queue"

$O(\ell)$ ajouter_fin (Ec, S, i, j)

$O(\ell)$ $\ell \leftarrow$ cree_ssliste (S, i, j)

$O(1)$

- cel $\leftarrow$ nouvelle cellule
- cel.valeur $\leftarrow \ell$
- cel.suivant $\leftarrow$ Nil
- si Ec.tete = Nil
  - Ec.tete $\leftarrow$ cel
- sinon
  - Ec.queue.suivant $\leftarrow$ cel
- Ec.queue $\leftarrow$ cel

$O(\ell)$ cree_ssliste (S, i, j)

tete $\leftarrow$ Nil

$\ell \times$ tant que $i \leq j$

$O(1)$

- cel $\leftarrow$ nouvelle cellule
- cel.valeur $\leftarrow$ S.tab[j]
- cel.suivant $\leftarrow$ tete
- tete $\leftarrow$ cel
- $j \leftarrow j-1$

retourner tete

Question 5 Que faut-il changer à l'algorithme pour obtenir le comportement attendu ? [2 pts]

- A utiliser « `pour i de 0 à S.longueur-1` »
- B initialiser avec $i \leftarrow 1$
- C changer $i \leftarrow j$ en $i \leftarrow j + 1$
- D initialiser Ec avec une cellule fictive
- E *Aucune réponse correcte.*
- F *Toutes les réponses sont correctes.*
- G *Manque de données dans l'énoncé.*
- H *La question est absurde.*

Question 6 Expliquez les étapes importantes de l'algorithme ajouter_fin, puis écrivez la fonction avec les détails bas-niveau. Vous devez expliciter en particulier toutes les modifications de liens de chaînage et création de cellules. Si vous utilisez des fonctions auxiliaires, donnez également leurs algorithmes. [4 pts]

Question 7 Quelle est la complexité de l'algorithme toutes_ss ? Justifier. [2 pts]

Soit la fonction récursive mystère ci à droite.

Question 8
On appelle mystère avec comme paramètres a et b les têtes des séquences $\langle 2,4,5,6\rangle$ et $\langle 3,9\rangle$. Dessinez **au dos de la feuille de réponses** (page 4) les listes chaînées représentant ces séquences, ainsi que les modifications de liens de chaînage effectuées par l'algorithme mystère. (Conseil : listez les appels récursifs effectués.)

Quelle est la valeur renvoyé par l'algorithme pour cet exemple ? [2 pts]

```
mystère (a, b : references de Cellule) : reference de Cellule
  si a = Nil alors
    retourner b
  si b = Nil alors
    retourner a
  si a.val < b.val alors
    a.suivant← mystère (a.suivant, b)
    retourner a
  sinon
    b.suivant← mystère (a, b.suivant)
    retourner b
```

L'algorithme mystère est utilisé dans l'algorithme ci à droite.

```
pain_au_chocolat (S)
  Ec ← toutes_ss(S)
  tant que Ec.tête.suivant≠ Nil faire
    Ec.tête.val ← mystère (Ec.tête.val,
      Ec.tête.suivant.val)
    tmp ← Ec.tête.suivant
    Ec.tête.suivant← Ec.tête.suivant.suivant
    libérer (tmp)
  retourner Ec.tête
```

a Durant l'exécution de l'algorithme pain_au_chocolat, avec $S = \langle 6,2,4,5,3,9,7,1,8,10\rangle$, mystère est appelée **récursivement** au bout d'un certain temps avec comme arguments $a = \langle 4,5,6\rangle$ et $b = \langle 9\rangle$ (a et b sont des références vers les cellules contenant 4 et 9).

Question 9 Dessinez **au dos de la feuille de réponses** le plus précisément possible l'état de la mémoire **au moment de cet appel**. Détaillez en particulier la pile d'appels récursifs (fonctions pain_au_chocolat et mystère), les variables locales, les structures de données et les liens de chainages. [5 pts]

Question 10 Pour quelle séquence S l'algorithme pain_au_chocolat va-t-il « planter » (erreur d'accès mémoire) ? [1 pt]

- A $S = \langle 6\rangle$
- B $S = \langle\rangle$
- C $S = \langle 2,4,9\rangle$
- D $S = \langle 1,2,3,4,5,6,7,8,9\rangle$
- E *Aucune réponse correcte.*
- F *Toutes les réponses sont correctes.*
- G *Manque de données dans l'énoncé.*
- H *La question est absurde.*

Question 11 Que fait l'algorithme pain_au_chocolat ? Justifiez (Expliquez ici la *fonctionnalité*, c'est-à-dire, le résultat et comment il est produit.) [2 pts]

Complexité de Toutes_ss :

à première vue on pourrait croire $O(\ell^2)$,
cependant la boucle "tant que" fait "avancer"
j plus rapidement : à chaque fois j se
retrouve en début de sous séquence max suivante
$\Rightarrow$ au final on parcourt chaque ssmax 2 fois
exactement : 1 pour ssmax, et 1 pour ajouter-fin
$\Rightarrow$ au total 2 parcours de la séquence

$$O(2 \times \ell) = O(\ell)$$

"Pain au chocolat" trie la séquence S par ordre "croissant" (ha ha !)
$\hookrightarrow$ elle commence par partitionner S en sous-séquences
croissantes. La fonction mystère fusionne deux séquences
croissante en une séquence croissante, et pain au chocolat
fusionne 2 à 2 toutes les séquences de Ec jusqu'à n'en
obtenir plus qu'une par appels successifs à mystère.

Feuille de réponses

Noircissez entièrement les cases.

Les réponses aux QCM sont à donner exclusivement sur cette feuille : les réponses données sur les feuilles précédentes ne seront pas prises en compte.

☐0	☐0	☐0	☐0	☐0	☐0	☐0	☐0
☐1	☐1	☐1	☐1	☐1	☐1	☐1	☐1
☐2	☐2	☐2	☐2	☐2	☐2	☐2	☐2
☐3	☐3	☐3	☐3	☐3	☐3	☐3	☐3
☐4	☐4	☐4	☐4	☐4	☐4	☐4	☐4
☐5	☐5	☐5	☐5	☐5	☐5	☐5	☐5
☐6	☐6	☐6	☐6	☐6	☐6	☐6	☐6
☐7	☐7	☐7	☐7	☐7	☐7	☐7	☐7
☐8	☐8	☐8	☐8	☐8	☐8	☐8	☐8
☐9	☐9	☐9	☐9	☐9	☐9	☐9	☐9

⟵ codez votre **numéro d'anonymat** ci-contre, et **ré-inscrivez le** ci-dessous.
↓

Numéro d'**anonymat** :

....

Question 1 : ☐A ☐B ☐C ☐D ☐E ☐F ☐G ☐H

Question 2 : ☐A ☐B ☐C ☐D ☐E ☐F ☐G ☐H

Question 3 : ☐A ☐B ☐C ☐D ☐E ☐F ☐G ☐H

Question 4 : .... ssmax algo ☐W ☐II ☐I ☐P ☐PP ☐C *Réservé*

Question 5 : ☐A ☐B ☐C ☐D ☐E ☐F ☐G ☐H

Question 6 : .... ajouter fin ☐W ☐II ☐I ☐P ☐PP ☐C *Réservé*

Question 7 : .... ssmax cpx ☐W ☐II ☐I ☐P ☐PP ☐C *Réservé*

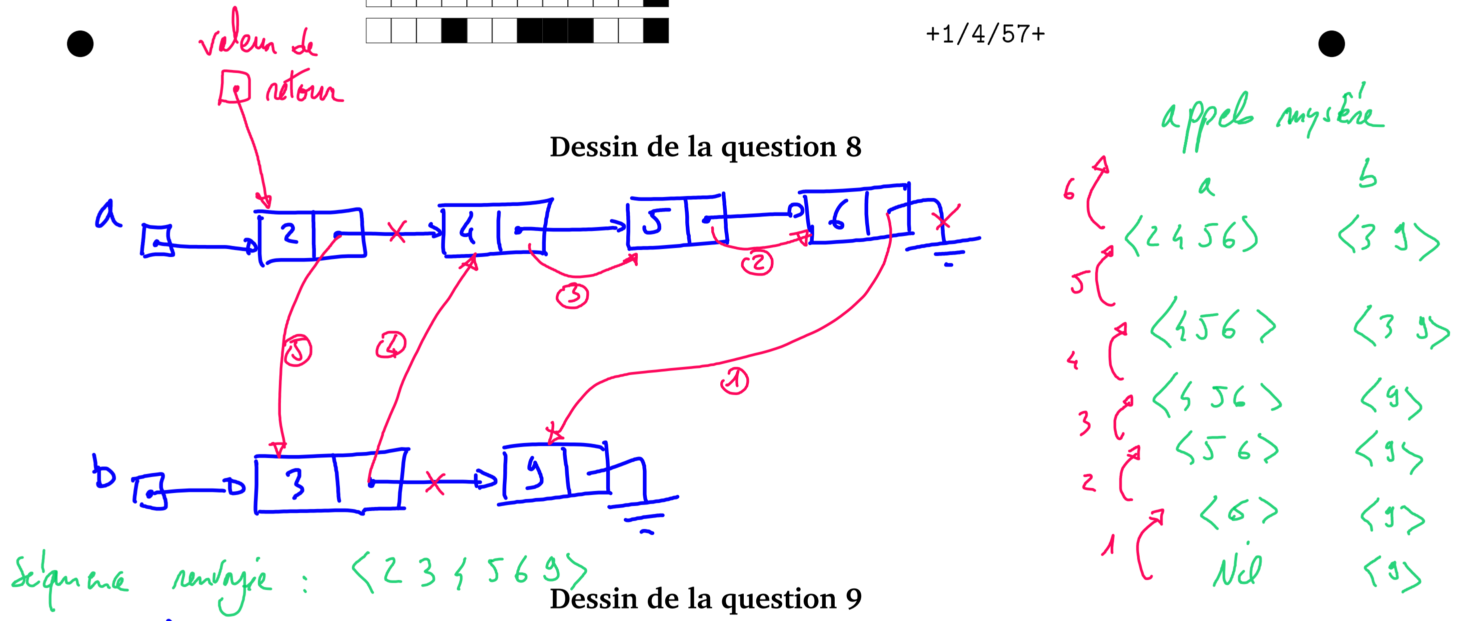
Question 8 : .... mystere ☐W ☐II ☐I ☐P ☐PP ☐C *Réservé*

Question 9 : .... dessin ☐W ☐II ☐I ☐P ☐PP ☐C *Réservé*

Question 10 : ☐A ☐B ☐C ☐D ☐E ☐F ☐G ☐H

Question 11 : .... chocolat ☐W ☐II ☐I ☐P ☐PP ☐C *Réservé*

Dessin de la question 8



Dessin de la question 9