



version après correction du barème

UE INF120

Informatique – Devoir Surveillé - Grenoble

Durée 2h, sans documents, appareils électroniques strictement interdits

19 Mars 2004

Il est demandé de respecter le lexique des énoncés. Le barème donné est indicatif. Les exercices sont indépendants. **REPENDRE UNIQUEMENT SUR LA FEUILLE FOURNIE EN ANNEXE.**

Les solutions seront données en utilisant la notation fonctionnelle du cours et des travaux dirigés (et non CAML).

Rappel des notations concernant les séquences

- Constructeurs : [], S•e, e o S ; testeur et sélecteurs : EstVide?, début, dernier, premier, fin.
- La concaténation est notée par le symbole &.

1. Vérification des types des noms d'une expression algébrique [3,5 points]

On s'intéresse ici à la vérification des types des noms qui apparaissent dans une expression algébrique par rapport aux spécifications des opérations qu'elles mettent en jeu. Par exemple, pour que l'expression **(3 o a)•premier(b)** soit correcte du point de vue des types, les contraintes suivantes doivent être respectées: **a** doit être de type séquence d'entiers et **b** de type séquence non vide d'entiers. Si ces conditions sont respectées, l'expression **(3 o a)•premier(b)** est de type séquence non vide d'entiers.

Q1.

— Pour chacune des expressions nommées E1, E2, E3 et E4 données ci-dessous, donner les contraintes de types que doivent respecter les noms y apparaissant, puis le type de l'expression si ces contraintes sont respectées. Indiquer si l'expression est incorrecte le cas échéant.

E1 : (si (3<a) alors (3>7) sinon (a=b)) et puis (a>b)

E2 : soit c = <23,"Janvier",2003> dans d o fin(c)

E3 : premier(fin(3 o e)) o f

E4 : fin([<g,5>,<h=2,7>])

2. Classement de trois valeurs [4 points]

On considère les définitions suivantes :

CoupleE : le type <un entier, un entier>

TripletE : le type <un entier, un entier, un entier>

PO2E : un CoupleE → un CoupleE

{ PO2E(<a,b>) est la permutation ordonnée du couple a,b, c'est à dire le couple <a,b> si b est strictement supérieur à a, et le couple <b,a> si a est strictement supérieur à b.

Précondition: a et b sont deux entiers distincts.

Par exemple PO2E(<-5,9>) = <-5,9> et PO2E(<7,3>) = <3,7> }

PO3E : un TripletE → un TripletE

{ PO3E(<a,b,c>) est la permutation ordonnée du triplet <a,b,c>, c'est à dire le triplet dans lequel les entiers a,b et c apparaissent en ordre croissant.

Précondition: a, b et c sont trois entiers distincts deux à deux.

Par exemple PO3E(<2,-5,9>) = <-5,2,9> et PO3E(<1,5,7>) = <1,5,7> }

Q2. Réalisation de PO2E

— Compléter la réalisation suivante de PO2E.

PO2E(C) : soit = dans

Q3. Réalisation de PO3E

(i) 1^{ère} version.

Le principe de la première version est de construire la permutation ordonnée des deux premières valeurs, puis de construire le résultat par insertion de la troisième valeur au bon endroit.

— Compléter la réalisation suivante en respectant ce principe de solution :

PO3E(<x,y,z>) : soit = dans

(ii) 2^{ème} version.

Le principe de la deuxième version consiste à chercher le minimum des 3 valeurs et de construire le résultat en utilisant la permutation ordonnée des deux valeurs restantes. On propose la réalisation suivante de cette fonction selon ce principe :

POS3E(<x,y,z>) : selon x,y,z

x<y et x<z : <x, PO2E(z,y)>

y<x et y<z : <y, PO2E(x,z)>

sinon : <z, PO2E(x,y)>

— Expliquer pourquoi cette réalisation est incorrecte. Donner un jeu d'essai montrant le problème.

— Réécrire la deuxième ligne pour corriger le problème concernant cette ligne :

x<y et x<z :

3. A propos d'anagrammes

Deux mots sont des *anagrammes* l'un de l'autre, s'ils comportent **exactement** les mêmes lettres, éventuellement dans un ordre différent. Deux mots sont de *vrais anagrammes*, s'ils sont différents et sont des anagrammes. On introduit les définitions suivantes. :

Lettre : le type caractère *{restreint aux lettres majuscules de l'alphabet, sans accent}*

Mot : le type séquence de Lettres

EstAna? : deux Mots → un booléen

{EstAna?(M1, M2) vrai \hat{U} les mots M1 et M2 sont anagrammes. Tout mot est anagramme de lui-même. Par ailleurs EstAna?([], []) = vrai.

Par exemple EstAna?("SEL", "LES") = vrai, EstAna?("ARBRE", "BARRE") = vrai.

Par contre EstAna?("LASSE", "SALLE") = faux car les deux mots n'ont pas le même nombre de L ni le même nombre de S. }

EstVraiAna? : deux Mots → un booléen

{EstVraiAna?(M1, M2) vrai \hat{U} les mots sont différents mais sont des anagrammes.

Par exemple EstVraiAna?("SEL", "LES") = vrai, mais EstVraiAna?("UN", "UN") = faux }.

Pour réaliser EstAna? et EstVraiAna?, trois fonctions intermédiaires sont introduites.

Ap? : une Lettre, un Mot → un booléen

{Ap?(L, M) a la valeur vrai \hat{U} L est une lettre de M.

Par exemple Ap?("A", "FAUX")=vrai, Ap?("F", "VRAI")=faux}

Moins : une Lettre, un Mot → un Mot

{Moins(L, M) est le mot déduit de M en lui enlevant le premier exemplaire de L. Si L n'appartient pas à M, Moins(L, M) = M. Par exemple Moins("I", "POINT")="PONT",

Moins("I", "ICI") = "CI", Moins("I", "PONT")="PONT" }

EgMots? : deux Mots → un booléen

{EgMots?(M1, M2) vrai \hat{U} les mots M1 et M2 sont identiques.

Par exemple EgMots?("", "SEL")=faux, EgMots?("SEL", "SEL")= vrai }

Q4. [3,5 points]

(i) Définition récursive de EgMots?.

— Donner une définition récursive pour la fonction EgMots?.

(ii) Réalisation de Ap?.

— Donner une définition récursive pour la fonction Ap?.

— Donner une réalisation récursive de Ap en utilisant uniquement les opérateurs suivant: "et puis", "premier", "fin", "EstVide?", "ou alors", "non".

— Ecrire une phrase en français traduisant le plus fidèlement possible le fonctionnement de la réalisation proposée ci-dessus.

(iii) Définition récursive de Moins.

— Donner une définition récursive pour la fonction Moins.

Q5. [1,5 points]

(i) Définition de la fonction EstAna?

— Donner une définition récursive de la fonction EstAna?

(ii) Réalisation de la fonction EstVraiAna?

— Donner une réalisation de la fonction EstVraiAna en utilisant les opérateurs "non" et "et puis"

Q6. Ensemble des vrais anagrammes d'un mot dans une phrase [2 points]

On désire déterminer l'ensemble des vrais anagrammes d'un mot dans une phrase. On définit pour cela le lexique suivant.

Phrase : le type séquence de Mots *{non vides}*

PH0 : la phrase ["UN", "ECRAN", "NU", "EN", "NACRE"]

PH1 : la phrase ["MOUDRE", "UNE", "BARRE", "AVEC", "UN", "ARBRE", "FAIT", "DU", "REMOUD", "SI", "ELLE", "EST", "MORDUE", "DANS", "UNE", "CAVE", "PAR", "DES", "AIGLES", "DE", "GLAISE"]

PH2 : la phrase ["UNE", "LOUPE", "NUE", "SUR", "UN", "CRANE", "NU", "DE", "NACRE", "PRES", "DE", "UN", "INGRAT", "ECRAN", "PLAT", "QUE", "GARNIT", "UNE", "POULE", "DE", "GRANIT"]

EnsembleDeMots : le type séquence de Mots *{distincts et ordonnés par ordre alphabétique croissant. Par exemple les valeurs EM1, EM2 et EM3 définies ci-dessous sont des EnsembleDeMots. Par contre ["DEUX", "DEUX", "UN"] n'est pas un ensemble de mots car les éléments ne sont pas tous distincts. De même ["UN", "DEUX", "TROIS"] n'est pas un EnsembleDeMots car l'ordre alphabétique n'est pas respecté. De même PH0, PH1 et PH2 ne sont pas non plus des ensembles de mots. }*

EM1 : l'EnsembleDeMots ["DEUX", "QUATRE", "TROIS", "UN"]

EM2 : l'EnsembleDeMots [["I"], "II", "IV", ["V"]]

EM3 : l'EnsembleDeMots ["FOUR", "ONE", "TWO"]

VAnaDUnMot : un Mot, une Phrase → un EnsembleDeMots

{VAnaDUnMot(M,S) est l'ensemble des mots de S qui sont de vrais anagrammes de M.

Par exemple VAnaDUnMot("OEUFs", ["NEUF", "EUX", "OEUFs"]) = [],

VAnaDUnMot ("REMOUD", PH1) = ["MORDUE", "MOUDRE"] }

InsèreMot : un Mot, une EnsembleDeMots → un EnsembleDeMots

{InsèreMot(M,E) est l'ensemble de mots E auquel a été ajouté M s'il n'était pas présent.

Par exemple InsèreMot("UN", ["DEUX", "TROIS"]) = ["DEUX", "TROIS", "UN"]

InsèreMot("DEUX", ["DEUX", "TROIS"]) = ["DEUX", "TROIS"],

InsèreMot("CRANE", ["ECRAN", "NACRE"]) = ["CRANE", "ECRAN", "NACRE"] }

— Donner une définition récursive de la fonction VAnaDUnMot en utilisant la fonction InsèreMot.

Q7. A propos d'ensembles ordonnés d'ensembles ordonnés de mots [2 points]

Dans cette question, on introduit un lexique qui sera utilisé dans la question suivante. Il s'agit de représenter des ensembles ordonnés d'ensembles ordonnés de mots pour structurer la réponse fournie par la fonction LesVAnagrammes définie dans la question suivante.

EnsembleDEnsemblesDeMots : le type séquence d'EnsembleDeMots

{Chaque élément de la séquence doit être un (1) EnsembleDeMots non vide. Les

ensembles de mots présents dans la séquence doivent être (2) distincts deux à deux et

(3) ordonnés par ordre alphabétique strictement croissant par rapport au premier mot de

chaque EnsembleDeMots ; (4) les premiers mots de chaque EnsembleDeMots étant

distincts deux à deux. Par exemple la valeur suivante [EM1,EM3,EM2] est un EnsembleDEnsemblesDeMots car chaque élément de cette séquence est un EnsembleDeMots et que les premiers mots de EM1, EM3 et EM2, à savoir "DEUX", "FOUR", ["I"] sont rangés en ordre alphabétique croissant.

Par contre [["TETE", "POULE"], ["TETE"], ["CRANE", "OEUF"], []] n'est pas un EnsembleDEnsemblesDeMots. En effet ["TETE", "POULE"] n'est pas un EnsembleDeMots (propriété (1) non satisfaite); le séquence des premiers mots ["TETE", "TETE", "CRANE"] n'est pas strictement croissante (propriétés (3) et (4) non satisfaites) ; [] n'est pas un EnsembleDeMots non vide (propriété (1) non satisfaite). }

InsèreEns : un EnsembleDeMots {non vide}, unEnsembleDEnsemblesDeMots

→ unEnsembleDEnsemblesDeMots

{ InsèreEns(e, ee) est l'ensemble ee dans lequel l'ensemble de mots e a été inséré à la bonne place. Si e était déjà présent dans ee, alors le résultat est ee. Pré-condition: le premier mot de e est différent de tous les premiers mots de chaque élément de ee.

Par exemple InsèreEns(EM1,[EM3,EM2])=[EM1,EM3,EM2]

alors que InsèreEns(EM1,[EM1,EM3,EM2])=[EM1,EM3,EM2] }

— Indiquer pour chaque valeur ci-dessous si les propriétés (1),(2),(3),(4) mentionnées ci-dessus sont satisfaites ou non, et s'il s'agit donc d'un EnsembleDEnsemblesDeMots ou non.

(a) [["CRANE", "ECRAN", "NACRE"], ["ECRAN", "NACRE"], ["NACRE"]]

(b) [["GIGANTESQUE"], ["GRAND", "MOYEN", "PETIT"], ["GRAND", "VIDE"]]

— Donner pour chaque expression ci-dessous sa valeur ou la raison de l'erreur le cas échéant.

(c) InsèreEns(["DEUX", "UN"], [["AMI", "MILAN"], ["LIBRE", "TEMPS"], ["ZOO"]])

(d) InsèreEns(["DEUX"], [["AMI", "MILAN"], ["DEUX", "MILAN"], ["LIBRE"]])

Q8. Anagrammes d'une phrase [4 points]

On désire déterminer, pour une phrase donnée, les différents ensembles d'anagrammes que l'on trouve dans cette phrase. On introduit pour cela le lexique suivant.

LesVAnagrammes : une Phrase → un EnsembleDEnsemblesDeMots

{ LesVAnagrammes(P) est l'ensemble des ensembles des vrais anagrammes d'une phrase.

Par exemple LesVAnagrammes ([])=[] ,

LesVAnagrammes(Ph0) = [["ECRAN", "NACRE"], ["NU", "UN"]],

LesVAnagrammes(Ph1) = [["AIGLES", "GLAISE"], ["ARBRE", "BARRE"], ["AVEC", "CAVE"], ["MORDUE", "MOUDRE", "REMOUD"]] }

— Donner la valeur de l'expression ci-dessous.

(e) LesVAnagrammes(Ph2)

— Compléter la définition récursive de LesVAnagrammes fournie ci-dessous en utilisant la fonction Moins spécifiée comme suit.

Moins : une séquence de mots, une Phrase → une Phrase

{ Moins(S,P) est la phrase P dans laquelle toutes les occurrences des mots présents dans S ont été enlevés. Par exemple Moins([],Ph1)=Ph1, Moins(["ZOO"],Ph1)=Ph1,

Moins(["UN", "DE"], ["UN", "UN", "DEUX", "DE", "UN"])=["DEUX"] }

LesVAnagrammes ([]) = •••••

LesVAnagrammes (M o phrase) =

soit lesVAnagrammesDeM = •••••

dans si EstVide?(lesVAnagrammesDeM)

alors ••••••••••

sinon soit MEtLesVAnagrammesDeM = ••••••••••

dans ••••••••••

NOM : _____ PRENOM : _____ GROUPE : _____



Devoir Surveillé, Informatique, Grenoble, UE INF120

Répondre uniquement sur cette feuille

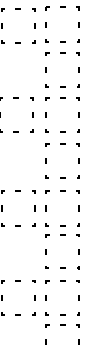
Utiliser un brouillon au préalable. La lisibilité des réponses pourra être prise en compte dans l'évaluation.



Ne pas remplir les cases

Q1

a: _____ b: _____
E1: _____
c: _____ d: _____
E2: _____
e: _____ f: _____
E3: _____
g: _____ h: _____
E4: _____



Q2

PO2E(C) = soit _____ = _____
dans _____

Q3i

PO3E(<x,y,z>) = soit _____ = _____
dans _____

Q3ii

La Q3ii est incorrecte car _____

Par exemple POE3E(_____) à pour valeur _____ et non pas _____

La ligne 2 corrigée devient : x<y et x<z : _____

Q4i

EgMots?(_____, _____) = _____
EgMots?(_____, _____) = _____
EgMots?(_____, _____) = _____
EgMots?(_____, _____) = _____

Q4ii

Ap?(____, _____) = _____

Ap?(____, _____) = _____

Ap?(L,M) : _____

Pour qu'une lettre L appartienne à un mot M il faut que _____

Q4iii

Moins(_____,_____) = _____

Moins(_____,_____) = _____

Q5

EstAna?(_____,_____) = _____

EstAna?(_____,_____) = _____

EstVraiAna?(M1,M2) : _____

Q6

VAnaDUnMot(M, _____) = _____

VAnaDUnMot(M, _____) = _____

Q7

[["CRANE", "ECRAN", "NACRE"], ["ECRAN", "NACRE"], ["NACRE"]]

(1) Ensemble d'ensembles non vide de mots. OUI NON

(2) Ensembles de mots distincts deux à deux. OUI NON

(3) Ordonnés par ordre alphabétique strictement croissant sur le 1er mot. OUI NON

(4) 1er mot de chaque ensemble distincts 2 à 2. OUI NON

[["GIGANTESQUE"], ["GRAND", "MOYEN", "PETIT"], ["GRAND", "VIDE"]]

(1) Ensemble d'ensembles non vide de mots. OUI NON

(2) Ensembles de mots distincts deux à deux. OUI NON

(3) Ordonnés par ordre alphabétique strictement croissant sur le 1er mot. OUI NON

(4) 1er mot de chaque ensemble distincts 2 à 2. OUI NON

InsèreEns(["DEUX", "UN"], [["AMI", "MILAN"], ["LIBRE", "TEMPS"], ["ZOO"]])

= _____

InsèreEns(["DEUX"], [["AMI", "MILAN"], ["DEUX", "MILAN"], ["LIBRE"]])

= _____

Q8

LesVAnagrammes(

["UNE", "LOUPE", "NUE", "SUR", "UN", "CRANE", "NU", "DE", "NACRE", "PRES",

"DE", "UN", "INGRAT", "ECRAN", "PLAT", "QUE", "GARNIT", "UNE", "POULE", "DE", "GRANIT"])

= _____

LesVAnagrammes ([]) = _____

LesVAnagrammes (M o phrase) =

soit lesVAnagrammesDeM = _____

dans si EstVide?(lesVAnagrammesDeM)

alors _____

sinon soit MEtLesVAnagrammesDeM = _____

dans _____

