M2 CCI, M1 GEOMAS – Algorithmique – Devoir surveillé

Durée 1h, sans documents **25 octobre 2022**

NE PAS RECOPIER les énoncés des questions. Ne pas perdre de temps à un soin excessif de la présentation. Les questions sont indépendantes. Le barème est indicatif.

Q1 Vérification des types dans une expression algébrique [3 points]

Pour chacune des expressions **E1**, **E2** et **E3** ci-dessous, donner les **contraintes de types** que doivent respecter les noms y apparaissant, puis le **type de l'expression**, dans l'hypothèse où ces contraintes sont respectées (le symbole & dénote l'opérateur de concaténation).

E1 : si ((a>b) ou m) alors d sinon "ab" & c
E2 : "ab" & a & "ba" & [b]
E3 : a = (si w et x=y alors z sinon x+10)

Q2 Relation sur des dates [3 points]

On représente des dates par 3 entiers correspondant au jour, au mois et à l'année.
Exemple : <25,10,2022> représente le 25 octobre 2022.

— Donner la définition d'un type nommé **Date** correspondant à cette représentation.

On spécifie la fonction :
InfDate : fonction (D1, D2 : Date) → booléen
{vrai si et seulement si D1 est strictement inférieure (dans l'ordre chronologique) à D2}.

— Donner une réalisation de la fonction **InfDate**.

Q3 Types et statuts de paramètres [5 points]

On considère l'extrait d'algorithme suivant :

• • • • • • • • *{compléter, 3 lignes attendues}* • • • • • • • • • • • • • • • • Ajouter : action (• • • • • • • •) *{compléter le profil de l'action}* *{Ajouter (X, T, L, i, Z) ajoute l'entier X en l'insérant en position i de la séquence T de longueur L, si c'est possible; dans ce cas, à l'état final, Z a la valeur vrai et l'entier X est ajouté dans la séquence en position i.* *Sinon, à l'état final, la séquence est inchangée et Z a la valeur faux.* *Exemple :* *Ajouter 30 en position 2 dans la séquence [1 ;2 ;3 ;4 ;5] donne [1 ;30 ;2 ;3 ;4 ;5]* *Il n'est pas possible d'ajouter 30 en position 2 dans [1 ;2 ;3 ;4 ;5] si le tableau contenant la séquence est de taille 5.}*

— **Compléter** les • • • • • • • précédant la spécification de l'action **Ajouter**, de manière à fournir le lexique nécessaire pour représenter des séquences d'entiers sous forme contiguë dans un tableau avec longueur explicite.
— **Compléter la spécification** de l'action **Ajouter** (type et statut des paramètres).
— Donner une **réalisation** de l'action **Ajouter.**

Q4 Transposée d'une matrice carrée [4 points]

On traite une matrice carrée de caractères d'ordre **n**, nommée **MCE** :

n : un entier > 0 ; MCE : un tableau sur [1…n] de tableaux sur [1…n] de caractères

(i) Algorithme de transposition

On rappelle que la transposée **M'** d'une matrice carrée **M** est définie par $\forall i, \forall j$, $\mathbf{M'_{i,j} = M_{j,i}}$.

— **Donner la réalisation d'un algorithme** (**pas de fonction ou d'action**) qui réarrange la matrice donnée **MCE** de telle sorte qu'à l'état final la valeur de **MCE** soit la transposée de sa valeur initiale. On supposera que **n** et **MCE** sont données et contiennent des valeurs. L'algorithme **doit procéder** par échanges de valeurs dans **MCE, sans tableau supplémentaire**.

Lors de la correction, on appréciera les solutions qui minimisent le nombre d'échanges.

(ii) Analyse quantitative de l'algorithme.

— Donner en fonction de **n**, le nombre d'échanges de valeurs dans **MCE**, qui sont exécutés par l'algorithme.

Q5 Nombre de Y après X [5 points]

Une séquence d'entiers est donnée sous forme contiguë dans un tableau avec longueur explicite.

On donne le lexique suivant :

Lmax : constante 100 de type entier > 0 *{longueur maximum de la séquence}*
T : tableau sur [1…Lmax] d'entier
L : entier sur [0…Lmax]
X, Y : entier

(i) Compréhension de la représentation

Soit l'expression $\mathbf{T_{L+1}}$: **que peut-on en dire** (correction, valeur) ?

(ii) Réalisation

Pour **X** et **Y** entiers donnés, on veut afficher le nombre d'éléments de valeur **Y** situés, dans une séquence, strictement après le premier exemplaire de valeur **X**. Si **X** n'appartient pas à la séquence, un message spécial est affiché. On précise que **X** peut être égal à **Y** (on ne compte alors que le nombre d'éléments après la première occurrence de la valeur).

— **Donner la réalisation d'un algorithme (pas de fonction ou d'action)** qui, étant donnés deux entiers **X** et **Y** (dont les valeurs sont supposées connues) et, étant donnée une séquence de longueur **L** représentée dans un tableau **T** (dont les valeurs sont aussi supposées connues) **affiche le nombre d'éléments** de valeur **Y** situés strictement après le premier exemplaire de valeur **X.**

Exemple :

Si les données sont **:** $\mathbf{T_{[1...L]}}$ = [6 ;4 ;7 ;**3** ;**7** ;3 ;5 ;**7**]**, L** = 8, **X**=3, **Y** =7 alors le résultat attendu est **2** car il y a deux **7** après le premier exemplaire de la valeur **3**.

Q6 Question subsidiaire : Invariants d'itérations (bonus jusqu'à 4 points)

On considère l'algorithme suivant :

```
n : constante de type entier > 0
T : tableau sur [1…n] d'entier
B : booléen
B ← faux; i ← 1
{*P1*}
tant que i ≠ n+1
        {*P2*}
        B ← B ou Ti < 0
        i ← i + 1
        {*P3*}
{*P4*}
```

(i) Candidats invariants

Pour chacune des assertions suivantes, indiquer s'il s'agit d'un invariant de l'itération. Justifier la réponse.

A1 : *$n > 1$ et $T_2 < 0$* **A2** : *$n > 1$ et $T_i < 0$* **A3** : *B a la valeur vrai*

(ii) Effet de l'algorithme

Soit l'assertion **A4** : *B a la valeur vrai $\Leftrightarrow$ au moins un élément de $T_{[1..n]}$ est négatif*

— Proposer une assertion **A5** invariant de l'itération permettant de déduire que **A4** est vérifiée en ***P4***.

— Justifier la réponse (preuve d'invariance de **A5**, utilisation de **A5** pour déduire le résultat).

M2 CCI, M1 GEOMAS – Algorithmique

AL – Quick 1 : des exemples de solutions **25 octobre 2022**

Q1 Vérification des types dans une expression algébrique [3 points, 1 point par expression]

E1 : **a** et **b** sont de même type (entier ou réel), **m** est de type booléen, **d** et **c** sont de type texte. **E1** est de type texte.
E2 : **a** est de type texte, **b** de type caractère. **E2** est de type texte non vide.
E3 : **a, x**, **y** et **z** sont de type entier, **w** de type booléen. **E3** est de type booléen.

Q2 Relation sur des dates [3 points]

Date : type <j : entier sur [1..31], m : entier sur [1..12], a : entier>

InfDate (D1, D2) :
retour : D1.a < D2.a
    ou alors (D1.a = D2.a et D1.m < D2.m)
    ou alors (D1.a = D2.a et D1.m = D2.m et D1.j < D2.j)

Q3 Types et statuts de paramètres

(i) [1,5 points]
Lmax : constante de type entier > 0 *{longueur maximum des séquences représentées}*
TabEnt : type tableau sur [1…Lmax] d'entier
Ent : type entier sur [0…Lmax]

(ii) [3,5 points]
Ajouter : action (donnée X : entier ; donnée-résultat T : TabEnt, L : Ent ; donnée i : Ent ; résultat Z : booléen)

Ajouter(X, T, L, i, Z) :
  Z ← L < Lmax et i<=L+1
  si Z alors *{on a la place d'ajouter un élément et on insère à une position contigue}*
    pour k allant de L+1 à i+1, pas -1
      $T_k \leftarrow T_{k-1}$
    L ← L+1
    $T_i \leftarrow X$

Q4 Transposée d'une matrice [4 points]

Pour réaliser la transposition, on procède par une succession d'échanges de valeurs de MCE situées dans des positions symétriques par rapport à la diagonale. Une première solution consiste à parcourir entièrement toutes les cases de la matrice. Il en découle $\mathbf{n^2}$ échanges. Il suffit toutefois de ne considérer que les cases de la demi-matrice inférieure (sans la diagonale).

(i)

C : un caractère *{pour réaliser l'échange de deux valeurs}*
i parcourant [2…n]
  j parcourant [1…i-1]
    $C \leftarrow MCE_{i,j}$; $MCE_{i,j} \leftarrow MCE_{j,i}$; $MCE_{j,i} \leftarrow C$

(ii)

Le corps de l'itération interne réalise un échange. A l'étape **i** de l'itération externe, il y en a **i-1**. Comme **i** parcourt l'intervalle [2…n], il y en a au total : 1+2+…+i+…+(n-1), soit **n*(n-1)/2**.

Q5 Nombre de Y après X [5 points]
(i) [1 point]

$\mathbf{T_{L+1}}$ est :
correcte si L < Lmax. Mais sa valeur n'est pas pertinente pour la séquence représentée.
incorrecte si L = Lmax.

(ii) [4 points]

```
i : entier sur [1.. Lmax]
nb : entier

i ← 1                                                   {recherche de X}
tant que i ≠ L+1 et puis Ti ≠ X
    i ← i + 1
si i = L+1 alors Écrire ("premier nombre absent")            {si absent}
sinon
    nb ← 0                                                   {si présent}
    pour j allant de i+1 a L
        si Tj = Y alors nb ← nb+1
    Écrire(nb)
```

Q6 Subsidiaire : À propos d'invariants d'itérations (4 points max)
(i)

- **A1** (*$n > 1$ et $T_2 < 0$*) est un invariant : le corps de l'itération ne modifie ni **n**, ni **T**. Ainsi, l'hypothèse ***A1*** en **P2** implique que ***A1*** est vraie en **P3**.
- **A2** (*$n > 1$ et $T_i < 0$*) n'est pas un invariant : soit **i0** la valeur de **i** en **P2**. Supposons **A2** vraie en **P2** : *$n > 1$ et $T_{i0} < 0$.* En **P3**, la valeur de **i** est **i0+1**. Il suffit que $\mathbf{T_{i0+1}}$ soit positif ou nul pour que ***A2*** ne soit pas vraie en **P3**.
- **A3** (*B a la valeur vrai)* est un invariant : dans le corps de l'itération **B** est modifiée par l'instruction **B ← B ou $\mathbf{T_i < 0}$**. L'hypothèse *B a la valeur vrai* en **P2** implique que *B a la valeur vrai* en **P3** (propriété de l'opérateur **ou**).

(ii)

Soit l'assertion **A5** suivante (déduite de **A4** par généralisation) :

A5 : *B a la valeur vrai ⇔ au moins un élément de $\mathbf{T}_{[1...i-1]}$ est négatif.*

A5 est un invariant de l'itération :

Supposons **A5** vraie en **P2** et soit **i0** la valeur de **i** en **P2**. En **P3**, **i** a la valeur **i0+1**. On considère deux cas :

- **B** a la valeur faux en **P2**. On considère à nouveau 2 cas en **P2** :
  - $\mathbf{T_{i0} < 0}$: en **P3**, on a bien un élément négatif dans $\mathbf{T_{[1..i0]}}$ et **B** a la valeur **vrai**
  - $\mathbf{T_{i0} >= 0}$: en **P3** il n'y a aucun élément négatif dans $\mathbf{T_{[1..i0-1]}}$ (d'après **A5**) et donc non plus dans $\mathbf{T_{[1..i0]}}$ et **B** a la valeur **faux**.
- **B** a la valeur vrai en **P2** : il existe au moins un élément négatif dans $\mathbf{T_{[1...i0-1]}}$ (d'après **A5**). En **P3**, quelle que soit la valeur de $\mathbf{T_{i0}}$, il existe au moins un élément négatif dans $\mathbf{T_{[1...i0]}}$ et **B** a la valeur **vrai**.

Ainsi dans tous les cas, si **A5** est vraie en ***P2***, **A5** est vraie en ***P3***.

A4 est vraie en ***P4*** : **A5** est vraie en **P1** et **A5** est un invariant. Par ailleurs, en sortie de boucle l'assertion **i=n+1** est vraie. On peut conclure qu'en **P4** l'assertion suivante est vraie :

$i = n+1$ et (B a la valeur vrai ⇔ au moins un élément de $\mathbf{T}_{[1...i-1]}$ est négatif.).

On en déduit : *B a la valeur vrai ⇔ au moins un élément de $\mathbf{T}_{[1...n]}$ est négatif.*