Grenoble INP
ENSIMAG

Annales d'examens 1ère année

Année universitaire 2010/2011

Novembre 2010

ANALYSE POUR L'INGENIEUR – 1 PAGE

THEORIE DES LANGAGES – 3 PAGES

PROBABILITES APPLIQUEES 1 – 2 PAGES

Grenoble INP-Ensimag
1ère année

Analyse pour l'ingénieur

Examen à mi-parcours
Mardi 23 novembre 2010 - 1h30

Documents manuscrits et calculatrice autorisés

Exercice 1 Déterminer si elle existe la limite suivante :

$$\lim_{n\to+\infty}\int_0^1 \frac{1+nx}{(1+x)^n}dx$$

Exercice 2 Soit a, b, et L, des réels tels que $0 < a < b$ et $L > 0$.
Soit $f : \mathbb{R}^2 \to \mathbb{R}$ définie par $f(x,y) = \sin(xy)\ \chi_{[a,b]\times[0,L]}(x,y)$.
(χ désignant la fonction caractéristique)

1. Montrer que $f \in L^1(\mathbb{R}^2)$

2. En déduire :

$$\lim_{L\to+\infty}\int_0^L \frac{\cos(ax)-\cos(bx)}{x}dx = \ln(\frac{b}{a})$$

Exercice 3
Calculer la transformée de Fourier de

$$f(x) = \chi_{[-\frac{1}{2},\frac{1}{2}]}(\frac{x-\frac{1}{2}}{a})$$

où $a > 0$ et $\chi_{[-1/2,1/2]}$ est la fonction indicatrice de $[-1/2, 1/2]$.

Exercice 4
On considère $([0,1], \mathcal{B}, \mu)$, l'espace mesuré où $\mathcal{B}$ est la tribu des boréliens sur $[0,1]$ et μ la mesure de Lebesgue. Soit E un intervalle ouvert de $[0,1]$ et χ_E sa fonction indicatrice.
Soit $n \in \mathbb{N}$. Pour $j = 1, \cdots 2^n$, on pose : $I_j^n =]\frac{j-1}{2^n}, \frac{j}{2^n}[$
(on notera que à n fixé, les I_j^n sont deux à deux disjoints de mesure 2^{-n} et que $[0,1] = \overline{\bigcup_j I_j^n}$).
On considère la fonction sur $[0,1]$:

$$h_n(x) = 2^n \sum_{j=1}^{2^n} \mu(E \bigcap I_j^n)\ \chi_{I_j^n}(x)$$

1. Soit $K = \{\frac{j}{2^n}\ ;\ 1 \le j \le 2^n \ \ et \ \ n \in \mathbb{N}\}$. Montrer que la mesure de K est nulle.

2. Montrer que si $x \notin K$, alors x appartient à une suite décroissante d'ensembles $I_{j_n}^n$.

3. En déduire :

(i) Si x appartient à $E \setminus K$ (i.e. E privé de K), alors $h_n(x)$ tend vers 1 quand $n \to +\infty$.
(ii) Si x appartient à $[0,1] \setminus (E \bigcup K)$, alors $h_n(x)$ tend vers 0 quand $n \to +\infty$.

4. En déduire que $h_n(x)$ tend vers l'indicatrice de l'ensemble E presque partout.

Grenoble INP - Ensimag 1ère année novembre 2010

Théorie des langages 1

Durée : 2h.
Documents : tous documents autorisés.

Ce sujet est constitué de 3 exercices notés respectivement sur 11 points, 5 points et 4 points. Le barème est donné à titre indicatif.

Exercice 1 - Langage et grammaire (11 points)

Soit la grammaire $G = (V_T, V_N, S, R)$ avec $V_T = \{a, b\}$, $V_N = \{S\}$ et R l'ensemble des règles :

$$S \to \epsilon \mid aS \mid aSbS$$

On rappelle que, pour tout non-terminal $X \in V_N$, le langage $L(X)$ est défini par $L(X) = \{w \ : \ w \in V_T^* \wedge X \Longrightarrow^* w\}$. On a $L(G) = L(S)$ avec S l'axiome.

▷ **Question 1** (1 point)
Montrer que la chaîne aabba appartient à $L(G)$.

▷ **Question 2** (3 points)
On pose $L = \{w \ : \ w \in \ V_T^* \wedge \forall u \ . \ (u \in \ Prefix(w) \Rightarrow \ |u|_a \geq |u|_b)\}$.

On rappelle que $|u|_x$ dénote le nombre d'occurrences du symbole x dans u et que $Prefix(w)$ désigne l'ensemble des préfixes de w, i.e. $\{u \mid \exists\, v \ . \ (w = uv)\}$.

Montrer que $L(G) \subseteq L$.

▷ **Question 3** (2 points)
Montrer que la grammaire G est ambiguë.

▷ **Question 4** (2 points)
Soit la grammaire $G' = (V_T, V_N, S, R)$ avec $V_T = \{a, b\}$, $V_N = \{S, S_1, S_2\}$ et R l'ensemble des règles :

$$S \to S_1 \mid S_2$$
$$S_1 \to \epsilon \mid aS_1bS_1$$
$$S_2 \to aS \mid aS_1bS_2$$

Donner une définition de chacun des ensembles $E_1 = L(S_1)$, $E_2 = L(S_2)$ et $E = L(S)$, en donnant les propriétés caractérisant les mots de ces langages. **On ne prouvera pas les définitions proposées.**

▷ **Question 5** (3 points)
Montrer que $E_1 \subseteq L(S_1)$.

Exercice 2 - Modélisation de langages (5 points)

On s'intéresse à un langage de calcul booléen défini par la grammaire suivante, B étant l'axiome :

$$\begin{array}{l} B \rightarrow T \vee B \mid T \\ T \rightarrow F \wedge T \mid F \\ F \rightarrow \neg F \mid x \mid y \mid z \mid 0 \mid 1 \mid (B) \end{array}$$

avec $V_T = \{x, y, z, 0, 1, \vee, \wedge, \neg, (,)\}$.

▷ **Question 1** (2 points)
Donner l'arbre de dérivation de l'expression $\neg x \vee \neg(x \wedge y)$.
Préciser les priorités entre les opérateurs $\wedge$, $\vee$ et $\neg$.

▷ **Question 2** (3 points)
On veut ajouter à ce langage l'opérateur $\Rightarrow$. On rappelle que cet opérateur est moins prioritaire que les opérateurs $\vee$ et $\wedge$ et qu'il est associatif à droite mais pas à gauche. $0 \Rightarrow 1 \Rightarrow 0$ s'évalue en $0 \Rightarrow (1 \Rightarrow 0)$ qui vaut 1 et non pas en $(0 \Rightarrow 1) \Rightarrow 0$ qui vaut 0.

Etendre la grammaire précédente pour prendre en compte cet opérateur. La grammaire proposée devra être non ambiguë et refléter les priorités et la propriété d'associativité de cet opérateur. Pensez à vérifier en particulier que la grammaire proposée reconnait bien les 3 écritures précédentes : $0 \Rightarrow 1 \Rightarrow 0$, $0 \Rightarrow (1 \Rightarrow 0)$ et $(0 \Rightarrow 1) \Rightarrow 0$.

Exercice 3 - Calculs sur les grammaires (4 points)

La définition D1 des grammaires linéraires à droite vue en cours caractérise les règles de ces grammaires par une des trois formes suivantes :

1. $A \rightarrow a$
2. $A \rightarrow aB$
3. $A \rightarrow \epsilon$

avec A et B des non-terminaux quelconques et a un élément quelconque de V_T. Voici une autre définition D2 caractérisant les règles des grammaires linéraires à droite :

1. $A \rightarrow w$
2. $A \rightarrow wB$

avec w une chaîne sur V_T^* et A et B des non-terminaux quelconques. De manière évidente les règles conformes à la définition D1 sont conformes à la définition D2.

▷ **Question 1** (1 point)

Soit la grammaire $S \rightarrow \epsilon \mid ab \mid caS$

Donner une grammaire équivalente respectant la définition D1.

▷ **Question 2** (3 points)

Une grammaire G est dite avec cycle si et seulement si il existe un non-terminal A de G tel que $A \Rightarrow^{+} A$. o8;

Donner un algorithme permettant de construire, pour toute grammaire G **sans cycle** conforme à la définition D2, une grammaire équivalente G' conforme à la définition D1. On fera bien attention d'être exhaustif sur les différents cas possibles.

Ensimag 1re année **2010-2011**

Probabilités Appliquées
Novembre 2010

Durée 2h00 - Documents et calculatrices non-autorisés. Veuillez inscrire votre numéro de groupe sur la copie.

Exercice I (5 pts) - Éva et Raph jouent aux dés. Éva lance un dé à six faces alors que Raph lance un dé à sept faces. Ils lancent les dés en même temps. On note N le rang d'apparition du premier 1 chez Éva ou chez Raph.

1) Calculer la probabilité de l'événement $(N > k)$, pour $k \geq 1$. Quelle est la loi de N ?

2) Le premier 1 gagne la partie. Quelle est la probabilité pour que Éva gagne ?

3) Quelle est la probabilité de match nul ?

Exercice II (2 pts) - A Las Vegas, on rencontre une proportion p de tricheurs, $0 < p < 1$. Lorsque l'on joue contre un tricheur, la probabilité de gagner une partie est nulle, tandis que, lorsque l'on joue contre une personne honnête, cette probabilité est $1/2$. On joue et perd une partie à Las Vegas. Quelle est la probabilité d'avoir joué contre un tricheur ?

Exercice III (5 pts) - Dans l'amphi de 100 personnes, chaque élève assiste indépendamment au cours de proba 2 fois sur 3.

1) Quelle est la loi du nombre d'élèves présents dans l'amphi ?

2) Le prof a repéré au moins un absent le lundi 9 octobre. Quelle était alors la loi du nombre d'élèves présents dans l'amphi ce jour là ?

3) Dans les deux questions précédentes, quelle est l'espérance du nombre d'élèves ?

Exercice IV (3 pts) - On lance un dé à 6 faces et, indépendamment, on tire une variable aléatoire U de loi uniforme sur $(0, 1)$. Si le résultat du dé est pair, on gagne $2U$. Si, de plus, le résultat du dé est un six, on gagne $4U$ supplémentaires. S'il est impair, on ne gagne rien. Il faut payer la valeur 1 pour jouer à ce jeu. Quelle est la fonction de répartition du gain à l'issue de ce jeu ? Représenter graphiquement cette fonction.

Exercice V (5 pts) - Erwin Schrödinger a un chat probabiliste. On suppose que le chat d'Erwin prend une (et une seule) décision par jour parmi les possibilités suivantes : rentrer, rester à l'intérieur, sortir, rester dehors. La décision quotidienne est prise à 0h00, pour simplifier. Lorsqu'il se trouve dà l'intérieur, le chat d'Erwin peut sortir avec la probabilité p. Lorsqu'il se trouve dehors, il peut rentrer avec la probabilité q $(0 < p, q < 1)$. Le 31 décembre 2010, considéré comme le jour zéro de l'année 2011, Erwin et son chat sont à la maison.

1) On note π_n la probabilité pour que le chat soit dehors le soir n, $n \geq 0$. Établir la relation suivante

$$\forall n \geq 1, \quad \pi_n = (1 - p - q)\pi_{n-1} + p$$

2) Calculer π_n, ainsi que la probabilité limite.

3) Erwin se comporte comme son chat, mais les probabilités sont différentes (α et β). Les décisions d'Erwin sont indépendantes de celles de son chat. Donner une estimation de la probabilité pour qu'Erwin et son chat se rencontrent à la maison le 1 mai 2011.

JANVIER 2011

ALGORITHMIQUE 1 – 8 PAGES

ANALYSE POUR L' INGENIEUR – 2 PAGES

ARCHITECTURE – 7 PAGES

ECONOMIE - 4 PAGES

INTRODUCTION AUX RESEAUX - 6 PAGES

PROBABILITES APPLIQUEES 1 – 2 PAGES

THEORIE DES LANGUANGES 1 – 2PAGES

Algorithmique 1

Durée : 3h
Machines électroniques interdites **Tous documents papiers autorisés**

Les deux parties du sujet sont indépendantes.

Veuillez respecter les notations introduites dans l'énoncé. Il est inutile de paraphraser l'énoncé dans vos réponses, mais des explications avec dessins sur votre code sont les bienvenues. Sauf dans les questions où le style récursif est explicitement demandé, vous devez répondre aux questions du sujet en écrivant des procédures et fonctions itératives.

I - Énumération de l'ensemble "rangées possibles" au jeu "Embouteillage" (8 points)

Le but de l'exercice est de faire le calcul de l'ensemble des "rangées possibles" qui est utilisé dans le TPL1. On formalise ici la notion de *rangée possible* de la manière suivante :

> *Étant donné un entier naturel n, une* "rangée possible de taille n" *est un nombre qui peut s'écrire sous la forme*
> $$\sum_{i \in 0..n-1} d_i.3^i$$
> *tel que pour tout $i \in 0..n-1$,*
> - $d_i \in 0..2$;
> - *et, si $d_i = 1$ alors $i < n-1$ et $d_{i+1} = 0$;*
> - *et, si $d_i = 2$ alors $i < n-2$ et $d_{i+1} = d_{i+2} = 0$.*

Avec cette définition, le TPL1 utilise l'ensemble des rangées possibles de taille 6.[1]

Pour commencer, on veut définir une suite $(U_n)_{n \in \mathbb{N}}$ telle que U_n corresponde au nombre de rangées possibles de taille n.

---

[1]On rappelle que le chiffre 1 représente une tête de voiture, le chiffre 2 représente une tête de camion et le chiffre 0 représente les autres cas (case vide ou occupée par autre chose qu'une tête de véhicule). Les contraintes ci-dessus signifient donc simplement qu'une voiture occupe 2 cases (e.g. la tête est suivie d'une case de code 0) et qu'un camion en occupe 3 (e.g. la tête est suivie de 2 cases de code 0).

▷ **Question 1 (1 point) :**

Justifier les égalités suivantes :
- $U_0 = 1$
- $U_1 = 1$
- $U_2 = 2$
- $U_3 = 4$

Pour chaque U_n ci-dessus, on détaillera chacun des U_n codages ternaires (en base 3) à n chiffres correspondant à une rangée possible de taille n, ainsi que leur valeur décimale.

▷ **Question 2 (2 points) :**

Définir par récurrence la suite $(U_n)_{n \in \mathbb{N}}$. Détailler le calcul de U_6 (sans donner le détail des U_6 rangées possibles de taille 6).

▷ **Question 3 (1 point) :**

Programmer une fonction itérative `DecompteRangee(N)` qui calcule $U_{\mathtt{N}}$.

```
function DecompteRangee(N: Natural) return Natural ;
```

La récursivité est **interdite** dans cette fonction !

▷ **Question 4 (3 points) :**

Écrire une procédure *récursive* `EnumRangee(N,Prefixe)` qui énumère l'ensemble des rangées possibles de taille `N`, en les affichant successivement une fois et une seule. Pour chaque rangée possible r, on affichera en fait la valeur décimale de "$\mathtt{Prefixe}.3^{\mathtt{N}} + r$". L'ordre d'affichage n'est pas important.

```
procedure EnumRangee(N,Prefixe: Natural) ;
```

Ainsi, `EnumRangee(6,0)` affiche bien la valeur décimale de toutes les rangées de taille 6.
Indication : La récursivité porte sur le paramètre `N`. On pourra prendre `N=0` comme cas de base, qui correspond simplement à afficher la valeur de `Prefixe`.

On définit maintenant le type suivant :

```
type TabNat is array(Natural range <>) of Natural ;
```

▷ **Question 5 (1 point) :**

En réutilisant `DecompteRangee`, écrire une fonction `EnsRangee(N)` qui retourne l'ensemble des rangées possibles de taille `N` sous la forme d'un tableau où les rangées sont indicées de 0 à $U_{\mathtt{N}} - 1$. L'ordre d'énumération des rangées dans le tableau est indifférent.

```
function EnsRangee(N: Natural) return TabNat ;
```

Cette fonction peut utiliser un sous-programme auxiliaire récursif inspiré de `EnumRangee`, mais doit éviter de faire des recopies du tableau lors des appels récursifs.

II - Machines séquentielles de tri (12 points)

L'objectif de cet exercice est d'étudier l'implémentation du paquetage `Tri` dont l'interface est donné ci-dessous.

```
package Tri is
   -- machine séquentielle de tri de modèle 2

   subtype Element is ... ;  -- peu importe

   type TabElement is array(Positive range <>) of Element ;

   type Enum is private ;

   procedure Demarrer(E: out Enum; T: TabElement) ;

   function Fin(E: Enum) return Boolean ;

   procedure Avancer(E: in out Enum) ;
   -- requiert: not Fin

   function Courant(E: Enum) return Element ;
   -- requiert: il y a eu un "Avancer" (modèle 2).

private
   type Enum is ... ; -- défini plus loin
end Tri ;
```

Ce paquetage correspond à une machine séquentielle (de modèle 2), dont l'état est explicite sous la forme d'une valeur de type `Enum`. Une telle machine sert à énumérer les maximums successifs du tableau `T` (donné au moment de l'initialisation de la machine via le `Demarrer`) par ordre décroissant (ordre ">=" supposé défini sur le type `Element`). Elle permet donc en particulier de réaliser un tri des éléments du tableau.

Par exemple, sous l'hypothèse que `Element` est le type `Character`, la procédure ci-dessous affiche les 4 premiers maximums de la chaîne en paramètre ou s'interrompt en affichant un message d'erreur si cette chaîne a moins de 4 caractères.

```
procedure Exemple(S: String) is
   use Tri ;
   E: Enum ;
begin
   Demarrer(E,TabElement(S)) ;
   for I in 1..4 loop
      if Fin(E) then
         Put_Line("ERREUR: """ & S & """ a moins de 4 éléments") ;
         return ;
      end if ;
      Avancer(E) ;
      Put_Line("Maximum suivant:" & Courant(E)) ;
   end loop ;
end ;
```

L'appel `Exemple("IFOPULYRSA")` ; provoque l'affichage de :

```
Maximum suivant:Y
Maximum suivant:U
Maximum suivant:S
Maximum suivant:R
```

Une première idée pour implémenter ce paquetage `Tri` serait de commencer par trier le tableau dans `Demarrer`. La procédure `Demarrer` coûterait alors $\Theta(\text{T'Length} \times \log(\text{T'Length}))$ comparaisons dans le pire cas (via un tri fusion par exemple) et la procédure `Avancer` (qui n'incrémenterait qu'un indice) se ferait à coût constant.

L'objectif est ici d'étudier deux autres approches, dans lesquelles `Demarrer` ne fait aucune comparaison (elle ne fait que recopier le tableau `T`) et le premier `Avancer` (calculant le premier maximum) se limite à faire $\text{T'Length} - 1$ comparaisons sur les valeurs de type `Element`.

1 Méthode du *tri à bulles* (4 points)

Dans une première version, `Avancer` effectue une étape du *tri à bulles*. Le principe de cet algorithme est rappelé plus loin. Dans cette version, on suppose que la partie "*private*" du paquetage `Tri` introduit les types donnés à la figure 1. Le champ `Elem` est allouée dans `Demarrer` et contient initialement une copie de `T` (voir figure 2). Ce tableau est ensuite lu ou modifié dans les autres sous-programmes du paquetage. On ignorera ici le problème de sa désallocation. Le champ `DejaTrie` est introduit pour gagner du temps dans la recherche du maximum suivant, lorsqu'on sait que le tableau est déjà trié.

```
type DynTab is access TabElement ;
type Enum is record
   NbElem: Natural ;  -- nombre d'éléments restant à énumérer
   Elem: DynTab ; -- éléments restant à énumérer dans Elem(1..NbElem)
   DejaTrie: Boolean ;
end record ;
```

Si `T` est le tableau donné à `Demarrer`, alors la valeur `E` de type `Enum` générée pour ce tableau vérifie l'invariant suivant (préservé par `Avancer`) :
- `E.NbElem <= T'Length`;
- et, `E.Elem` est une permutation de `T` telle que `E.Elem'Range` est `1..T'Length`;
- et, si `E.NbElem < T'Length` alors `E.Elem(E.NbElem+1)` est le maximum de la tranche `E.Elem(1..E.NbElem+1)` ;
- et, pour tout `I` de `E.NbElem+1..T'Length-1`, on a `E.Elem(I+1) >= E.Elem(I)` ;
- et, si `E.DejaTrie` vaut `True` alors `E.Elem` est trié par ordre croissant.

FIG. 1 – Implémentation du type `Enum` (version 1)

```
procedure Demarrer(E: out Enum; T: TabElement) is
begin
   E.NbElem := T'Length ;
   E.Elem := new TabElement(1..T'Length) ;
   E.Elem.all := T ;
   E.DejaTrie := (E.NbElem <= 1) ;
end ;
```

FIG. 2 – Implémentation de Demarrer (version 1)

▷ **Question 6 (2 points) :**

Écrire une procédure intermédiaire `PlaceMax` qui permute les éléments de la tranche `E.Elem(1..E.NbElem)` de manière à ce que l'élément d'indice `E.NbElem` soit le maximum de cette tranche. Le champ `E.NbElem` n'est pas modifié par cette procédure, par contre `E.DejaTrie` peut passer à `True`. Cette procédure doit faire au plus `E.NbElem-1` comparaisons sur les valeurs de type `Element`. Elle doit préserver l'invariant donné figure 1 sur `E`.

```
procedure PlaceMax(E: in out Enum) is
 -- requiert E.NbElem > 0
```

Indication : En échangeant successivement, depuis les petits indices, les éléments successifs inversés, on peut à la fois propager le maximum à la bonne place et détecter si le tableau est trié. C'est la méthode du *tri à bulles*.

▷ **Question 7 (2 points) :**

En utilisant `PlaceMax`, écrire le code de `Avancer`. Écrire aussi le code de `Fin` et `Courant`.

2 Méthode du *tournoi* (8 points)

Dans la version précédente, via la méthode du tri à bulles, le deuxième `Avancer` fait `T'Length`-2 comparaisons dans le pire cas. On étudie ici une seconde méthode où le nombre de comparaisons du second `Avancer` (et des suivants) est en $\Theta(\log$ `T'Length`$)$ dans le pire cas. On se contentera toutefois d'admettre que l'algorithme décrit ci-dessous vérifie cette affirmation sans chercher à la démontrer.

Dans cette seconde version, la procédure `Avancer` suit l'algorithme d'un *tournoi* pour désigner le maximum suivant. Un tournoi est défini comme une suite de *tours* dans lequel chaque tour consiste à comparer deux à deux les éléments en compétition pour ce tour. On appelle *match* une telle comparaison entre une paire d'éléments : seul le maximum de chaque paire, appelé *vainqueur du match*, reste en compétition pour le prochain tour. On dit alors qu'il est *qualifié* pour le tour suivant. Dans le cas où le nombre d'éléments qualifiés pour un tour est impair, un des éléments est qualifié d'office pour le tour suivant. Le tournoi s'arrête lorsqu'il n'y a plus qu'un unique élément qualifié : c'est le *vainqueur du tournoi*, autrement dit le maximum des éléments initialement en compétition.

Remarquons ici que le deuxième maximum se trouve nécessairement parmi les éléments qui ont perdu un match contre le vainqueur du tournoi. En effet, par transitivité de la relation d'ordre, il est inutile de considérer les éléments qui ont perdu face à un autre perdant. Par conséquent, on adopte la règle suivante : *un élément qui perd un match est disqualifié des tournois suivants tant que le vainqueur de ce match n'est pas lui-même le vainqueur d'un tournoi.* Autrement dit, les éléments initialement qualifiés pour le tournoi suivant correspondent exactement aux éléments ayant perdu un match contre le vainqueur du tournoi précédent lors d'un tournoi antérieur.

En conclusion, la liste des éléments qualifiés pour un tournoi donné doit tenir compte de l'historique de l'ensemble des matchs de tous les tournois précédents. On représente cet historique via la structure de donnée `Championnat` définie figure 3.

```
type Cellule ;

type Championnat
     is access Cellule ;

type Cellule is record
   Val: Element ;
   Vaincus: Championnat ;
   Suiv: Championnat ;
end record ;
```

Toute cellule d'adresse `X: Championnat` (avec `X/=`**null**) vérifie l'invariant suivant :
- pour tout `Y: Championnat` accessible via `X.Vaincus` (en suivant les champs `Suiv` ou `Vaincus`), si `Y /=` **null**, alors on a `X.Val >= Y.Val`;
- et, les ensembles de cellules accessibles via `X.Suiv` et `X.Vaincus` sont disjoints et ne contiennent pas `X`.

FIG. 3 – Définition de la structure de données `Championnat`.

Si `Tete:Championnat` alors `Tete` et `Tete.Suiv`, ainsi que tous autres les éléments accessibles depuis `Tete` en ne suivant que les champs `Suiv`, forment la liste simplement chaînée des éléments qualifiés dans le tournoi en cours. Par ailleurs, pour tout élément `X:Championnat`, la liste chaînée (via les champs `Suiv`) dénotée `X.Vaincus` contient tous les éléments qui ont perdu un match contre `X.Val`, dans le tournoi en cours ou lors d'un précédent tournoi. Voir l'exemple de la figure 4 page 7.

On suppose donc maintenant que la partie "*private*" du paquetage `Tri` introduit le type `Enum` donné ci-dessous :

```
type Enum is record
   Tete: Championnat ;
   Max: Element ;
end record ;
```

Les fonctions `Fin` et `Courant` sont implémentées comme suit :

```
function Fin(E: Enum) return Boolean is
begin
   return E.Tete = null ;
end ;

function Courant(E: Enum) return Element is
begin
   return E.Max ;
end ;
```

Tournoi 1	$I, F, O, P, U, L, Y, R, S, A$
Fin tour 1	S^A, Y^R, U^L, P^O, I^F
Fin tour 2	$I^F, U^{P^O,L}, Y^{S^A,R}$
Fin tour 3	$Y^{S^A,R}, U^{I^F,P^O,L}$
Fin tour 4	$Y^{U^{I^F,P^O,L},S^A,R}$
Tournoi 2	$U^{I^F,P^O,L}, S^A, R$
Fin tour 1	$R, U^{S^A,I^F,P^O,L}$
Fin tour 2	$U^{R,S^A,I^F,P^O,L}$
Tournoi 3	R, S^A, I^F, P^O, L
Fin tour 1	$L, P^{I^F,O}, S^{R,A}$
Fin tour 2	$S^{R,A}, P^{L,I^F,O}$
Fin tour 3	$S^{P^{L,I^F,O},R,A}$
Tournoi 4	$P^{L,I^F,O}, R, A$
Fin tour 1	$A, R^{P^{L,I^F,O}}$
Fin tour 2	$R^{A,P^{L,I^F,O}}$

Dans ce tableau, on considère que le type `Element` correspond au type `Character`. La colonne de droite détaille l'évolution de la structure `Championnat` lorsqu'on énumère les 4 premiers maximums du tableau initial à 10 éléments : $I, F, O, P, U, L, Y, R, S, A$.

Pour i compris entre 1 et 4, le i-ème maximum correspond à l'unique élément qualifié à l'issue du dernier tour du "**Tournoi** i" ci-contre. Dans chaque tournoi, la première ligne indique la liste des éléments qui y sont initialement qualifiés ; et les lignes suivantes indiquent la liste des éléments qualifiés à la fin de chaque tour.

Ici, une telle liste (qui correspond à une liste chaînée suivant les champs `Suiv` de `Championnat`) est représentée horizontalement en séparant les éléments successifs par des virgules. De plus, chaque élément a lui-même en exposant la liste des éléments qu'il a vaincus (donnée par son champ `Vaincus`), cette liste étant elle-même représentée de la même façon.

FIG. 4 – Évolution de la structure `Championnat`

▷ Question 8 (1 point) :

Écrire la procédure `Demarrer`. Initialement, tous les éléments du tableau `T` sont qualifiés pour le premier tournoi. On demande ici que les éléments dans la liste des qualifiés apparaissent dans le même ordre que `T`, comme c'est illustré à la figure 4 (cette convention facilite le débogage).

```
procedure JoueTour(Tete: in out Championnat) is
   Cour, Suiv: Championnat ;
   ... -- A COMPLETER ÉVENTUELLEMENT

begin
   Cour:=Tete;

   while Cour /= null and then Cour.Suiv /= null loop
      Suiv := Cour.Suiv.Suiv ;

      -- match entre Cour et Cour.Suiv
      ...  -- A COMPLETER

      -- passage aux autres matchs du tour
      Cour := Suiv ;
   end loop ;

   if Cour=null then
      -- nombre pair d'éléments initialement qualifiés
      ... -- A COMPLETER
   else
      -- nombre impair d'éléments initialement qualifiés
      ... -- A COMPLETER
   end if ;
end ;
```

FIG. 5 – Un squelette possible de JoueTour

▷ **Question 9 (5 points) :**

Écrire une procédure intermédiaire `JoueTour` qui effectue un tour de tournoi. En entrée, `Tete` représente la liste des éléments qualifiés pour ce tour. En sortie, `Tete` représente la liste des qualifiés pour le tour suivant.

```
procedure JoueTour(Tete: in out Championnat) ;
```

La procédure `JoueTour` doit préserver l'invariant donné figure 3. Elle doit aussi suivre le principe des tours illustré à la figure 4. En particulier, l'ordre de chaînage des éléments qualifiés en sortie doit être inverse à celui de leur ordre de chaînage en entrée : cela permet une implémentation simple uniquement par réorganisation du chaînage, c'est-à-dire sans **allocation de cellule**, même pas une sentinelle.
Indication : on peut s'inspirer du squelette de la figure 5.

On introduit maintenant la procédure `Liberer` suivante :

```
procedure Liberer is
   new Ada.Unchecked_Deallocation(Cellule,Championnat) ;
```

▷ **Question 10 (2 points) :**

Écrire `Avancer` en utilisant `JoueTour`. Cette procédure effectue un tournoi comme illustré en figure 4. Elle doit libérer la cellule correspondant au maximum trouvé.

Grenoble INP-Ensimag **Analyse pour l'ingénieur**
1ère année

Examen 2

Mercredi 26 janvier 2011, 14h30-16h, documents de cours autorisés

Exercice 1

1. Calculer dans $\mathcal{D}'(\mathbb{R})$ la limite quand $h \to 0$ des distributions-fonctions, associées aux fonctions suivantes définies sur $\mathbb{R}$ par:

a) $f_h(t) = \frac{h}{h^2+t^2}$

b) $f_h(t) = \frac{ht}{h^2+t^2}$

Indication : on fera un changement de variable approprié.

2. Pour $n \in \mathbb{Z}$, on rappelle la masse de Dirac en n : $\delta_n : \varphi \to \varphi(n)$.

a) Calculer δ'_n, dérivée de δ_n dans $\mathcal{D}'(\mathbb{R})$, ainsi que la dérivée dans $\mathcal{D}'(\mathbb{R})$ de $\sum\limits_{n\in\mathbb{Z}} \delta_n$.
(on montrera auparavant que $\sum_n \delta_n$ est une distribution).

b) Soit f la fonction de période 1 égale à $f(t) = t$ si $t \in [0, 1[$. Calculer sa dérivée, ainsi que sa dérivée seconde dans $\mathcal{D}'(\mathbb{R})$.

Exercice 2 Soit ϕ une fonction appartenant à $L^1(\mathbb{R})$ telle que $\hat{\phi}$ appartienne à $L^1(\mathbb{R})$, le but de cet exercice est de résoudre l'équation fonctionnelle d'inconnue $f \in L^1(\mathbb{R})$:

$$f(x) - k(f(x+1) + f(x-1)) = \phi(x), \quad |k| < \frac{1}{2}$$

1. Trouver l'expression de $\hat{f}$ en fonction de $\hat{\phi}$.
2. Montrer que $\hat{f} \in L^1(\mathbb{R})$.
3. En déduire l'expression de f comme une intégrale de $\hat{\phi}$.

Exercice 3 *Le principe d'incertitude d'Heisenberg*
Soit f une fonction de $\mathbb{R} \to \mathbb{C}$ telle que f, $x \to xf(x)$ et $\xi \to \xi\hat{f}(\xi)$ soient dans $L^2(\mathbb{R})$. On définit les trois quantités suivantes :

$$\begin{aligned} \sigma_f^2 &= \int_{\mathbb{R}} x^2|f(x)|^2 dx \quad \text{(dispersion d'énergie de } f \text{ en temps)} \\ \sigma_{\hat{f}}^2 &= \int_{\mathbb{R}} \xi^2|\hat{f}(\xi)|^2 d\xi \quad \text{(dispersion d'énergie de } f \text{ en fréquence)} \\ E_f &= \int_{\mathbb{R}} |f(x)|^2 dx \quad \text{(énergie de } f\text{)} \end{aligned}$$

1. Justifier l'existence de ces trois quantités.

Dans les questions suivantes, on suppose de plus que f appartient à $C^1(\mathbb{R})$, est à valeur dans $\mathbb{C}$, et est telle que f,f' et $x \to xf(x)$ sont dans $L^2(\mathbb{R})$.

2. On pose $g(x) = x|f(x)|^2$. On rappelle que $|f(x)|^2 = f(x)\overline{f(x)}$ (où $\bar{y}$ est le complexe conjugué de y).

a) Montrer que g est continue, dérivable sur $\mathbb{R}$, et que g, g' sont dans $L^1(\mathbb{R})$.

b) En remarquant que

$$g(x) = g(0) + \int_0^x g'(t)dt,$$

montrer que g a des limites en $\pm\infty$, puis que cette limite est nécessairement nulle, c'est à dire $\lim_{|x|\to+\infty} x|f(x)|^2 = 0$.

3.

a) Montrer que la dispersion d'énergie de f en fréquence vérifie :

$$\sigma_{\hat{f}}^2 = \frac{1}{4\pi^2}\int_{\mathbb{R}} |f'(x)|^2 dx,$$

b) puis que

$$E_f = -\int_{\mathbb{R}} x\,(f(x)\bar{f}(x))'\,dx.$$

c) En remarquant alors que

$$|\int_{\mathbb{R}} x\,(f(x)\bar{f}(x))'\,dx| \leq \int_{\mathbb{R}} |xf'(x)\bar{f}(x)|dx + \int_{\mathbb{R}} |xf(x)\bar{f}'(x)|dx, \tag{1}$$

montrer, en utilisant l'inégalité de Cauchy-Schwarz que :

$$\sigma_f \sigma_{\hat{f}} \geq \frac{E_f}{4\pi}. \tag{2}$$

Ceci constitue le théorème d'incertitude de Heisenberg.

4. *(Facultatif)* En supposant que f est à valeur dans $\mathbb{R}$, étudier les cas d'égalité pour l'équation (1), puis pour l'équation (2). En déduire que les seules fonctions qui vérifient le cas d'égalité dans (2) sont de la forme $f(x) = \alpha e^{-cx^2}$ pour $c > 0$.

Architecture 1 : Circuits Numériques et Éléments d'Architecture
Examen

ENSIMAG 1A

Année scolaire 2010–2011

- Durée : 3h. Tous documents et calculatrices autorisés.
- Le barème est donné à titre <u>indicatif</u>.
- Les exercices sont indépendants et peuvent être traités dans le désordre.
- Pensez à indiquer votre numéro de groupe sur chacune de vos copies.

Vous devez rendre les 3 exercices sur des copies séparées.

Ex. 1 : Circuit de transcodage ASCII vers code Morse (4 pts)

Le code Morse a été utilisé de nombreuses années pour transmettre de l'information à distance, grâce à, par exemple, des impulsions lumineuses ou électriques, de durée soit courte, soit longue. On se propose ici de réaliser un circuit de transcodage qui, à partir du code ASCII d'une lettre majuscule (A-Z), va produire un code sur 1 à 4 bits, sachant qu'un bit à zéro représente une impulsion courte et un bit à 1 une impulsion longue. Comme le code résultant est à longueur variable, il faut également avoir 2 bits qui en indiquent la taille. On choisi le codage des entiers naturels pour 1, 2 et 3, et on code 00 la valeur 4.

Ainsi, les données du problème sont les suivantes :

Caractère	code ASCII (hexa)	code morse binaire	taille	code taille binaire	Caractère	code ASCII (hexa)	code morse binaire	taille	code taille binaire
A	41	0 1	2	10	N	4E	1 0	2	10
B	42	1 0 0 0	4	00	O	4F	1 1 1	3	11
C	43	1 0 1 0	4	00	P	50	0 1 1 0	4	00
D	44	1 0 0	3	11	Q	51	1 1 0 1	4	00
E	45	0	1	01	R	52	0 1 0	3	11
F	46	0 0 1 0	4	00	S	53	0 0 0	3	11
G	47	1 1 0	3	11	T	54	1	1	01
H	48	0 0 0 0	4	00	U	55	0 0 1	3	11
I	49	0 0	2	10	V	56	0 0 0 1	4	00
J	4A	0 1 1 1	4	00	W	57	0 1 1	3	11
K	4B	1 0 1	3	11	X	58	1 0 0 1	4	00
L	4C	0 1 0 0	4	00	Y	59	1 0 1 1	4	00
M	4D	1 1	2	10	Z	5A	1 1 0 0	4	00

On note $(c_7, ..., c_0)$ les bits codant le caractère ASCII d'entrée, $(m_3, ..., m_0)$ les bits donnant le code Morse de sortie, et (t_1, t_0) les bits donnant la taille du code Morse.

Question 1 Déterminer le nombre de bits significatifs permettant de distinguer les 26 caractères de l'alphabet. En considérant que 0 indique le bit le moins significatif et 7 le bit le plus significatif, indiquez la position des bits du code ASCII qui permettent de distinguer les différents caractères considérés.

Dans la suite, ne pas oublier de prendre en compte les optimisations possible pour les fonctions ϕ-booléennes (aussi dites « non complètement spécifiées »). Par ailleurs, afin que l'exercice ne soit pas trop long, on ne s'intéressera dans la suite qu'aux bits m_2 et m_1 du code Morse.

Question 2 On ne considère dans un premier temps que les lettres A à O incluses. Parmi les bits significatifs trouvés à la question 1, distinguez ceux qui le sont toujours pour cette restriction de ceux qui deviennent invariants (donnez dans ce cas leur valeur). Pour quelle valeur du ou des autres bits cette table est-elle valide ? Dessinez les tables de Karnaugh de m_2 et m_1 en fonction de ces bits significatifs et construisez les groupements permettant de minimiser la fonction. On ne demande pas les équations des fonctions simplifiées.

Question 3 On ne considère dans un deuxième temps que les lettres P à Z incluses. Quels sont les bits significatifs de cette restriction ? Pour quelle valeur du ou des autres bits cette table est-elle valide ? Dessinez les tables de Karnaugh de m_2 et m_1 en fonction de ces bits significatifs et construisez les groupements permettant de minimiser la fonction. On ne demande toujours pas les équations des fonctions simplifiées.

Question 4 En remarquant que les tables des 2 questions précédentes font intervenir les mêmes bits et qu'elles sont établies pour un même bit qui dans le premier cas vaut 0 et dans le second cas vaut 1, on constate que 2 cases aux mêmes « coordonnées » dans les 2 tables sont voisines, et qu'ainsi on peut construire des groupements de bits également entre les 2 tableaux de Karnaugh à 4 variables. Une vision en 3D de ces deux tableaux donnerait des groupements verticaux.

En prenant en compte cette remarque, construisez les groupements à l'intérieur des tables et entre tables qui permettent de minimiser la fonction.

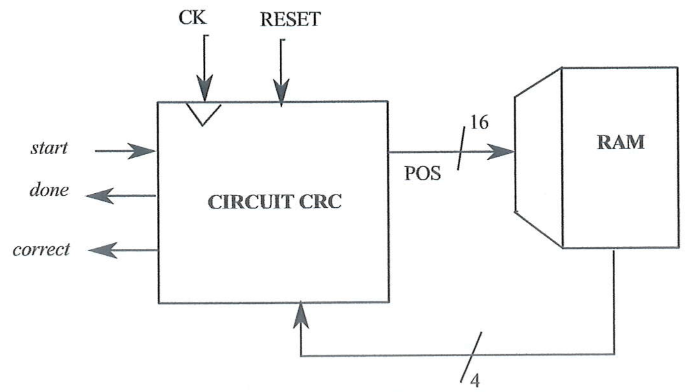
Donnez enfin les équations simplifiées de m_2 et m_1.

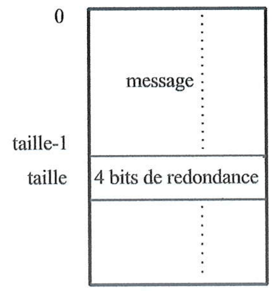
Ex. 2 : Conception Partie-Contrôle/Partie Opérative d'un code CRC (6 pts)

Un code de contrôle de redondance cyclique (CRC) est un mécanisme facile à mettre en œuvre et très efficace pour détecter des erreurs dans une transmission numérique. L'émetteur ajoute simplement à la suite du message à transmettre des bits de redondance calculés en fonction des bits du message et d'un polynôme générateur, également connu du récepteur. Ce dernier peut donc exécuter le même algorithme sur le message reçu et comparer les bits de redondance obtenus avec ceux transmis par l'émetteur. En cas d'égalité, le message reçu a une probabilité d'autant plus forte d'être correcte que le degré du polynôme est grand (et que ce dernier est bien choisi).

Nous étudierons ici un code CRC utilisant un polynôme de degré 4, ce qui signifie que 4 bits seront ajoutés à la fin du message. On supposera que le message et les bits de redondance sont disponibles dans une mémoire RAM contenant des mots de 4 bits, ayant un bus d'adresse de 16 bits et n'ayant pas de signaux de contrôle[1] Le message est de longueur `nbMots` x4 bits.
Le schéma ci-dessous résume ces informations :

1. La mémoire étant toujours en lecture dans cet exercice, on ne vous demande pas de générer les signaux de contrôle.





contenu de la RAM

start : début de la vérification (le message est déjà chargé dans la RAM, suivi des 4 bits de redondance)
done : la vérification est terminée ; la sortie ***correct*** à 1 indique qu'il n'y a pas eu d'erreur détectée

L'algorithme utilisé pour le décodage est le suivant [2] :

```
// poly est une constante sur 4 bits décrivant le polynôme de degré 4
   P(x) = x^4 + x^3 + x + 1. Le terme de degré max, toujours à 1, n'étant pas
   représenté, poly vaut 0xB.
taille ← nbMots ;                                          // sur 16 bits
pos ← 0 ;
courant(7 : 4) ← message[pos] ;   // (7:4) désigne les 4 bits de poids fort de courant
pos ← pos + 1 ;
courant(3 : 0) ← message[pos] ;             // 4 bits de poids faible de courant
while pos < taille do
    pos ← pos + 1 ;
    pos_bit ← 0 ;
    while pos_bit < 4 do
        if courant(7) = 1 then
            courant ← courant << 1 ;
            courant(7 : 4) ← courant(7 : 4)^poly ;   // ^ est l'opérateur XOR bit à bit
        else
            courant ← courant << 1 ;
        endif
        pos_bit ← pos_bit + 1 ;
    endw
    courant(3 : 0) ← message[pos] ;         // 4 bits de poids faible de courant
endw
return courant(7 : 4) = courant(3 : 0)
```

L'entrée `nbMots` n'est valide qu'au début de l'algorithme.

On va utiliser la méthode présentée en cours et mise en œuvre en TD pour proposer une implantation de cet algorithme sous la forme PC/PO.

Question 1 Définissez l'ensemble des opérations à réaliser ; déduisez-en le type des unités fonctionnelles (registres ou opérateurs). On cherchera à utiliser les opérateurs les plus simples pour chaque opération.

Question 2 Quelles opérations peut-on réaliser en parallèle ?
Réécrivez le programme en utilisant la notion d'affectation concurrente présentée en cours, par exemple $(a, b) \leftarrow (3, 1)$ affecte 3 dans a et 1 dans b.

2. Pour l'émetteur, c'est identique sauf que la fonction doit retourner les bits de redondance contenu dans courant(7:4)

Question 3 Proposez une partie opérative interconnectant les unités fonctionnelles. Indiquer clairement les signaux de commandes (sélecteurs de mux, signaux de chargement de registres, etc) et les comptes-rendus.

Question 4 Proposez une partie contrôle qui pilote cette partie opérative, en prenant en compte les parallélisations éventuelles trouvées (ou non) à la question 2.
Pour permettre la synchronisation entre le circuit utilisateur et le circuit CRC, on utilisera les signaux de commandes *start* et *done* introduit dans la figure.

Ex. 3 : Conception de processeur (10 pts)

Cet exercice utilise et complète le processeur vu en TD et TP, dont le jeu d'instructions et la partie opérative sont donnés à la fin du sujet.

Il s'agit d'adresser des tableaux, en particulier en les parcourant du début à la fin. Il faut pour cela rajouter un registre d'adressage `areg` de 16 bits, qui contient l'adresse de l'élément courant du tableau. Le registre `areg` :

- peut être initialisé à l'adresse de début du tableau par une instruction `la imm16`
- peut être incrémenté par l'instruction `incra` et décrémenté par l'instruction `decra`
- est utilisé pour lire un élément du tableau par l'instruction `ld (areg), rd`
- est utilisé pour écrire dans le tableau par l'instruction `st rs,(areg)`

Question 1 Utiliser ces instructions pour programmer en assembleur le programme suivant [3] :

```
i : integer;
i := n ;
while i /= 0  loop
    T(n-i) := T(n-i) + 1;
    i := i - 1;
endloop;
```

```
i : integer;
i := n;
areg = adresse de T(0);
while i /= 0  loop
    mem(areg) := mem(areg) + 1;
    i := i - 1;
    areg := areg + 1 ;
endloop;
```

Ce programme ajoute 1 à tous les éléments d'un tableau T d'entiers codés sur 8 bits, avec $n = 128$ éléments. Le tableau commence à l'adresse 0x2500. On utilisera `r0` pour mémoriser i, `r1` pour mémoriser n, `r2` pour mémoriser la constante 1, `r3` pour mémoriser $T(i)$ et `areg` pour mémoriser l'adresse de l'élément courant de T.

Question 2 Modifier la partie opérative du processeur pour pouvoir exécuter les nouvelles instructions ; pour le registre `areg`, vous pouvez

- soit le connecter à l'opérateur `+1` déjà existant, en le transformant en opérateur `+1/-1`,
- soit utiliser un registre compteur/décompteur à chargement parallèle.

Question 3 Proposer un codage des nouvelles instructions.

Question 4 Compléter le graphe d'exécution des instructions vu en TD par l'exécution des nouvelles instructions, et associer les signaux de contrôle activés à chacun des nouveaux états.

On cherche à programmer un algorithme de tri par bulle sur ce processeur (T est un tableau d'entiers naturels de 128 éléments, qui commence à l'adresse 0x2500) :

3. La version de gauche est la représentation sous forme algorithmique classique. La version de droite détaille en plus les calculs d'adresses du tableau qui sont implicites sur la version classique.

```
1   permut : boolean;
2   i : integer;
3   permut := true;
4   while permut = true loop
5       permut := false;
6       i := n - 1;
7       while i /= 0 loop
8           if T(n-1-i) > T(n-i) then
9               echanger (T(n-1-i), T(n-i));
10              permut := true ;
11          endif;
12          i := i - 1;
13      endloop;
14  endloop;
```

Question 5 Programmez cet algorithme en assembleur, avec le jeu d'instructions complété par les instructions définies plus haut. On rappelle que l'indicateur C du registre STATUS est mis à 1 quand la soustraction d'entiers naturels donne un résultat négatif (`rd < rs`).

On utilisera `areg` pour mémoriser l'adresse de l'élément courant de T, `r0` pour mémoriser *permut*, `r1` pour mémoriser i, `r2` pour mémoriser $T(i)$, `r3` pour mémoriser $T(i+1)$, `r2` et `r3` pouvant aussi être utilisés pour d'autres valeurs.

Question 6 Evaluez le nombre de cycles nécessaires à l'exécution de cet algorithme pour une itération (ligne 8 à 11) dans le cas où un échange a lieu.

On cherche à accélérer l'exécution de ce type de programme : on crée pour cela un indicateur dans le registre STATUS et des instructions spécifiques :

- l'indicateur X permet d'implémenter facilement un booléen de test ; il peut être mis à 1 par l'instruction `setx`, à 0 par l'instruction `clrx`, et peut être modifié par l'instruction `texch` définie ci-dessous. Aucune autre instruction ne le modifie.
- l'instruction `jx AD` est un branchement conditionnel (saut si X = 1)
- l'instruction complexe "test and exchange" `texch rs,rd`.

L'instruction `texch rs,rd` implémente :

rs := MEM(AREG) ;

rd := MEM (AREG+1) ;

C := (rd < rs) ;

if C = 1 then MEM(AREG) := rd ; MEM(AREG+1) := rs ; X := 1 ;

AREG := AREG + 1 ;

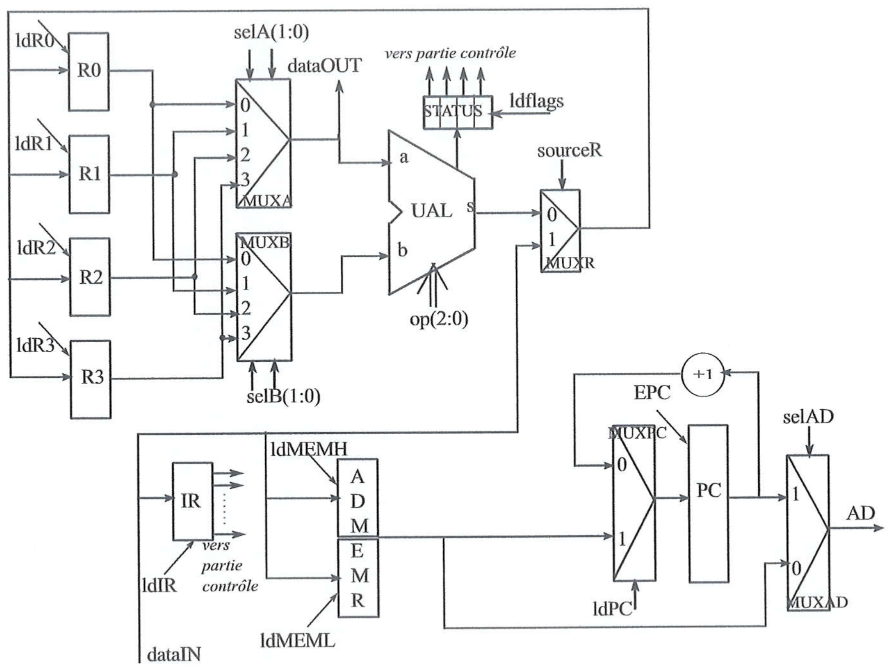
Question 7 Proposez un programme assembleur implémentant l'algorithme de tri par bulles en utilisant ces nouvelles instructions.

Question 8 Donnez la partie du graphe d'états de la partie contrôle nécessaire à l'exécution de l'instruction `texch`. Evaluez le nombre de cycles nécessaires à l'exécution de cette instruction dans le cas où l'échange est effectué et comparez au résultat obtenu en question 6. Etudiez les modifications de la PO nécessaires.

JEU D'INSTRUCTIONS DEFINI EN TD

Instruction	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
Opérations de la forme : $rd := rd\ op\ rs$								
`or rs, rd`	0	0	0	0	rs_1	rs_0	rd_1	rd_0
`xor rs, rd`	0	0	0	1	rs_1	rs_0	rd_1	rd_0
`and rs, rd`	0	0	1	0	rs_1	rs_0	rd_1	rd_0
`add rs, rd`	0	1	0	0	rs_1	rs_0	rd_1	rd_0
`sub rs, rd`	0	1	0	1	rs_1	rs_0	rd_1	rd_0
Opérations de la forme : $rd := op\ rd$								
`not rd`	0	0	1	1	0	0	rd_1	rd_0
`shl rd`	0	1	1	0	0	0	rd_1	rd_0
`shr rd`	0	1	1	1	0	0	rd_1	rd_0
Chargement : $rd := MEM(AD)$								
`ld AD, rd`	1	0	0	0	0	0	rd_1	rd_0
	ADH							
	ADL							
Stockage : $MEM(AD) := rs$								
`st rs, AD`	1	1	0	0	0	0	rs_1	rs_0
	ADH							
	ADL							
Branchement inconditionnel : $PC := AD$								
`jmp AD`	1	0	0	0	0	1	0	0
	ADH							
	ADL							
Chargement de constante : $rd = imm8$								
`li imm8, rd`	1	0	1	0	0	0	rd_1	rd_0
	imm8							
Branchements conditionnels : si $COND$ alors $PC = AD$								
`jz AD`	1	0	0	0	0	1	0	1
	ADH							
	ADL							
`jc AD`	1	0	0	0	0	1	1	0
	ADH							
	ADL							

PARTIE OPERATIVE DÉFINIE EN TD



ENSIMAG 1ère année – 2010/2011

Economie

DS du lundi 24 janvier 2011

Durée : 2 heures
Documents autorisés.

Travail demandé :

Choisir l'un des textes ci-joints.
Rédiger un **commentaire libre** du texte choisi.

Veillez à structurer votre travail (introduction, développement, conclusion) et à soigner la rédaction.

L'introduction doit obligatoirement être structurée comme suit :

Une phrase de synthèse qui résume la problématique du texte.
Une phrase qui justifie (en énonçant son rapport avec le texte) et annonce l'idée générale que vous souhaitez développez.
Une phrase ou deux phrases qui présentent l'enchaînement des parties qui composeront votre développement et qui montrent que ce dernier sera bien la « démonstration » de votre idée générale.

Veillez également à mettre en valeur vos **connaissances personnelles** sur le sujet retenu. C'est à partir de ces connaissances que vous devez élaborer votre commentaire, en montrant en quoi elles permettent d'éclairer les informations contenues dans le texte que vous avez choisi.

L'ensemble de votre travail ne doit pas dépasser **quatre pages**.

Texte 1

« Flexisécurité » : le Conseil d'analyse économique livre ses propositions

Le Monde – 6 mai 2010

Un salaire d'appoint et un accès plus large à la formation favoriseraient les mobilités choisies, qu'elles soient interprofessionnelles ou géographiques, estime le Conseil d'analyse économique dans un rapport publié jeudi. La France s'est engagée dans un processus de « flexisécurité » mais « *la flexibilité a augmenté plus vite que la sécurité des travailleurs* », constatent ses auteurs, Mathilde Lemoine (HSBC France) et Etienne Wasmer (Sciences Po Paris-OFCE).

Avec 15,3 % des salariés en contrats temporaires en 2007, contre 9,6 % en 1990 et 6,6 % en 1982, le marché de l'emploi a certes accru sa vitesse d'ajustement aux variations cycliques[1] mais au prix de fortes inégalités qui, au bout du compte, réduisent sa performance globale, constatent-ils. Les salariés précaires ou flexibles ont peu accès à la formation, les entreprises n'ayant aucun intérêt à les former puisqu'elles ne les gardent pas. Et pour les salariés qui en bénéficient, les formations sont souvent trop spécialisées, leur permettant d'être plus efficaces à leur poste mais non de développer des compétences générales. Avec peu d'espoir d'améliorer leur employabilité[2], les conséquences sont des pertes de « capital humain » pour ces travailleurs et pour l'économie dans son ensemble. « *Qui dit amélioration du capital humain dit amélioration du potentiel de croissance* », a souligné Mathilde Lemoine lors d'une présentation à la presse.

A l'opposé de la précarité, le fait que la protection de l'emploi soit basée sur l'ancienneté représente, après un certain temps passé dans l'entreprise, une désincitation puissante à la mobilité, observe le rapport en demandant que l'ancienneté retenue soit désormais celle en activité, et non plus dans l'entreprise. Pour encourager les mobilités choisies, le document préconise le versement d'un complément salarial, pendant un à deux ans et sous conditions de ressources du ménage, aux personnes qui font le pari de changer de secteur ou d'entreprise et qui, de ce fait, subissent une baisse de salaire.

Les entreprises, de leur côté, seraient responsabilisées en matière d'employabilité de leurs salariés avec l'instauration d'un bonus-malus sur les cotisations d'assurance-chômage et sur leurs obligations de reclassement. En fonction de la part des salariés formés et du caractère diplômant ou certifiant des formations, elles paieraient plus ou moins. « *Le reclassement des salariés est d'autant plus facile que leur employabilité a été maintenue, c'est-à-dire qu'ils ont pu développer leurs compétences générales* », note le rapport. La qualité même des formations, trop spécialisées et souvent inaccessibles aux salariés des petites entreprises ou aux travailleurs peu qualifiés, est mise en accusation.

Pour remédier à ces déséquilibres, le rapport propose d'étendre les contrats de transition professionnelle (CTP) – actuellement limités aux salariés menacés de licenciement

---

[1] Les variations cycliques désignent ici les variations de la croissance économique.
[2] Employabilité : ensemble des aptitudes qui permettent de prétendre à un emploi.

économique dans certains bassins d'emploi – à tous les travailleurs en contrat temporaire et de les généraliser sur l'ensemble du territoire. « Quand on voit que 6 % des salariés bénéficient de 50 % des formations, est-ce que c'est vraiment efficace ? », a noté Mathilde Lemoine en prônant une réorientation vers les formations certifiantes et diplômantes.

Les salariés seraient également invités à se former en amont avec une meilleure information et la création d'un « chèque formation » librement utilisable. Le rapport pousse la réflexion jusqu'aux difficultés de logement auxquelles sont confrontés les travailleurs qui changent de ville, de département ou de région, proposant des formes de baux plus souples pour aider les chômeurs et les salariés à faible revenu à déménager. Les auteurs chiffrent le coût de l'extension du CTP à 5 ou 6 milliards d'euros, à quoi s'ajouterait 1,5 milliard au maximum pour les aides à la mobilité et les compléments salariaux, soit un total de 8 milliards d'euros, somme qui serait assurée par le redéploiement d'autres moyens.

« *Si on regarde les moyens de la formation professionnelle et d'autres dispositifs, on a largement de quoi faire une flexisécurité à la française* », a affirmé Mathilde Lemoine.

Texte 2

Une leçon mal apprise

Source : www.challenges.fr, 13/11/2008

Le livre de Galbraith sur la crise de 1929 (*La Crise économique de 1929*, par John Kenneth Galbraith, Petite Bibliothèque Payot) permet de déceler de troublants points communs avec l'actualité : la spéculation foncière en Floride dans les années 1920 est l'ancêtre des subprimes.

Extraits (p. 28-29 et 43-44).

« Une chose aurait dû paraître évidente dans les années 1920, même à Coolidge [Calvin Coolidge a été président républicain d'août 1923 à mars 1929. NDLR] : elle concernait le caractère du peuple américain dont il avait si bien parlé. A côté des solides qualités dont il avait fait l'éloge, les Américains manifestaient un désir excessif de s'enrichir avec le minimum d'efforts. La première manifestation frappante de ce trait de leur personnalité intervint en Floride.
Là, vers le milieu des années 1920, Miami, Miami Beach, Coral Gables, la côte Est jusqu'au Palm Beach au nord et les villes en bordure du golfe du Mexique avaient été touchés par le grand boom sur les terrains en Floride. Le boom de la Floride contenait tous les éléments de la classique affaire véreuse en matière de spéculation. Il y avait l'indispensable élément de fond. La Floride avait un meilleur climat d'hiver que New York, Chicago ou Minneapolis. Des revenus plus élevés et de meilleures communications la rendaient de plus en plus accessible au Nord souffrant du froid. L'époque arrivait, en effet, où le vol annuel vers le sud serait aussi régulier et impressionnant que les migrations de l'oie au Canada.
En Floride, la terre fut partagée en lotissements à construire et vendue pour un règlement comptant de 10%. Manifestement, une grande partie du terrain ingrat qui changeait ainsi de main était aussi peu agréable à voir aux gens qui l'achetaient qu'à celui qui passait devant. Les acquéreurs ne comptaient pas y vivre; il n'était pas facile de supposer que quiconque le ferait jamais; mais ces considérations étaient théoriques. La réalité était que ces biens douteux gagnaient en valeur tous les jours et pouvaient être vendus avec un solide bénéfice quinze jours après. »

Notre avis. La tornade financière actuelle donne envie de se repasser le film de 1929. En France, le livre de Galbraith atteint des records de vente. Les éditions Payot, qui en ont vendu 6000 en vingt ans, ont réédité l'ouvrage le 24 septembre.
En trois semaines, les 3000 exemplaires ont été écoulés. Un nouveau tirage de 2000 exemplaires est en rayon. L'ouvrage, paru pour la première fois en 1955, se lit plutôt facilement et permet de déceler de troublants points communs entre hier et aujourd'hui : c'est le virus du crédit qui, à chaque fois, déclenche la crise. La frénésie spéculative qui a embrasé la Floride en 1929 fait penser au marché immobilier californien, qui a été le coeur de la crise des crédits hypothécaires à taux variables, ou subprimes. Et puis, tout l'édifice bancaire vacille... La différence, c'est le traitement de la crise.
Baisse des taux, intervention étatique et massive : les autorités américaines ont fait le contraire de ce qui s'était produit en octobre 1929. Ben Bernanke, le président de la Réserve fédérale américaine, est un grand lecteur de Galbraith.

Grenoble INP – ENSIMAG
Année 2010-2011

N° d'inscription (carte étudiant) :

NOM :

Prénom :

Né(e) le : **à (ville + dépt ou pays) :**

N° du groupe de TD:

N° de la place :

Introduction aux réseaux

Examen du 26/01/2011. Durée 2h.
Tous documents et calculettes autorisés

NOTE IMPORTANTE : vous devez répondre sur les feuilles de ce sujet d'examen. Préparez vos réponses d'abord sur un brouillon, et reportez les sur ce sujet qui sera ramassé en fin d'examen.
Le barème est indicatif.

Partie 1. Questions à Choix Multiples (10 points)

Modalités des QCM
Pour chaque QCM, une seule des solutions proposées est correcte, c'est à dire une réponse adéquate à la question posée (certaines « solutions » peuvent être des affirmations vraies sans répondre à la question). Entourez le numéro (alphabétique) de la réponse exacte.

Barème :
Réponse exacte : 1 point
Réponse fausse : -0,5 point
Omission : 0 point.

Question 1. *Combien de bits sont nécessaires pour transmettre le résultat d'une partie d'échecs (gain, perte, nulle) entre deux joueurs de force comparable?*

A. 1 bit
B. 1,6 bit
C. 2 bits
D. 3 bits

Question 2. *Le numérique a progressivement supplanté l'analogique dans tous les domaines du multimédia (voix, musique, TV) parce que :*

A. Les USA ont imposé des standards numériques, en s'appuyant sur leur industrie informatique
B. L'internet a été fait au départ pour transférer des paquets de bits
C. L'électronique est devenue assez rapide pour faire les traitements numériques

Question 3. *Quel protocole utilise-t-on pour trouver une adresse de niveau 4 ?*

A. Aucun protocole.
B. DHCP
C. ARP
D. DNS
E. Une combinaison d'ARP et de DNS

Question 4. *Dans le chiffrement asymétrique vu en cours, un attaquant (Oscar) ne peut pas déchiffrer le message de Alice à Bernard parce que :*

A. Le logarithme n'est pas facile à calculer dans Z/nZ
B. Alice et Bernard possèdent un secret partagé (grâce à Diffie Hellman)
C. La fonction mathématique m-> (m^e mod n) n'est pas inversible dans Z/nZ

Question 5. *Une paire torsadée a une fréquence de coupure de 4 MHz. On souhaite faire une transmission binaire en bande de base avec ce support. Quel débit maximal théorique pourra-t-on avoir ?*

A. 4 Mbit/s
B. 8 Mbit/s
C. 2 Mbit/s
D. 3 Mbit/s

Question 6. *De quelle bande passante a-t-on besoin pour faire une transmission d'un signal à 120 Mbit/s avec une modulation QAM32 et un filtrage global de Nyquist à coefficient d'arrondi 0,5 ?*

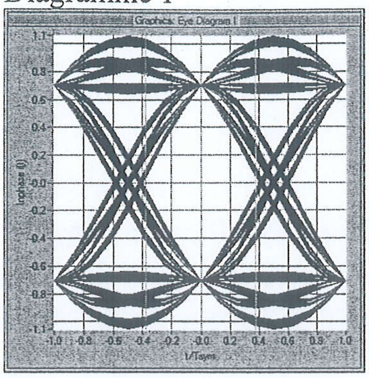
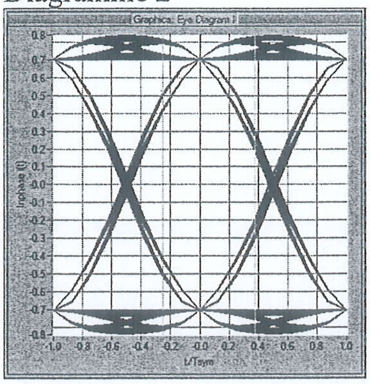
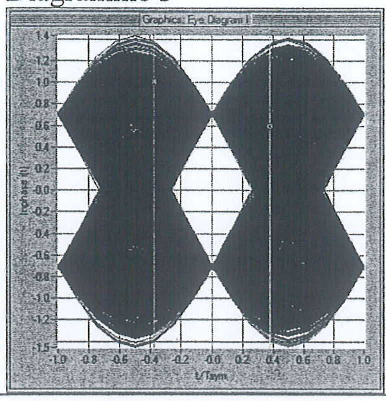
A. 18 MHz
B. 3.75 MHz
C. 5.625 MHz
D. 24 MHz
E. 36 MHz

Question 7. *Un système radio permet initialement de transmettre un signal à 2 Mbit/s avec une modulation QPSK. Le taux d'erreurs garanti est de 10^{-3}. Le système évolue pour transmettre un débit de 3 Mbit/s. On utilise alors une modulation de type PSK8. Pour garantir le même taux d'erreurs binaires, il faut modifier la puissance d'émission. De quelle valeur ?*

A. 3 dB
B. 4 dB
C. 4,8 dB
D. 1,2 dB

Question 8. *On relève le diagramme de l'œil pour une transmission QPSK, avec 3 coefficients d'arrondi 0,2, 0,5 et 0,8. Donnez le classement par ordre croissant des coefficients d'arrondi des diagrammes de l'œil que l'on obtient.*

A. diagramme 1 – diagramme 2 – diagramme 3
B. diagramme 3 – diagramme 1 – diagramme 2
C. diagramme 2 – diagramme 1 – diagramme 3
D. diagramme 3 – diagramme 2 – diagramme 1

Diagramme 1	Diagramme 2	Diagramme 3
		

Question 9. *On souhaite réaliser une transmission par fibre optique à 2,5 Gbit/s sur 2 kilomètres. Quels composants proposeriez-vous d'utiliser, qui permettrait d'optimiser à la fois les performances et le coût ?*

A. une fibre optique monomode, une diode laser à 1,3 μm, modulée en direct

B. une fibre optique multimode, une diode laser à 1,55 μm, un modulateur externe

C. une fibre optique monomode, une diode laser à 1,55 μm, un modulateur externe

D. une fibre optique multimode, une diode laser à 1,3 μm modulée en direct

Question 10. *On utilise une diode laser qui émet à 1,55 μm 5 mW de puissance optique moyenne. Elle est couplée à une fibre optique monomode standard de 10 km. Quelle puissance optique moyenne va-t-on typiquement retrouver en sortie de fibre ?*

A. 3 dBm

B. 3 dB

C. 5 dB

D. 5 dBm

Questions ouvertes à réponse courte (3 points)

Question 11. *Qu'appelle-t-on une 'technique d'accès' dans un système de télécommunication ? Quelles sont les techniques d'accès utilisées dans Ethernet et dans le Wi-Fi ? Expliquez brièvement leurs points communs et leurs différences.*

Roland Groz & Ghislaine Maury

Question 12. *Quelle est la conséquence néfaste du phénomène de trajets multiples dans un système de transmission radio ? Donnez deux exemples de systèmes de transmission où ces effets interviennent de manière importante. Citez des méthodes que l'on peut utiliser pour diminuer l'impact de ces trajets multiples.*

Exercice cryptographique (2 points)

On veut chiffrer le texte clair suivant (on omettra les caractères non alphabétiques, espaces, ponctuation, apostrophe, guillemets):
"CSE ET TELECOM, C'EST SUPER !"

Question 13. *Donner le texte chiffré selon la méthode dite de transposition, avec pour clé de chiffrement le mot : WEBMAIL.*

Exercice « tunnelling » (5 points)

L'étudiante en compilation Lexi Kal (toute ressemblance avec des personnages réels serait une fiction cinématographique) est interrogée sur une île, par un professeur retors qui lui a donné 1h30 pour trouver l'intrus dans une liste de compilations (de musique ☺). Elle doit choisir entre 3 réponses (QCM). Lexi a son ordinateur portable (lexibook). L'île est couverte par un hotspot Wifi de la société Marron, mais Lexi n'a pas de compte chez cet opérateur, ni d'argent pour acheter des crédits de connexion. *Elle va trouver un moyen détourné pour accéder quand même à Internet.* Un hotspot filtre (empêche de sortir) tous les trafics, sauf http qui est renvoyé systématiquement vers la page de paiement (serveur Web de paiement de Marron) et sauf le trafic DNS qui passe par le serveur DNS de hotspot hsp.marron.com qui transmet les requêtes hors zone aux autres serveurs de l'Internet et retransmet leurs réponses. On suppose que son cache est vide au démarrage et que hsp.marron.com est directement accessible en Wifi (cet ordinateur sert de borne d'accès sur l'île et de serveur DNS).
Lexi va communiquer avec son ami Otto Matt qui est sur le continent en exploitant ce trou dans les filtrages effectués par le hotspot.

Otto Matt gère une zone DNS appelée bde.truand.com dans laquelle il peut créer des noms; son serveur DNS est ns.bde.truand.com qui est sur un réseau local Ethernet chez Otto Matt. Lexi Kal envoie une requête DNS de type A avec le nom :
`pas_d_oeuvre_orgue_solo_dans_1=BWV_2=BuxWV_3=RV.bde.truand.com`
La réponse que fournit le serveur d'autorité est 0.0.0.n avec n=1, 2 ou 3 selon la bonne réponse.
En généralisant ce principe, Lexi va pouvoir faire passer n'importe quel protocole de couche supérieure ou égale à 3 en codant les paquets IP dans des requêtes DNS. Elle utilise un logiciel dénommé WENG-client dit de « tunnelling », qui code les paquets IP en requêtes DNS (par exemple on utilise la chaîne de caractères base64 codant le paquet IP comme un nom DNS, et les paquets IP en retour seront codés dans des réponses DNS. Le logiciel WENG-serveur s'exécute sur ns.bde.truand.com. Il reconstitue les paquets IP et les route vers leur destination finale. WENG joue le rôle d'une couche liaison (protocole point à point) juste en dessous d'IP, et offre les mêmes services à la couche supérieure.
Comme Lexi, grâce à l'aide d'Otto Matt et de ses amis à l'Ensimag (cf bonus) a trouvé la réponse à la première question, le professeur lui pose une question subsidiaire portant sur la Chine que seule Lexi (qui lit le chinois, contrairement à Otto) peut trouver sur Baidu. Elle va donc utiliser WENG pour se connecter à baidu.com (également sur un réseau Ethernet).

Question 14. *Représentez l'empilement de toutes les couches de protocoles sur chacune des quatre machines soulignées (attention, empilements « truandés », donc non conventionnels) : lexibook, hsp.marron.com, ns.bde.truand.com, baidu.com*

Question 15. *Faites figurer le chemin (à travers les couches et de machine en machine) suivi par la requête HTTP GET de Lexi vers Baidu sur le schéma précédent.*

Question 16. *Représentez l'encapsulation des messages dans la trame Wifi partant de Lexibook qui contient la requête HTTP GET de Lexi vers Baidu. On fera figurer simplement pour chaque en-tête le nom du protocole, selon le schéma amorcé ci-dessus (complétez la partie gauche du grand rectangle représentant la trame).*

	TCP	HTTP

QCM bonus (points supplémentaires, culture «mathémusique»)

Question 17. *Saurez-vous aider Lexi ? Répondez à la QCM qui lui était posée.*
« Mademoiselle Kal, comme vous le savez, depuis Pythagore, la musique est une branche des mathématiques (aux côtés de l'arithmétique, de la géométrie…). L'orgue est l'instrument de musique le plus complexe construit par les ingénieurs en Occident. N'êtes vous point férue de mathématiques aspirant à être ingénieur ? Alors laquelle des compilations suivantes ne contient aucune œuvre pour orgue seul ? »

A. BWV (Bach Werke Verzeichnis, œuvres de J-S. Bach)
B. BuxWV (Buxtehude Werke Verzeichnis, œuvres de D. Buxtehude)
C. RV (Ryom Verzeichnis, catalogue des œuvres d'A. Vivaldi, le « prêtre roux »).

N.B. Il faut bien qu'un examen de réseaux posé par un professeur retors soit bourré de sigles ☺ !

生日快乐 珠丽 JG !

Pour ceux qui ne sont pas « branchés » maths traditionnelles il vous reste, pour gagner un point, la branche plus ensimagienne : le paradoxe du compresseur, que vous étiez invités à regarder.

Question 18. *Combien de bits sont nécessaires pour transmettre en forme compressée le texte de la Bible en anglais (non compressé, cela fait 4 millions de caractères ASCII), si l'émetteur et le destinataire ont toute liberté pour définir à l'avance le compresseur qu'ils utiliseront ? Bien entendu, il faut que la compression soit sans perte d'information, quel que soit le message transmis (que ce soit la Bible ou n'importe quel autre message).*

A. 1 bit
B. $\log_2 (32 \times 10^6)$
C. $\mathrm{sqrt}(32 \times 10^6)$
D. 32×10^6

S'il vous reste du temps, justifiez ci-dessous en quelques mots votre réponse, pour bien mériter votre point bonus !

Ensimag 1re année **2010-2011**

Probabilités Appliquées
Janvier 2011

Durée 2h00 - Documents et calculatrices non-autorisés. Veuillez inscrire votre numéro de groupe sur la copie.

Exercice I (pts) - Soit $(A_i)_{i\geq 1}$ une suite d'événements indépendants tels que, pour tout $i \geq 1$,

$$P(A_i) = p\,, \quad 0 < p < 1.$$

et N une variable de loi de Poisson de paramètre λ indépendante des événements A_i.

1) Calculer la fonction génératrice $G_N(z)$ de la variable N.

2) Soit $n \geq 0$. On pose

$$S_n = \sum_{i=1}^{n} \mathbb{1}_{A_i}\,, \quad (\text{convention } \sum_{i=1}^{0} = 0).$$

Quelle est la loi de la variable S_n?

3) En utilisant la représentation suivante,

$$z^{S_N} = \sum_{n=0}^{\infty} z^{S_n} \mathbb{1}_{(N=n)}, \quad |z| < 1\,,$$

démontrer que la fonction génératrice de S_N, $H(z) = E[z^{S_N}]$, vérifie

$$H(z) = G_N(1 - p + pz)\,, \quad |z| < 1\,.$$

(On justifiera brièvement l'inversion des somme et espérance).

4) Identifier la loi de la variable aléatoire S_N.

5) Calculer l'espérance et la variance de S_N.

6) On tire N selon la loi de Poisson de paramètre λ, puis X de loi binomiale $\text{Bin}(N, p)$. Quelle est la loi de X?

Exercice II (pts) - On considère une variable X de loi de fonction de répartition $F(t)$ définie par

$$\forall t \in \mathbb{R}, \quad F(t) = \begin{cases} 0 & \text{si } t \leq 0\,, \\ t/2 & \text{si } 0 \leq t < 1\,, \\ 1 & \text{si } t \geq 1\,. \end{cases}$$

1) Rappeler le principe de simulation par inversion et écrire un algorithme de simulation par inversion pour cette loi.

2) Soient U et V deux variables aléatoires indépendantes de loi uniforme sur $(0,1)$. On considère la variable Y définie par

$$Y = \begin{cases} U & \text{si } V \leq 1/2\,, \\ 1 & \text{sinon.} \end{cases}$$

Calculer la fonction de répartition de Y. À quel type d'algorithme de simulation la construction de Y fait elle appel ?

3) Calculer l'espérance et la variance de X.

Exercice III (pts) - Soit $c > 0$ et $f(x)$ une densité de probabilité définie par

$$f(x) = c\left(x + \frac{1}{\sqrt{x}}\right)\mathbb{1}_{(0,1)}(x), \quad x \in \mathbb{R}.$$

1) Calculer c.

2) Écrire $f(x)$ comme la somme pondérée de deux densités facilement simulables par inversion et décrire les deux algorithmes d'inversion.

3) Soient U et V deux variables aléatoires indépendantes de loi uniforme sur $(0,1)$. Déduire de la question précédente que la loi de la variable aléatoire définie par

$$X = \begin{cases} \sqrt{U} & \text{si } V \leq 1/5\,, \\ U^2 & \text{sinon.} \end{cases}$$

admet $f(x)$ pour densité.

4) Déduire les valeurs de $\mathrm{E}[X]$ et $\mathrm{E}[X^2]$.

Grenoble InP - Ensimag 1ère année Janvier 2011

Théorie des langages 1

Durée : 2h.
Documents : tous documents autorisés.

Nb : le barême est donné à titre indicatif; la rigueur des preuves et la clarté de la rédaction seront prises en compte dans la notation.

Exercice 1 - Transformations d'automates (8 points)

Soit $Q_1 = \{p_0, \ldots, p_8\}$. On considère l'automate $A_1 = (Q_1, \{a,b\}, \delta_1, \{p_0\}, \{p_3, p_6\})$, où la fonction de transition δ_1 est définie par :

δ_1	a	b	ε
p_0	p_2	-	p_1
p_1	-	p_1	p_4
p_2	-	p_3	p_4
p_3	-	-	-
p_4	p_4, p_5	-	-
p_5	-	p_6	-
p_6	p_7	-	-
p_7	p_7	p_8	-
p_8	p_7	p_8	-

Soit $Q_2 = \{q_0, \ldots, q_4\}$ et soit l'automate $A_2 = (Q_2, \{a,b\}, \delta_2, \{q_0\}, \{q_3\})$, où la fonction de transition δ_2 est définie dans le tableau ci-dessous :

δ_2	a	b	ε
q_0	q_1	q_2	-
q_1	q_1	q_3	-
q_2	q_4	q_2	-
q_3	q_3	q_3	-
q_4	q_4	q_3	-

On note L_1 le langage reconnu par A_1, L_2 le langage reconnu par A_2 et $L_3 = L_1 L_2$ (c'est-à-dire que L_3 est la concaténation de L_1 et de L_2).

▷ **Question 1** *(3 points) Construire un automate minimal équivalent à A_1, en développant les étapes de la construction.*

▷ **Question 2** *(1 point) Donner une expression régulière correspondant à l'automate A_2 en résolvant le système d'équations qui lui est associé.*

▷ **Question 3** *(2 points) Constuire un automate déterministe minimal reconnaissant L_3.*

On définit le langage suivant :

$$L = \{w \in \{a,b\}^* \mid w \textbf{ ne contient pas } \text{exactement deux occurrences de la chaîne } ab\}.$$

Par exemple :

- les chaînes *ab*, *ababab*, *babaa*, *bbbab* et *babbabbaba* sont dans L ;

– les chaînes *abab*, *babaab*, *aabbabba* et *aabbaabb* ne sont pas dans L.

▷ **Question 4** *(2 points) En se basant sur la construction de la question précédente, construire un automate déterministe minimal qui reconnaît L.*

Exercice 2 - Opérations sur des langages (12 points)

Etant donné un vocabulaire V, on définit inductivement la fonction Rev : $V^* \to V^*$ de la manière suivante :

$$\begin{aligned} \mathrm{Rev}(\varepsilon) &= \varepsilon \\ \mathrm{Rev}(a.w) &= \mathrm{Rev}(w).a, \text{ où } a \in V \text{ et } w \in V^*. \end{aligned}$$

Cette fonction est étendue aux sous-ensembles de V^* :

$$\forall L \subseteq V^*,\ \mathrm{Rev}(L) = \{\mathrm{Rev}(w) \mid w \in L\}.$$

▷ **Question 5** *(2 points) On pose $V = \{a, b\}$.*
- *Calculer* $\mathrm{Rev}(abbab)$.
- *Définir en français le langage* $\mathrm{Rev}(L)$, *où* $L = \{w \in V^* \mid w \text{ se termine par } abb\}$.

Application aux automates

Reprenons le langage L ci-dessus : $L = \{w \in V^* \mid w \text{ se termine par } abb\}$.

▷ **Question 6** *(1 points) Construire un automate qui reconnaît le langage L et en déduire un automate qui reconnaît* $\mathrm{Rev}(L)$.

On souhaite généraliser la construction ci-dessus. Pour un automate $A = (Q, V, \delta, I, F)$, on définit $\mathrm{Rev}(A) = (Q, V, \delta', I', F')$ où :

$$\begin{aligned} I' &= F \text{ et } F' = I, \\ \delta' &= \{(q', a, q) \mid (q, a, q') \in \delta\}. \end{aligned}$$

▷ **Question 7** *(2 points) Démontrer par induction sur la longueur des chemins le résultat suivant : $(p_0, w_1, p_1) \cdots (p_{n-1}, w_n, p_n)$ est un chemin de trace $w_1 \cdots w_n$ dans A si et seulement si $(p_n, w_n, p_{n-1}) \cdots (p_1, w_1, p_0)$ est un chemin de trace $w_n \cdots w_1$ dans* $\mathrm{Rev}(A)$.

▷ **Question 8** *(1 point) En déduire que* $\mathcal{L}(\mathrm{Rev}(A)) = \mathrm{Rev}(\mathcal{L}(A))$.

▷ **Question 9** *(1 point) Si L est un langage hors-contexte qui n'est pas régulier, est-ce que* $\mathrm{Rev}(L)$ *peut être régulier ? Justifier.*

Application aux grammaires hors-contexte

Soit $G = (V_T, V_N, S, R)$ une grammaire hors-contexte, on définit la grammaire $\mathrm{Rev}(G) = (V_T, V_N, S, R')$, où $R' = \{A \to \mathrm{Rev}(\alpha) \mid A \to \alpha \in R\}$.

▷ **Question 10** *(1 point) On prend $V_T = \{a, b, c, d\}$, $V_N = \{S, A, B, C\}$ et on considère l'ensemble de règles suivant :*

$$\begin{aligned} S &\to AC \\ A &\to aAb \mid bB \\ B &\to bB \mid \varepsilon \\ C &\to cCdd \mid \varepsilon \end{aligned}$$

Quel est le langage engendré par $\mathrm{Rev}(G)$ *?* ***On n'attend pas une démonstration de ce résultat.***

▷ **Question 11** *(3 points) Soit G une grammaire hors-contexte, on note M le langage engendré par G. Démontrer par induction sur la longueur des dérivations que le langage engendré par* $\mathrm{Rev}(G)$ *est* $\mathrm{Rev}(M)$.

▷ **Question 12** *(1 point)* **En admettant** *que la classe des langages réguliers est engendrée par les grammaires linéaires à droite,* **démontrer** *que cette classe est engendrée par les grammaires linéaires à gauche.*

1° SESSION MAI 2011

ALGORITHMIQUE – 9 PAGES

ANALYSE APPLIQUEES – 2 PAGES

LOGIQUE POUR L'INFORMATIQUE - 4 PAGES

METHODES NUMERIQUES - 3 PAGES

POLITIQUE ECONOMIQUE - 3 PAGES

PROBABILITES APPLIQUEES – 3 PAGES

RECHERCHE OPERATIONNELLE - 2 PAGES

THEORIE DES LANGUANGES 2 – 3 PAGES

TRAITEMENT du SIGNAL - 3 PAGES

Algorithmique et structures de données : examen de première session

Ensimag 1A

Année scolaire 2010–2011

<u>Consignes générales :</u>

- Durée : 3 heures.
- Calculatrices, portables et tous instruments électroniques interdits. Livres interdits. Autres documents autorisés.
- Le barème est donné à titre indicatif.
- Les deux exercices sont indépendants et peuvent être traités dans le désordre.
- La syntaxe Ada ne sera pas un critère déterminant de l'évaluation des copies. En d'autres termes, les correcteurs n'enlèveront pas de point pour une syntaxe Ada inexacte mais compréhensible (pensez à bien commenter votre code !).
- **Merci d'indiquer votre numéro de groupe de TD et de rendre votre copie dans le tas correspondant à votre groupe.**

Exercice 1 : Pavage par des triominos *(7 pts)*

Définition 1 (triomino) *Un triomino est un objet de forme L comportant* 3 *carrés* 1×1, *voir Figure 1 (a).*

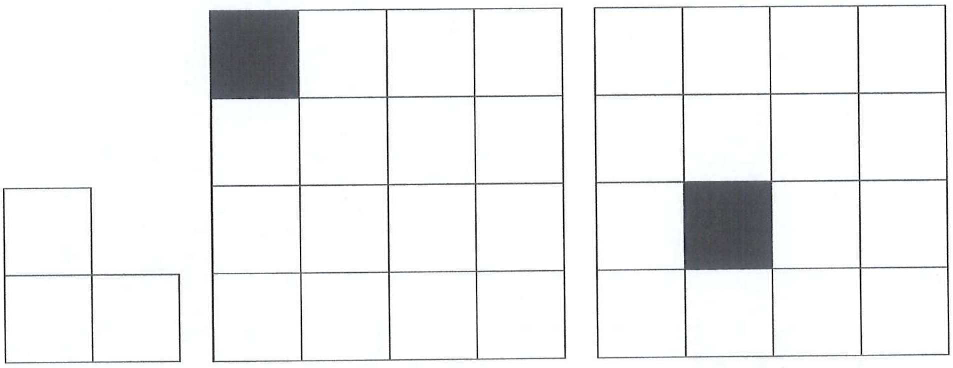
Définition 2 (échiquier à défaut) *Un échiquier* $n\times n$ *avec un carré manquant quelque part sera appelé échiquier à défaut. Pour un exemple avec* $n=4$, *voir Figure 1 (b) et (c).*

On considère le problème de déterminer si un échiquier à défaut peut être recouvert par des triominos, sachant que les triominos peuvent être tournés dans tous les sens.

Définition 3 (pavage) *Un recouvrement, une partition des cases de l'échiquier à défaut en triominos s'appelle pavage.*

Algorithme "Diviser pour régner"

Question 1 (0,5 pts) Donner un pavage des échiquiers 4×4 à défaut de la Figure 1 (b) et (c) par des triominos.



(a) (b) (c)

FIGURE 1 – (a) Triomino, (b) un échiquier 4×4 à défaut, (c) un autre échiquier 4×4 à défaut.

Question 2 (1 pts) Donner un échiquier 5×5 à défaut pour lequel il n'existe pas de pavage par triominos. Bien justifier pourquoi un tel pavage n'existe pas.

Question 3 (2,5 pts) On suppose que $n=2^k, k\in\mathbb{N}$.

1. Décrire un algorithme de type *"diviser pour régner"* qui prend en entrée l'entier n et la position du carré manquant, et produit en sortie un pavage par des triominos de l'échiquier n^2 à défaut correspondant. On pourra couper l'échiquier à défaut en sous-échiquiers, tous de même taille, et placer un triomino de manière à créer un carré manquant sur chacun des sous-échiquiers. On se contentera ici d'une description de l'algorithme en français accompagnée de dessins. Par contre, il faut bien expliquer le cas de base et le traitement récursif.
2. Analyser le coût de l'algorithme proposé.

Représentation du pavage

Définition 4 (quadtree) *Un quadtree est une structure d'arbre général, où tout nœud possède exactement 4 fils, et permettant de partitionner un espace bidimensionnel. La racine du quadtree représente l'espace entier, et chaque fils d'un nœud N représente un quart de l'espace représenté par N.*

```
type NoeudQT ;
type Quadtree is access NoeudQT ;
type NoeudQT is record
   ...
   FilsNO, FilsNE, FilsSO, FilsSE: Quadtree ;
end record ;
```

Question 4 (1,5 pts) Expliquer comment on peut utiliser un quadtree pour représenter un pavage d'un échiquier à défaut par des triominos, selon l'algorithme "Diviser pour Régner" proposé à la question précédente. Expliquer notamment :

1. quelle est la relation entre les triominos et les nœuds du quadtree ;
2. ce que représente une feuille du quadtree ;
3. quel(s) champ(s) doi(ven)t être ajouté(s) à la structure `NoeudQT`.

Question 5 (0,5 pts) Expliquer comment construire le quadtree dans l'algorithme "Diviser pour Régner" de pavage proposé à la question 3.

Question 6 (1 pts) On cherche à savoir par quel triomino est recouverte une case donnée de l'échiquier à défaut. Décrire l'algorithme de parcours du quadtree renvoyant cette information. Quelle est sa complexité en pire cas ?

Exercice 2 : Tas appariants (pairing heaps) *(16 pts)*

Cet exercice étudie l'implémentation d'une file de priorités par un "*tas appariant*" (traduction de "*pairing heap*"), une structure de données alternative aux tas binaires classiques [1], proposée en 1986 par Fredman, Sedgewick, Sleator et Tarjan. Si le coût dans le pire cas *amorti* des opérations sur un tas appariant est asympotiquement identique à celui sur un tas binaire, le coût dans le meilleur cas est bien plus intéressant.
Ainsi, une file de priorités est implémentée ci-dessous par un tas appariant de type `Arbre` (cf. définition 7), et on considére les opérations `Inserer` et `DetruireMax` spécifiées ci-dessous, où `Element` est le type des éléments. On assimile ici le type `Element` à celui des priorités : on néglige donc le problème des données attachées aux éléments, et le type `Element` est supposé muni d'un ordre total. Par ailleurs, on rappelle qu'une file peut contenir plusieurs fois un même élément.

```
procedure Inserer(A: in out Arbre; X: in Element) ;
-- insère X dans la file A

procedure DetruireMax(A: in out Arbre; Max: out Element) ;
-- requiert A non vide
-- garantit Max est un élément de priorité maximale qui a été
--          retiré de A
```

Définition 5 (coût amorti) *On considère une file donnée* `A`*. On appelle* coût amorti *le nombre "*$T(m)/m$*", où* $T(m)$ *est le nombre de comparaisons dans le* pire cas *effectuées lors d'une séquence* quelconque *de* m *appels* successifs *d'opérations de* `Insere` *et* `DetruireMax` *sur la file* `A` *depuis un état initial où* `A` *est vide.*

La structure de "*tas appariant*" se base sur une structure d'arbres binaires. On introduit donc les définitions de type ci-dessous. Dans la suite, si A est un arbre, on note $|A|$ sa taille (son nombre de nœuds).

```
type Noeud ;
type Arbre is access Noeud ;
type Noeud is record
   Etiq: Element ;
   FilsG, FilsD: Arbre ;
end record ;
```

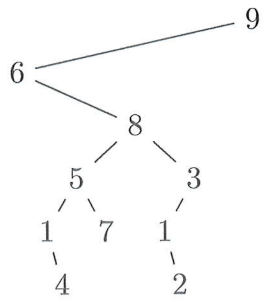
Définition 6 (arbre binaire demi-ordonné) *On dit qu'un arbre binaire* `A` *est* demi-ordonné *ssi lorsque* A *est non vide (e.g.* `A /= null`*), alors ses deux fils sont eux-mêmes* demi-ordonnés *et si pour tout sous-arbre* `B` *non vide de* `A.FilsG`*, on a* `A.Etiq >= B.Etiq`*.*

Ainsi, intuitivement, `A.FilsG` contient les éléments dont *on sait* qu'ils sont *inférieurs ou égaux* à `A.Etiq`. Alors que pour `A.FilsD`, on ne sait pas encore.

1. "tas binaires classiques" = arbres binaires parfaits partiellement ordonnés.

Définition 7 (Tas appariant) *Un* tas appariant *est défini comme un arbre binaire demi-ordonné sans fils droit.*

Autrement dit, un tas appariant est un arbre `A` demi-ordonné tel que `A = null` ou `A.FilsD = null`. Ainsi, si `A /= null`, alors `A.Etiq` contient un élément maximum. A titre d'exemple, l'arbre ci-contre est un tas appariant.



Question 7 (2 pts) Écrire une fonction `VerifTas` qui, sous la précondition que `A` a bien une structure d'arbre binaire, retourne `True` ssi `A` est un *tas appariant*. On pourra introduire une fonction récursive auxiliaire.

```
function VerifTas(A: Arbre) return Boolean ;
```

Indiquer le nombre de comparaisons effectuées par votre algorithme en fonction du nombre de nœuds $|A|$.

Opérations sur les tas appariants (avec tassage naïf)

Dans toute la suite, on ne compare les éléments des arbres binaires demi-ordonnés que par l'intermédiaire de la procédure `Compare` ci-dessous. Comme cette procédure fait exactement une comparaison entre 2 éléments, compter le nombre de comparaisons d'éléments effectuées par les opérations reviendra donc à compter le nombre d'appels à `Compare`.
Sous la précondition que `A` et `B` sont des arbres demi-ordonnés non vides, `Compare(A,B)` détache les fils droits de `A` et `B`, pour faire de `A` et `B` deux tas appariants, dont elle calcule ensuite l'union multiple (e.g. avec répétition des éléments communs) pour obtenir un tas appariant qu'elle place dans `A`.

```
procedure Compare(A: in out Arbre; B: in Arbre) is
   Minimum: Arbre ;
begin
   if A.Etiq >= B.Etiq then
      Minimum := B ;
   else
      Minimum := A ;
      A := B ;
   end if ;
   Minimum.FilsD := A.FilsG ;
   A.FilsG := Minimum ;
   A.FilsD := null ;
end ;
```

Question 8 (1 pts) En utilisant au plus un appel à `Compare` (et aucune autre comparaison d'éléments), écrire la procédure `Inserer` qui ajoute `X` comme un nouvel élément du tas appariant `A`.

```
procedure Inserer(A: in out Arbre; X: in Element) ;
```

Pour coder l'opération `DetruireMax`, on introduit l'opération `Tasser` suivante. Dans un premier temps, on l'implémente via un algo naïf qu'on va optimiser par la suite.

```
procedure Tasser(A: in out Arbre) is
   -- requiert A /= null
begin
   if A.FilsD = null then
      return ;
   end if ;
   Tasser(A.FilsD) ;
   Compare(A,A.FilsD) ;
end ;
```

Question 9 (1,5 pts) Si `A` est initialement un arbre non vide *demi-ordonné* quelconque, que fait `Tasser` ? En déduire comment utiliser `Tasser` pour implémenter `DetruireMax` sous la précondition que `A` est un tas appariant non vide. On désallouera le nœud devenu inutile.

```
procedure DetruireMax(A: in out Arbre; Max: out Element) ;
```

On considère maintenant le code suivant qui utilise les procédures précédentes pour trier un tableau par ordre croissant :

```
type Tab is array(Positive range <>) of Element ;
procedure Trier(T: in out Tab) is
   A: Arbre := null ;
begin
   for I in T'Range loop
      Inserer(A,T(I)) ;
   end loop ;
   for I in reverse T'Range loop
      DetruireMax(A,T(I)) ;
   end loop ;
end ;
```

Question 10 (3 pts) On s'intéresse au *nombre de comparaisons entre éléments* effectuées par la procédure `Trier` en fonction de la taille n du tableau d'entrée `T`.

1. Calculer en fonction de n le nombre *exact* de comparaisons effectuées lorsque le tableau est strictement croissant initialement.
2. Même chose lorsque le tableau est strictement décroissant initialement.
3. En déduire des bornes Θ en fonction de n pour les coûts dans le meilleur cas et le pire cas de la procédure `Trier`.
4. En déduire une borne Θ pour le coût amorti en fonction de m (le nombre d'opérations de la déf 5).

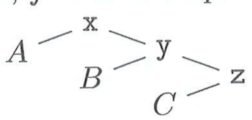
On prendra soin de bien justifier les réponses.

Tassage par paires

Fredman & co ont proposé d'effectuer `Tasser` en appliquant l'algorithme décrit ci-dessous, et qui explique le nom de la structure de données.

Définition 8 (algorithme du tassage par paires) *Comme dans la version naïve, on compare les sous-arbres de la branche complètement à droite en commençant par le nœud sans fils droit. Ce nœud* `Q` *est donc un tas appariant. Si* `Q` *a un père* `P` *et un grand-père* `G`*, on applique la séquence "*`Compare(G,P); Compare(G,Q)`*". On effectue ensuite* `Q:=G` *et on recommence tant que* `Q` *a un père et un grand-père. À la fin, soit le tas* `Q` *est la racine de l'arbre initial (auquel cas, il n'y a plus rien à faire), soit son père est cette racine. Dans ce dernier cas, on effectue un dernier* `Compare` *entre* `Q` *et son père.*

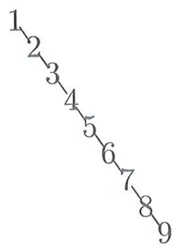
Question 11 (1 pts) On considère la situation décrite définition 8 où Q est un tas appariant, avec un père P (dont Q est le fils droit) et un grand-père G (dont P est le fils droit). Ici, on note A, B et C, les fils gauches respectifs de G, P, et Q. Et on note `x`, `y` et `z` les étiquettes respectives. L'état initial de G est donc représenté par :



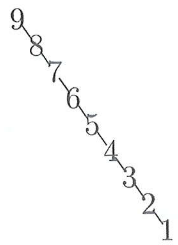
Dessiner les 4 états finaux possibles de G en fonction des résultats au 2 comparaisons successives effectuées dans "`Compare(G,P); Compare(G,Q)`".

Question 12 (1 pts) Indiquer les états obtenus si on emploie cet algorithme alternatif pour effectuer `Tasser(A)` :

1. quand `A` vaut initialement



2. puis, quand `A` vaut initialement



Question 13 (1,5 pts) Écrire une nouvelle implémentation de `Tasser` en employant cet algorithme. On introduira au préalable une procédure récursive intermédiaire appelée `TasserParPairesRec` qui s'occupe d'appliquer l'algorithme en dehors du cas particulier de la racine. Le cas particulier de la racine sera ainsi traité dans `Tasser` lui-même.

```
procedure TasserParPairesRec(A: in out Arbre) ;
-- requiert A /= null
-- garantit A réorganisé tel que  A.FilsD = null
--                            ou  A.FilsD.FilsD = null.
```

Analyse amortie pour le tassage par paires

On va faire ici les calculs de coût amorti via la méthode du potentiel dont on rappelle le principe ci-dessous. On commence par définir le potentiel utilisé ici.

Définition 9 (rang d'un arbre binaire) *Le* rang $r(A)$ *d'un arbre* A *est le réel positif ou nul défini par :*
- $r(A) = 0$ *si* $A = \texttt{null}$.
- *et,* $r(A) = \log_2 |A|$ *si* $A \neq \texttt{null}$.

Définition 10 (potentiel d'un arbre binaire) *Le potentiel* $\phi(A)$ *d'un arbre* A *est un réel positif ou nul défini par récurrence sur les arbres via :*
- $\phi(A) = 0$ *si* $A = \texttt{null}$.
- *et,* $\phi(A) = r(A) + \phi(B) + \phi(C)$ *si* A *est un arbre non vide dont les deux fils sont* B *et* C *(l'ordre des fils n'a ici aucune importance).* [2]

Définition 11 (coût compensé d'une instruction) *Le coût compensé* $cc(S)$ *d'une instruction* S *modifiant l'arbre* A *est défini par*

$$cc(S) = cr(S) + \phi(A) - \phi_0(A)$$

où $cr(S)$ *est le coût réel de* S *(ici le nombre de comparaisons),* $\phi(A)$ *le potentiel de* A *dans l'état final de* S *et* $\phi_0(A)$ *son potentiel dans l'état initial.*

L'intérêt du coût compensé est qu'il vérifie les propriétés suivantes :

1. Pour une séquence d'instructions "$S; S'$" on a

$$cc(S; S') = cc(S) + cc(S')$$

   En effet, en notant $\phi_1(A)$ le potentiel dans l'état final de S, on a $cc(S) = cr(S) + \phi_1(A) - \phi_0(A)$ et $cc(S') = cr(S') + \phi(A) - \phi_1(A)$. Par ailleurs, $cr(S; S') = cr(S) + cr(S')$.

2. Si dans l'état initial de S, A est l'arbre vide, alors

$$cr(S) \leq cc(S)$$

   En effet, $cr(S) = cc(S) + \phi_0(A) - \phi(A)$ avec $\phi_0(A) = 0$ et $\phi(A) \geq 0$.

Concrètement, **on va montrer** ici que le coût compensé des opérations sur le tas appariant A est **majoré par** "$1+3.r(A)$" où $r(A)$ représente le rang de A après l'appel de l'opération. Comme $|A| \leq m$, il s'en suit que le coût amorti est en $\mathcal{O}(\log_2\ m)$.

Question 14 (2 pts) Dans l'opération `Inserer(`A`,X)`, on note $\phi(A)$ le potentiel de A après l'appel et $\phi_0(A)$ celui avant appel.

1. Montrer que $\phi(A) - \phi_0(A) = r(A)$, où $r(A)$ est la valeur après l'appel.
2. En déduire que le coût compensé de `Inserer` est bien majoré par $1 + 3.r(A)$.

Les questions 15 et 16 suivantes ont pour but de montrer par récurrence sur A que le nombre de comparaisons compensé de `TasserParPairesRec(A)` est majoré par $3.r(A)$.

---

2. D'après ce qui précède, on a donc ici $r(A) = \log_2(1 + |B| + |C|)$.

Question 15 (1 pts) Justifier que pour montrer cette propriété par récurrence, il suffit de montrer que, dans la transformation effectuée à la question 11, on a :

$$2 + \phi(G) - \phi_1(G) + 3.r_1(Q) \leq 3.r(G)$$

avec
- $\phi(G)$ et $r(G)$ qui représentent les valeurs après la transformation ;
- $\phi_1(G)$ et $r_1(Q)$ qui représente les valeurs avant la transformation.

On justifiera soigneusement en utilisant les notations ϕ_0 et r_0 pour les valeurs avant l'appel à `TasserParPairesRec`.

Question 16 (1 pts) On montre maintenant l'inégalité de la question 15.

1. Expliquer pourquoi on peut se contenter de montrer cette inégalité pour un seul des 4 cas.
2. Dans la suite, on considère le cas $\mathtt{x} < \mathtt{y}$ et $\mathtt{y} < \mathtt{z}$. Expliquer pourquoi il suffit de montrer que
$$2 + r_1(\mathtt{z}) + r(\mathtt{x}) \leq 2.r(\mathtt{z})$$
(ici, on utilise les étiquettes pour dénoter les nœuds).
3. Conclure en remarquant que pour tous réels a et b, $4.a.b \leq (a+b)^2$.

Question 17 (1 pts) Conclure :

1. Terminer l'analyse amortie en majorant le coût compensé de `DetruireMax`.
2. En déduire les bornes en Θ pour le pire cas et le meilleur cas de `Trier` (cf. question 10) lorsqu'on utilise le tassage par paires.

Grenoble INP **Analyse Appliquée**
ENSIMAG

Examen du 16 mai 2011, durée 2 heures

Exercice 1

Soit H un espace de Hilbert sur $\mathbb{R}$ et soit F un s.e.v fermé de H. Pour $u \in H$ on note Pu la projection orthogonale de u sur F. On définit une application $f : H \to \mathbb{R}_+$ par $f(u) = \|Pu\|$, où $\|.\|$ désigne la norme sur H. Déterminer l'ensemble des points de H où f est différentiable et donner sa différentielle.

Exercice 2

Soit $l^2(\mathbb{R}) = \{(x_n)_{n\in\mathbb{N}} \in \mathbb{R}^{\mathbb{N}}, \sum_{n=0}^{+\infty} x_n^2 < +\infty\}$ qui, muni du produit sclaire $< x, y >= \sum_n x_n y_n$, est un espace de Hilbert. On définit

$$F = \{(x_n)_{n\in\mathbb{N}} \in \mathbb{R}^N, \exists p \in \mathbb{N} \text{ tel que } \forall n \geq p \; x_n = 0\}$$

1. Montrez que F n'est pas fermé
2. Montrez que F est dense dans $l^2(\mathbb{R})$
3. Montrez qu'il existe x appartenant à $l^2(\mathbb{R})$ tel que :

$$\forall y \in F \quad \forall z \in F^\perp \quad x \neq y + z.$$

Exercice 3 Soit H un espace de Hilbert sur $\mathbb{R}$. On note $< .,. >$ le produit scalaire et $\|.\|$ la norme associée. Soit $K \subset H$ un convexe fermé, borné et non vide et soit T une applicattion de K dans K telle que:

$$\forall x, y \in K \quad \|Tx - Ty\| \leq \|x - y\|.$$

1. Soient $\varepsilon > 0$ et $z \in K$. Montrer que l'équation $\varepsilon(x - z) + x - Tx = 0$ admet une solution unique dans K notée x_ε.

2. On admet le résultat suivant : il existe $x^* \in K$ et une suite ε_n avec $\varepsilon_n > 0$ pour tout n et $\lim_{n\to+\infty} \varepsilon_n = 0$ tels que

$$\forall y \in H \quad \lim_{n\to+\infty} < x_{\varepsilon_n} - x^*, y >= 0$$

i) Montrer que $\lim_{n\to+\infty} \|x_{\varepsilon_n} - Tx_{\varepsilon_n}\| = 0$

ii) Soit P_K l'opérateur de projection sur K. Montrer que pour tout $u \in H$, on a

$$< x_{\varepsilon_n} - u - (Tx_{\varepsilon_n} - TP_K u), x_{\varepsilon_n} - u >\geq 0,$$

et en déduire que

$$< -u + TP_K u, x^* - u >\geq 0$$

iii) Montrer que $< -x^* + TP_K x^*, v >\leq 0$ pour tout $v \in H$.

3. Montrer que T admet un point fixe dans K.
4. Considérons l'espace vectoriel

$$E = \{s = (s_n); \; s_n \in \mathbb{R} \; \forall n \in \mathbb{N}, \; \lim_{n\to+\infty} s_n = 0\},$$

muni de la norme $\|s\|_\infty = \sup_{n\in\mathbb{N}} |s_n|$. Soient $K = \{s \in E; \|s\|_\infty \leq 1\}$ et T l'opérateur défini par $Ts = r$ avec $r = (r_n)$ et $r_0 = 1$, $r_n = s_{n-1}$, pour $n \geq 1$. T admet-il un point fixe? Quelle conclusion en tirez-vous?

Exercice 4 On rappelle que $\widehat{\Delta}_a = \frac{1}{a}\Delta_{\frac{1}{a}}$.

1. Montrer que pour toute fonction $\phi \in \mathcal{S}(\mathbb{R})$ on a :

$$\sum_{n\in\mathbb{Z}} \phi(n) = \sum_{n\in\mathbb{Z}} \hat{\phi}(n)$$

2. En déduire alors que pour tout $\phi \in \mathcal{S}(\mathbb{R})$ on a :

$$\sum_{n\in\mathbb{Z}} \phi(\frac{n}{a}) = a \sum_{n\in\mathbb{Z}} \hat{\phi}(na) \ (a \neq 0)$$

3. Montrer que $e^{-\pi t^2} \in \mathcal{S}(\mathbb{R})$. En déduire alors que

$$\sum_{n\in\mathbb{Z}} e^{-\pi n^2 t} = \frac{1}{\sqrt{t}} \sum_{n\in\mathbb{Z}} e^{-\frac{\pi n^2}{t}} \ (t > 0)$$

EXAMEN DE "LOGIQUE POUR L'INFORMATIQUE"

Ensimag 1ère Année - mai 2011

DURÉE: 3 heures.

SEULS DOCUMENTS AUTORISÉS: Poly "Une approche simple de la logique pour l'Informatique et la formalisation de l'inférence" et notes personnelles.

- S.V.P. **avant de commencer** lisez les 7 points ci-dessous.

- Cet examen comporte 10 questions. Toutes les questions sont indépendantes.
- L'ordre des questions n'a, en principe, aucun rapport avec leur difficulté.
- *S.V.P. rédigez vos réponses de façon similaire à celle des corrigés du poly.*
- **S.V.P. séparez** (par une ligne horizontale de la largeur de la feuille) et **numérotez clairement** les réponses aux différentes questions.
- **S.V.P.** respectez le nombre de lignes suggérées pour les réponses.
- Les réponses à l'aide des méthodes ***autres*** que celles demandées explicitement seront ***ignorées***.
- On tiendra compte dans la correction de la clarté, de la simplicité et de la concision des réponses ainsi que de la qualité de la présentation.

Question 1

(Voir exercice 29, poly pages 63-64 et 313)

Soit un système formel $\mathcal{S} =< \mathcal{L}, \mathcal{R}, \mathcal{A} >$

avec $\mathcal{L}$: le même que pour $\mathcal{S}_1$ (voir poly page 57).

On suppose les schémas axiomes de $\mathcal{A}$ représentés sous forme de termes (comme il est fait en page 89 du poly pour *(A1)* et en page 90 du poly pour *(A2)*).

a) Supposez que pour l'un des schémas d'axiomes, disons A_i il existe un autre, dans $\mathcal{A}$, disons A_j tel que

$(\star)$ $\qquad \sigma A_j = A_i$, où $\sigma = \text{upg}(A_j, A_i)$.

A_i est-il indépendant ? Répondre par **OUI** ou par **NON**.

b) Supposez maintenant que pour l'un des schémas d'axiomes, disons A_i, la condition $(\star)$ n'est vérifiée pour **aucun** des autres schémas d'axiomes dans $\mathcal{A}$.

A_i est-il indépendant ? Répondre par **OUI** ou par **NON NÉCESSAIREMENT**.

Question 2

Étant donné les deux clauses ci-dessous (comme d'habitude f, g, h, i, k dénotent des symboles de fonction et x, y, z, u, v, w des variables) :

C_1 : $P(x,\ g(x),\ y,\ h(x,y),\ z,\ k(x,y,z))$

C_2 : $\neg P(u,\ v,\ i(v),\ w,\ f(v,w),\ x)$

Peut-on appliquer la règle de résolution ?

a) Si la réponse est 'non' expliquer pourquoi.

b) Si la réponse est 'oui' donner la résolvante et la substitution correspondante.

Question 3

Considérer le raisonnement suivant :

Tous les A sont B
Tous les B sont C

Quelques A sont C

Après l'avoir formalisé en utilisant la traduction suggérée en Remarque 7 de page 26 du poly, pouvez-vous ?

a) Essayer de vérifier sa correction ***en utilisant la méthode des tableaux sémantiques ?*** (on vous demande de construire l'arbre correspondant jusqu'à une profondeur de 3).

b) Voyez-vous un problème ? ***Maximum 3 lignes.***

c) Voyez-vous une solution ? ***Maximum 3 lignes.***

Question 4

Étant données les formules i) et ii) ci-dessous, où x, y, z, u dénotent des variables, a une constante, f un symbole de fonction binaire, A et P des symboles de prédicat :

i) $\forall x P(a,x,x)$

ii) $\forall x \forall y \forall z \forall u\ [\ (A(u) \wedge P(x,y,z)) \Rightarrow P(f(u,x),y,f(u,z))\ ]$

On a choisi pour l'interpréter :

$D = \mathbb{N} \bigcup \{$ listes finies de naturels $\}$

$P(x,y,z) \mapsto$: z est la concatenation ("append") des listes x et y dont les éléments sont des nombres naturels.

Donner les interprétations de a, f et A, de façon à rendre **vrai** la formule i) ainsi que l' antécedent (i.e. les formules de gauche) et le conséquent (i.e. la formule de droite) de '$\Rightarrow$' dans ii), c'est-à-dire d'obtenir un **modèle** de i) et ii). *On ne s'intéresse pas au cas où l'antécedent de '$\Rightarrow$' dans la formule ii) est faux.*

Question 5

Considérer le raisonnement suivant

$\forall x[H(x) \Rightarrow A(x)]$
$\exists y \forall v[A(v) \Rightarrow L(v,y)]$

$\exists u \forall z[H(z) \Rightarrow L(z,u)]$
Pouvez-vous prouver la correction (ou incorrection) de ce raisonnement
en utilisant la méthode de résolution ?

Question 6

Considérer le raisonnement suivant

$\forall x[H(x) \Rightarrow A(x)]$
$\exists y \forall v[A(v) \Rightarrow L(v,y)]$

$\exists u \forall z[H(z) \Rightarrow L(z,u)]$
Pouvez-vous prouver la correction (ou incorrection) de ce raisonnement
en utilisant la méthode des tableaux sémantiques ?

Question 7

Le schéma :
$P_0 \wedge P_n \wedge \bigwedge_0^n (P_i \Leftrightarrow \neg P_{i+1})$
a) Est-il (in-)satisfaisable pour $n = 2$?
b) Est-il (in-)satisfaisable pour $n = 3$?
Répondre ***en utilisant la méthode des tableaux sémantiques***.

Question 8

Le schéma :
$P_0 \wedge P_n \wedge \bigwedge_0^n (P_i \Leftrightarrow \neg P_{i+1})$
Est-il (in-)satisfaisable pour $n = 3$?
Répondre ***en utilisant la méthode de résolution***.

Question 9

Il s'agit de vérifier sur un exemple que l'ordre d'application des règles de la méthode de Davis et Putnam peut être quelconque, par exemple sur l'ensemble des clauses ci-dessous les règles **R-1b)** et **R-3** s'appliquent. Vérifier qu'en en appliquant l'une ou l'autre, jusqu'à ce que plus aucune règle ne soit applicable on arrive à la même conclusion.

$\{\neg E \vee A,\ D,\ \neg A,\ A \vee C \vee E,\ A \vee \neg D \vee E,\ \neg B \vee C \vee E,\ \neg B \vee \neg D \vee E,\ \neg C \vee E\}$

Question 10

Il s'agit de *prouver* que dans la méthode de Davis et Putnam l'ordre d'application des règles **R-1b)** et **R-3** (quand toutes les deux sont applicables) est indifférent : c'est à dire la détection de l'insatisfaisabilité ou de la satisfaisabilité n'est pas affectée par l'ordre choisi.

Vous pouvez considérer comme ensemble de clauses générique :

$\{L,\ L^c \vee \alpha,\ P \vee \beta,\ P^c \vee \gamma,\ \Delta\}$

avec

$L,\ L^c,\ P,\ P^c$: littéraux
$\alpha,\ \beta,\ \gamma$: clauses
Δ : ensemble de clauses
Maximum 10 lignes.

Fin-Examen

ENSIMAG 1ère année (2010-11)

Méthodes Numériques

Examen du 11 Mai 2011

Durée : 3h.

Les seuls documents autorisés sont les notes du cours et des travaux dirigés de méthodes numériques. Les calculatrices sont interdites.

La qualité de la rédaction et de la présentation sera prise en compte pour l'évaluation des copies.

Exercice 1

On considère une matrice réelle $A = (a_{ij})_{1 \le i,j \le n}$ telle que pour tout $j = 1, \ldots, n$

$$|a_{jj}| > \sum_{i \neq j} |a_{ij}|.$$

1. Montrer que A est inversible.

2. Montrer qu'il existe $L \in M_n(\mathbb{R})$ triangulaire inférieure de diagonale unité et $U \in M_n(\mathbb{R})$ triangulaire supérieure telles que $A = L\,U$.

3. La matrice A suivante admet-elle une factorisation $A = LU$? Calculer cette factorisation s'il y a lieu.

$$A = \begin{pmatrix} 2 & 1 & 0 \\ 0 & -3 & -1 \\ -1 & 1 & -2 \end{pmatrix}.$$

Exercice 2

Soit $A \in M_n(\mathbb{R})$ une matrice inversible. On considère la suite de matrices $(B_k)_{k \ge 0}$ définie par

$$B_{k+1} = B_k\,(2\,I - A\,B_k), \quad k \ge 0, \tag{1}$$

la matrice $B_0 \in M_n(\mathbb{R})$ étant fixée.

1. Montrer que si la suite (B_k) converge et que sa limite est une matrice inversible, alors cette limite vaut A^{-1}.

2. Vérifier que $I - A\,B_{k+1} = (I - A\,B_k)^2$.

3. Montrer que si $\rho(I - A\,B_0) < 1$ alors $\lim_{k \to +\infty} B_k = A^{-1}$. A quelle vitesse s'effectue cette convergence ?

Exercice 3

Soit $A \in M_n(\mathbb{R})$ une matrice symétrique définie positive (on suppose que A n'est pas un multiple de la matrice identité). On note $\alpha > 0$ la plus petite valeur propre de A et β sa plus grande valeur propre. Etant donné $b \in \mathbb{R}^n$, on considère la solution $x \in \mathbb{R}^n$ du système linéaire

$$A\,x = b. \tag{2}$$

On souhaite calculer x en utilisant une méthode itérative de la forme

$$x_{k+1} = x_k + w_{k+1}\,r_k, \quad k \geq 0, \tag{3}$$

où l'on note $r_k = b - A\,x_k$, et où les coefficients w_k $(k \geq 1)$ sont des paramètres strictement positifs qui seront déterminés par les valeurs propres de la matrice A.

1. La méthode du gradient à pas optimal rentre-t-elle dans cette classe de méthodes itératives ?

2. On considère le polynôme q_k défini de la manière suivante pour tout entier $k \geq 1$

$$q_k(\lambda) = (1 - \omega_k \lambda)\,(1 - \omega_{k-1}\lambda)\,\ldots\,(1 - \omega_1 \lambda)$$

et on définit la matrice $q_k(A) = (I - \omega_k A)\,(I - \omega_{k-1}A)\,\ldots\,(I - \omega_1 A)$. Montrer que pour tout $k \geq 1$ on a $r_k = q_k(A)\,r_0$.

3. En déduire que pour tout $k \geq 1$ on a

$$\|r_k\|_2 \leq M\,\|r_0\|_2,$$

où $\|\;\|_2$ désigne la norme euclidienne et

$$M = \operatorname*{Max}_{\lambda \in [\alpha,\beta]} |q_k(\lambda)|. \tag{4}$$

Dans la suite du problème, on se donne un entier $N \geq 1$ et une permutation $(k_1, k_2, \ldots, k_{N-1}, k_N)$ de $(1, 2, \ldots, N-1, N)$. On fixe

$$\omega_i = 2\left[\beta + \alpha - (\beta - \alpha)\,\cos\left(\frac{2\,k_i - 1}{2N}\,\pi\right)\right]^{-1}, \quad i = 1, \ldots, N. \tag{5}$$

Remarques : Ce choix de coefficients ω_i minimise la constante (4) pour $k = N$. Par ailleurs, bien que le polynôme q_N soit indépendant de la permutation définissant l'ordre des coefficients ω_i, le schéma (3) nécessite un choix astucieux de permutation afin d'éviter des instabilités numériques dûes aux erreurs d'arrondis lors des calculs sur machine.

4. Montrer l'identité

$$q_N(\lambda) = \frac{T_N(\mu - \frac{2\lambda}{\beta-\alpha})}{T_N(\mu)},$$

où T_N désigne le Nième polynôme de Tchebychev (voir les rappels en fin d'exercice) et

$$\mu = \frac{\beta + \alpha}{\beta - \alpha}.$$

5. On note κ_2 le conditionnement de la matrice A pour la norme euclidienne et on introduit

$$\sigma = \frac{\sqrt{\kappa_2} - 1}{\sqrt{\kappa_2} + 1}.$$

Vérifier que $\mu = \frac{1}{2}(\sigma + \sigma^{-1})$.

6. En déduire que $T_N(\mu) = \frac{1}{2}(\sigma^N + \sigma^{-N})$.

7. Montrer que

$$\|r_N\|_2 \leq \frac{2\,\sigma^N}{1+\sigma^{2N}}\,\|r_0\|_2.$$

8. Etant donné $\epsilon > 0$, déterminer une borne $N(\epsilon)$ garantissant que $\|r_N\|_2 \leq \epsilon\,\|r_0\|_2$ pour tout $N \geq N(\epsilon)$. Comment cette borne se comporte-t-elle lorsque $\kappa_2 \to +\infty$?

Dans la méthode itérative (3), on fixe maintenant $N \geq 1$ et on considère la suite $(\omega_i)_{i\geq 1}$ de période N définie par (5) lorsque $1 \leq i \leq N$.

9. Montrer que pour tout $p \geq 0$ on a

$$\operatorname*{Max}_{1\leq i\leq N} \|r_{i+pN}\|_2 \leq C_N \left(\frac{2\,\sigma^N}{1+\sigma^{2N}}\right)^p \|r_0\|_2,$$

où

$$C_N = \operatorname*{Max}_{1\leq i\leq N,\lambda\in[\alpha,\beta]} |q_i(\lambda)|.$$

10. Etudier la convergence du schéma (3) quand $k \to +\infty$.

11. Comment peut-on adapter cette méthode itérative si on ne connaît pas exactement α et β mais si on en a des estimations $\tilde{\alpha} \leq \alpha$ et $\beta \leq \tilde{\beta}$?

Rappels sur les polynômes de Tchebychev.

Les polynômes de Tchebychev T_k ($k \geq 0$) sont définis par la relation de récurrence

$$T_{k+1}(\xi) = 2\,\xi\,T_k(\xi) - T_{k-1}(\xi),$$

avec $T_0 = 1$, $T_1 = \xi$. Ils vérifient par ailleurs pour tout $z \in \mathbb{C}$

$$\cos(k\,z) = T_k(\cos z)$$

où $\cos z = \frac{1}{2}(e^{i\,z} + e^{-i\,z})$.

ENSIMAG - 1ère année - 2010/2011

Politiques économiques

DS du lundi 16 mai 2011

Durée : 2 heures
Documents autorisés

Traitez **au choix** l'un des deux sujets suivants.

Pour chacun d'eux, veillez à mobiliser au mieux vos connaissances ainsi qu'à soigner l'organisation des développements, la clarté de l'argumentation et la qualité de la rédaction.

Sujet 1

Dans une dépêche datée du 23-02-2011, l'AFP (Agence France-Presse) rapporte l'essentiel d'un entretien accordé par Paul Krugman au sujet de l'avenir de l'Euro :

L'Américain Paul Krugman, lauréat du prix Nobel d'économie 2008, estime que les « *architectes de l'Euro ont décidé d'ignorer les difficultés inhérentes à une monnaie unique* » parlant d'une « *tragédie* ».

Les inventeurs de l'Euro « *ont surtout ignoré les mises en garde contre l'absence d'institutions nécessaires pour que fonctionne la monnaie unique* ».

« *Le résultat n'est pas seulement une tragédie pour l'Europe mais pour le monde entier* », ajoute-t-il en évoquant les déboires actuels de l'Irlande et de l'Espagne alors que le succès de l'union monétaire devrait être l'un des enjeux les plus importants du projet européen. L'Euro conduit au contraire les pays à risques tels que la Grèce ou l'Espagne à une « *orgie d'emprunts financée par le boom des exportations allemandes* », selon lui.

Paul Krugman évoque plusieurs scénarios pour sortir de la crise de l'Euro. Il propose notamment aux responsables de la zone Euro « *d'imiter la zone Dollar, à savoir les Etats-Unis* », en la transformant en union fiscale qui permettrait des transferts financiers d'une zone à l'autre. Citant l'exemple du Nevada, aussi gravement touché par la crise que l'Irlande, l'économiste explique que ses effets en ont été beaucoup moins lourds pour cet Etat de l'ouest américain parce qu'une « *grande partie des dépenses publiques (telles que les retraites) viennent de Washington* », c'est-à-dire de l'Etat fédéral. A l'inverse, en Irlande, « *les pensions de retraites et les dépenses de santé figurent sur la liste des coupes budgétaires de l'Etat* », constate-t-il.

Aux Etats-Unis, « *les Etats qui ne sont pas encore en faillite paient pour les Etats qui sont en faillite* », résume Paul Krugman, qui émet des doutes sur la capacité de l'Euro à fonctionner encore longtemps si un mécanisme similaire n'est pas adopté.

Travail demandé :

Rédigez une note explicitant la position de Paul Krugman.

Indications :

Imaginez par exemple que vous ayez à expliquer, à quelqu'un qui en sait moins que vous, ce qu'il faut savoir pour comprendre les tenants et les aboutissants des propos formulés par Krugman, voire pour en proposer une critique.
La note comportera une introduction, des développements et une conclusion.

Sujet 2

Dans un article publié le 31 mars 2011 sur le site www.project-syndicate.org, et intitulé « *Anatomie de la relance lente* » ; J. Bradford DeLong, ancien Secrétaire assistant au Trésor américain et Professeur d'Economie à Berkeley, conclut sa réflexion par la formule suivante :

« *L'Amérique ne peut donc simplement se contenter d'une reprise de la demande ; il lui faut aussi un ajustement structurel. Malheureusement, le Marché ne peut à lui seul produire une relance rapide de la demande. Et il ne peut produire aucun ajustement structurel tant qu'une reprise de la demande ne sera pas déjà largement engagée.* »

Travail demandé :

Commentaire libre.

Indications :
Votre commentaire comportera une introduction, des développements et une conclusion.
L'introduction devra explicitement présenter, en la justifiant, l'idée principale que vous aurez choisie comme fil conducteur de vos propos, et l'énoncé de la manière avec laquelle vous allez la développer.

Ensimag 1^{re} année **2010-2011**

Probabilités Appliquées 2
Mai 2011

Durée 3h00 - Documents et calculatrices non-autorisés. Veuillez inscrire votre numéro de groupe sur la copie.

Exercice I (3 pts) - On considère une suite (X_n) de variables aléatoires indépendantes de loi uniforme sur $(0,1)$. Soit a un nombre tel que $0 < a < 1$. On définit la variable aléatoire $N(a)$ de sorte que

$$N(a) = \min\{n \geq 1\,,\, X_1 + \cdots + X_n > a\}.$$

1) Soit $x \in (0,1)$. En discutant selon les valeurs $x > a$ ou $x \leq a$, donner une expression de l'espérance conditionnelle $\mathrm{E}[N(a) \mid X_1 = x]$ ne laissant plus apparaître le conditionnement (ne pas chercher à calculer $\mathrm{P}(N(a) > n | X_1 = x)$).

2) Déduire de la question précédente que, pour tout $a \in (0,1)$, nous avons

$$\mathrm{E}[N(a)] = 1 + \int_0^a E[N(x)]dx.$$

3) Résoudre l'équation précédente pour (re)trouver l'expression de $\mathrm{E}[N(a)]$.

Exercice II (4 pts) - Soit X et Y deux variables aléatoires indépendantes de loi de densité

$$\forall x \in \mathbb{R}, \quad g(x) = xe^{-x}\mathbb{1}_{\mathbb{R}_+}(x).$$

1) Montrer que la loi de la variable aléatoire

$$Z = Y/X$$

admet pour fonction de répartition

$$\forall t \in \mathbb{R}_+, \quad F_Z(t) = 1 - \frac{1+3t}{(1+t)^3}\,.$$

2) Démontrer que la variable $T = 1/(1+Z)$ admet pour densité

$$\forall t \in (0,1), \quad f(t) = 6t(1-t)$$

3) Justifier le fait que la loi de densité g peut être obtenue comme la loi de la somme de deux variables de loi exponentielle indépendantes de paramètre 1. En déduire un algorithme de simulation de la loi de densité f.

Exercice III (9 pts) - Soit (X, Y) un couple de variables aléatoires de densité

$$\forall (x, y) \in \mathbb{R}^2 , \quad f_{(X,Y)}(x, y) = A \exp(-\frac{2x^2 - 2cxy + y^2}{4 - 2c^2})$$

où A est une constante positive et c une constante réelle.

1) Montrer que le couple (X, Y) est gaussien, de moyenne $m = (0, 0)$ et de matrice de covariance

$$K = \begin{pmatrix} 1 & c \\ c & 2 \end{pmatrix}$$

si et seulement si $|c| < \sqrt{2}$. En déduire la valeur de la constante A.

2) Déterminer les lois marginales de X et de Y. Quelle est la valeur de $cov(X, Y)$? Les variables X et Y sont-elles indépendantes ?

3) Soit $\alpha \in \mathbb{R}$. On pose

$$\begin{cases} Z &= X + \alpha Y \\ T &= Y \end{cases}$$

Montrer que le couple (Z, T) est gaussien et décrire sa moyenne et sa matrice de covariance. Sous quelle condition portant sur α, les variables Z et T sont-elles indépendantes ?

4) Écrire un algorithme de simulation pour le couple (X, Y). On prendra soin de démontrer que le couple en sortie de cet algorithme est bien de densité $f_{(X,Y)}$.

5) Déterminer la loi conditionnelle de Y sachant $X = x$.

6) À la suite de la question précédente, écrire un nouvel algorithme de simulation du couple (X, Y).

Exercice IV (4 pts) - Une urne contient une boule noire et une boule blanche. On effectue dans l'urne une série de tirages avec remise selon la règle suivante. Lorsque l'on tire une boule blanche, on remet dans l'urne une boule noire. Lorsque l'on tire une boule noire, on remet dans l'urne une boule noire avec la probabilité p, ou une boule blanche avec la probabilité $q = 1-p$ $(0 \leq p \leq 1)$. On note π_n la probabilité pour que l'urne contienne une boule noire et une boule blanche à l'issue du n^e tirage après remise.

1) Soit X_n le nombre de boule(s) noire(s) contenue(s) dans l'urne à l'issue du n^e tirage après remise. Décrire la loi de la variable X_1.

2) Décrire la loi conditionnelle de la variable X_n sachant $X_{n-1} = 0$, 1 ou 2 (3 cas).

3) Montrer que la probabilité π_n converge vers une constante $\pi(p)$ que l'on calculera (ne pas chercher à calculer π_n).

4) Justifier le fait que la vitesse de convergence est d'ordre linéaire, c.à.d,

$$\frac{1}{n} \log |\pi_n - \pi(p)| \to \log(\lambda), \quad \lambda > 0$$

et calculer la constante λ.

EXAMEN du 11 mai 2011. Durée: 3h. 2 pages numérotées.
Documents manuscrits ou polycopiés autorisés. Aucun livre.

Il est important de bien expliquer ce que vous faites. Il n'est pas nécessaire de tout faire pour avoir une bonne note, par contre il sera enlevé des points pour une rédaction trop succincte.

Veuillez noter sur votre copie le nom de votre enseignant : BIENIA ou NADDEF ou SZIGETI

EXERCICE 1:

La compagnie Catermachin fabrique des gros engins, les uns de la ligne E pour l'industrie forestière et les autres de la ligne F pour les travaux publics. Deux modèles les E9 et F9 sont fabriqués dans les mêmes ateliers A et B. Le service commercial estime que l'on pourra vendre dans le mois qui suit autant de E9 et F9 que l'on peut produire. Une unité de E9 rapporte 5000 € et une de F9 rapporte 4000 €. La production passe par les ateliers A et B. Le nombre d'heures nécessaire par unité dans chaque atelier selon le modèle et le nombre total d'heures disponible pour chaque atelier sont détaillés dans le tableau suivant:

atelier	heures par E9	heures par F9	temps disponible
A	10	15	150
B	20	10	160

Le test de qualité est fait dans un troisième atelier. Un E9 a besoin de 30 heures de test et un F9 a besoin de 10 heures. Un accord syndical stipule que le temps utilisé dans les tests de qualité ne peut être inférieur à 135 heures. Les instances dirigentes ont décidé, pour des raisons de positionnement sur le marché, que l'on doit fabriquer au moins un F9 pour chaque trois E9 fabriqués.

Enfin un important client a commandé un total de 5 E9 et F9, quelle que soit la combinaison. Ce client étant stratégique, il doit être satisfait.

a) Ecrire un programme linéaire qui permet de maximiser le bénéfice tout en satisfaisant toutes les contraintes imposées.
b) Ecrire son dual.
c) Montrer que fabriquer 4,5 unités de E9 et 7 de F9 est une solution optimale.
d) Donner une solution optimale du dual. Est-elle unique?
e) Donner une explication économique des valeurs de cette solution du dual. Certaines des valeurs trouvées sont nulles. Si elles n'étaient pas nulles quel serait leur signe et que voudrait-elle dire?

EXERCICE 2 : Considérons le projet simple ci-contre:

a) Quelle est la durée minimum du projet?
b) Donner la date de début au plus tôt et au plus tard pour chaque tâche qui permet de terminer dans cette durée minimum.
c) Donner la liste des tâches critiques et la marge des autres.
d) Supposons que l'on peut diminuer la durée d'une seule tâche. Quelle tâche doit-on choisir pour diminuer le plus fortement la durée minimum du projet? (Justifier la réponse.) Donner la nouvelle durée de la tâche choisie et la nouvelle durée minimum du projet.

On utilisera au choix une des méthodes de chemin critique à savoir la méthode *potentiels-tâches* ou *PERT*.

tâche	durée	tâches précédentes
A	6	–
B	4	–
C	5	–
D	3	A, B, C
E	2	A, C
F	4	A, C
G	7	A
H	4	F, G
I	6	J, F, G
J	4	D, E
K	8	D, E

EXERCICE 3 :

Résoudre le programme linéaire ci-contre en appliquant à la lettre la méthode du simplexe présentée en cours.

$$
\begin{array}{lrcl}
\textit{maximiser:} & z = & 3x_1 - x_2 & \\
\textit{sous:} & 2x_1 + x_2 & \geq & 2 \\
 & x_1 + 3x_2 & \leq & 3 \\
 & x_2 & \leq & 4 \\
 & x_1 \geq 0, & & x_2 \geq 0
\end{array}
$$

EXERCICE 4 :

Deux entreprises A et B se partagent un marché de consommateurs et chaque année A (resp. B) a le choix entre m (resp. n) campagnes publicitaires. Chacun ignore le choix de l'autre et on a une matrice $m \times n$ $C = (c_{ij})$ telle que :

$c_{ij} > 0$: part du marché de B repris par A si A choisi i et B choisi j ;
$c_{ij} < 0$: part du marché de A repris par B si A choisi i et B choisi j.

Au début de chaque année l'entreprise **A** décide de repartir son budget publicitaire et d'utiliser une fraction p_i (à déterminer) à la campagne i de manière que la fraction moyenne du marché reprise à **B** soit aussi grande que possible.

a) Formuler ce problème par un programme linéaire.
b) Donner son dual et interpréter les variables duales.
On va résoudre ce problème avec la matrice C ci-contre :
c) En utilisant les conditions des écarts complémentaires décider si utiliser uniquement sa campagne n°2 est une solution optimale pour A ou non.

6	-3	-6
2	2	0
4	5	-2

EXERCICE 5 :

Dans une usine, on veut installer des rails permettant à des wagonnets de se déplacer de n'importe quel site à n'importe quel autre. Un coût de construction est associé à chaque liaison directe. On souhaite minimiser le coût total de construction.

a) De quel problème classique de la théorie des graphes s'agit-il ?
b) Pour chaque site s on note $m(s)$ le minimum des coûts des liaisons ayant s pour extrémité. Peut-on dire que pour toute solution à coût minimum :

(i) On construira pour chaque s au moins une liaison d'extrémité s de coût $m(s)$?
(ii) On construira pour chaque s toutes les liaisons d'extrémité s de coût $m(s)$?
(iii) On ne construira pour chaque s que des liaisons d'extrémité s de coût $m(s)$?

c) Proposer un algorithme qui trouve une solution de coût minimum et qui maximise le nombre de liaisons directes entre le site "*livraison*" et les autres sites.

d) Résoudre l'exemple de la **Fig.1** où le coût associé à la construction de chaque liaison réalisable est indiqué.

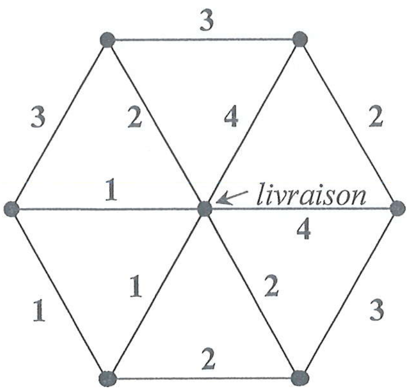


Fig.1.

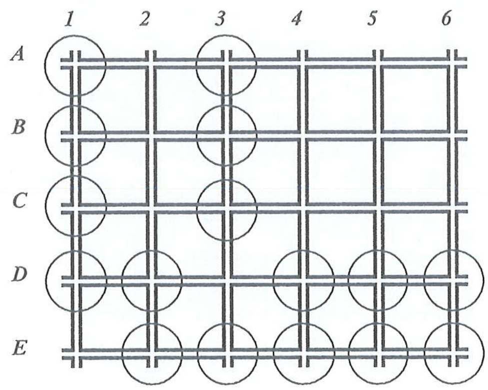


Fig.2.

EXERCICE 6 :

Dans un entrepôt, certaines zones sont à protéger spécialement contre l'incendie. On souhaite disposer des extincteurs sur le pourtour de l'entrepôt de façon à ce que chaque zone à protéger (entourée sur la **Fig. 2**) soit accessible pour un extincteur placée dans la même rangée (ligne ou colonne). On veut déterminer le nombre minimum d'extincteurs nécessaires et leur localisation.

En vous basant sur les modèles et les notions des graphes présentés en cours, choisir une modélisation la plus appropriée du problème posé. La solution de l'exemple présenté n'est pas demandée, il faut seulement bien justifier le choix de votre modèle pour le cas général.

Ensimag 1ère année   Le 12 mai 2011

Théorie des Langages 2

Durée : 3h.
Documents : tous documents autorisés.

Exercice 1 (6 points)

▷ **Question 1 (3 points)**
Soit $f : \mathbb{N} \rightarrow \mathbb{N}$ la fonction définie comme suit :

$$f(x) = \begin{cases} i & \text{si } x = 2 * i \\ i & \text{si } x = (2 * i) + 1 \end{cases}$$

Donner une machine de Turing qui réalise f. On utilisera un codage unaire pour l'entrée x. On pourra utiliser une machine à un ou plusieurs rubans. Expliquer la solution proposée.

▷ **Question 2 (3 points)**
On rappelle qu'un ensemble est récursivement énumérable si et seulement si il est le domaine d'une fonction calculable. Les ensembles suivants sont-ils récursivement énumérables ? Justifiez votre réponse.

1. $L_1 = \{a^{2^n} \mid n \in \mathbb{N}\}$
2. $L_2 = \{x \in \mathbb{N} \mid MTU(x, x)$ ne s'arrête pas $\}$

Exercice 2 (14 points)

Les grammaires étendues sont une notation plus compacte pour les grammaires hors-contexte. Elles permettent de décrire dans les parties droites de règles des parties optionnelles ou itérées.

Dans la définition classique des grammaires une partie droite de règle est une chaîne de l'ensemble $L_D = (V_T \cup V_N)^*$. Dans une grammaire étendue le langage des parties droites de règles est défini par les constructions suivantes :

(1)	$\epsilon \in L_D$	
(2)	si $X \in (V_T \cup V_N)$ alors $X \in L_D$	
(3)	si $w \in L_D$ alors $[w] \in L_D$	w : partie optionnelle
(4)	si $w \in L_D$ alors $\{w\} \in L_D$	w : partie itérée
(5)	si $w_1 \in L_D$ et $w_2 \in L_D$ alors $w_1 w_2 \in L_D$	

Les symboles [,], { et } sont des méta-symboles, i.e. des symboles appartenant au langage de description des règles.

On peut construire de manière systématique une grammaire " classique " à partir d'une grammaire étendue en appliquant (récursivement) les transformations suivantes :

1. Remplacer dans les parties droites de règles chaque sous-terme de la forme $[w]$ par un nouveau symbole non-terminal X et ajouter les deux règles :

   $X \rightarrow \epsilon \qquad X \rightarrow w$

2. Remplacer dans les parties droites de règles chaque sous-terme $\{w\}$ par un nouveau symbole Y et ajouter les deux règles :

   $Y \rightarrow \epsilon \qquad Y \rightarrow wY$

On notera $\mathcal{T}(G)$ la grammaire classique associée à la grammaire étendue G. On dira qu'une grammaire étendue G est LL(1) si et seulement si sa transformée $\mathcal{T}(G)$ est LL(1).

▷ **Question 1 (2 points).**

Soit la grammaire étendue G suivante :

(1)	SI	→	I {I}
(2)	I	→	affect
(3)	I	→	**if** exp **then** SI [**else** SI] **end** **;**
(4)	affect	→	**idf :=** exp **;**

Le non-terminal SI est l'axiome et les éléments du vocabulaire terminal sont notés en gras. Le non-terminal exp n'est pas défini ici. Donner la grammaire transformée $\mathcal{T}(G)$ associée à G.

▷ **Question 2 (2 points).**

Appliquer les calculs LL(1) sur la grammaire $\mathcal{T}(G)$ et proposer une grammaire équivalente LL(1), si besoin est. Soit G' la grammaire LL(1) obtenue.

▷ **Question 3 (3 points).**

Certaines normes de développement imposent des règles de programmation permettant de maîtriser la complexité du code. Une règle classique est de limiter la profondeur d'imbrication des conditionnelles. Par exemple la profondeur maximale d'imbrication des conditionnelles pour l'exemple suivant est 2 :

```
if e1 then
   if e2 then x:=3 ; end ;
else x:=1 ; end ;
if e3 then  x:=4 ; else x:=0 ; end ;
```

Ajouter un calcul d'attributs sur la grammaire G' de la question 2 permettant de calculer la profondeur maximale d'imbrication des conditionnelles dans une suite d'instructions (non-terminal SI). On expliquera précisément la signification des attributs utilisés.

▷ **Question 4 (4 points).**

A partir de la grammaire G' écrire un analyseur LL(1) qui reconnait une instruction (non-terminal I) et calcule le nombre maximal d'imbrication de conditionnelles. On supposera écrites les procédures d'analyse SI, exp et affect. On écrira toute autre procédure d'analyse nécessaire en fonction de la grammaire G' proposée.

▷ **Question 5 (3 points).**

On s'intéresse maintenant à étendre le principe de l'analyse LL(1) directement sur les grammaires étendues. Pour cela nous devons étendre les définitions $Eps(w)$, $Premier(w)$ et les conditions LL(1). On peut définir le prédicat $Eps(w)$ qui calcule si il existe une dérivation de m vers ϵ de la manière suivante :

$$\begin{array}{llcll}
(1) & Eps(\epsilon) & = & true & \\
(2) & Eps(xw) & = & false & x \in V_T \\
(3) & Eps(Aw) & = & (\vee_{A \to w_i} Eps(w_i)) \wedge Eps(w) & A \in V_N \\
(4) & Eps(\{r\}w) & = & Eps(w) & \\
(5) & Eps([r]w) & = & Eps(w) &
\end{array}$$

avec r et w des mots de L_D.

1. Étendre le calcul vu en cours des ensembles $Premier(w)$ pour les deux nouvelles constructions $[r]w$ et $\{r\}w$).
2. Pour les grammaires étendues l'ensemble des directeurs d'une règle se calcule de la même manière que pour les grammaires classiques. Par contre des conditions supplémentaires sont introduites par les règles contenant des occurrences de la forme $[w]$ ou $\{w\}$.

   Soit une règle de la forme $A \to w_1[w_2]w_3$. Donner des conditions, portant sur les chaînes w_2, w_3 et A, qui garantissent le caractère LL(1) de la grammaire étendue.
3. Même question pour une règle de la forme $A \to w_1\{w_2\}w_3$.

Examen de Traitement du Signal - Ensimag - 1A - Mai 2010

DOCUMENTS ET CALCULATRICES AUTORISÉES — SUJET DE 3 PAGES

1 QCM (4 points)

Les questions suivantes admettent aucune, une, ou plusieurs bonnes réponses.

Questions	Réponses
1. La transformée de Fourier d'un signal réel est :	☐ réelle ☐ paire ☐ hermitienne ☐ impaire
2. Soit $f(t)$ un signal $\mathbb{C}^\infty$ et $\delta(t-t_0)$ une impulsion de Dirac en t_0. On peut écrire :	☐ $f(t)\delta(t-t_0) = f(t_0)\delta(t-t_0)$ ☐ $f(t)\delta(t-t_0) = f(t-t_0)$ ☐ $f(t)\delta(t-t_0) = f(t+t_0)$ ☐ $f(t)\delta(t-t_0) = f(t_0)$
3. L'échantillonnage au sens de Shannon peut être réalisé sans perte d'information si :	☐ le signal à numériser est limité dans le temps ☐ le signal à numériser est à bande limitée ☐ la fréquence d'échantillonnage est supérieure à 1.5 fois la bande occupée par ce signal ☐ le signal à numériser est discontinu
4. Une opération de convolution représente un système :	☐ linéaire ☐ évolutif ☐ invariant
5. Un filtre linéaire invariant discret est totalement caractérisé par :	☐ sa fréquence de coupure ☐ sa réponse impulsionnelle ☐ sa fonction de transfert en z ☐ sa réponse fréquentielle
6. Un filtre réalisable est :	☐ à réponse impulsionnelle finie ☐ causal ☐ stable
7. La FFT d'un signal $\{x_k\}_{k\in\{0,\ldots,N-1\}}$:	☐ donne une représentation fréquentielle discrète ☐ donne une représentation fréquentielle continue
8. Un filtre à phase linéaire :	☐ peut être réalisé avec une structure transversale ☐ peut être réalisé avec une structure récursive

2 Fréquence pure, filtrage et quantification (6 points)

On considère le signal $s(t) = \cos[2\pi\nu_0 t]$, t réel.

1. A quelle fréquence minimale doit-on échantillonner $s(t)$ pour respecter le théorème d'échantillonnage de Shannon ?
2. $s(t)$ est maintenant observé sur l'intervalle $[0, T]$ seulement $(T > 0)$.
   (a) Comment peut-on modéliser cette restriction de l'intervalle d'observation ?
   (b) Quelle est l'impact de la troncature sur le contenu fréquentiel du signal ? Calculer la transformée de Fourier du signal tronqué.
   (c) A quelle fréquence doit-on échantillonner ce signal tronqué pour respecter les conditions du théorème d'échantillonnage ?
3. On note $i^2 = -1$. On filtre $s(t)$ par un filtre de gain complexe (fonction de transfert en Fourier)

$$H(\nu) = \begin{cases} \exp\left[-i\frac{\pi}{2}\right] & \text{pour} \quad \nu \in [-3\nu_0, 0) \\ \exp\left[+i\frac{\pi}{2}\right] & \text{pour} \quad \nu \in (0, +3\nu_0] \\ 0 & \text{sinon.} \end{cases}$$

   Calculer la sortie du filtre (dans le domaine temporel).
4. Quantification sur un bit : la sortie du filtre est transformée en un signal à 2 niveaux, les valeurs supérieures à 0 sont transformées en la valeur $+1$, les valeurs inférieures à 0 sont transformées en la valeur -1.
   (a) Quelle sont les fréquences contenues dans ce signal binaire ?
   (b) Que peut-on dire de la fonction de transfert (gain complexe en Fourier) de ce système de quantification ?

3 Étude d'un filtre numérique (6 points)

Soit un filtre linéaire causal d'entrée x_k et de sortie y_k dont le comportement est décrit par l'équation récurrente suivante, avec $r > 0$ et $\theta \in [0; 2\pi)$.

$$y_k = x_k + 2r\cos(\theta)y_{k-1} - r^2 y_{k-2},$$

1. Donner la fonction de transfert en z du filtre ainsi que son domaine de convergence.
2. Déterminer les pôles et les zéros de la fonction de transfert.
3. Comment peut-on vérifier la causalité du filtre dans le domaine z ?
4. Comment peut-on vérifier la stabilité du filtre dans le domaine z ? Étudier la stabilité du filtre en fonction de la valeur des paramètres r et θ.
5. Quelle est la réponse fréquentielle de ce filtre ? Par quel terme peut-on le qualifier (passe-haut, passe-bas, passe-bande, ...) ?

6. Ce filtre est-il à réponse impulsionnelle finie ?
7. On considère l'entrée $\{x_k\}_{k\in\mathbb{Z}}$ valant 1 pour $k \geq 0$ et 0 sinon. Quelle est la transformée en z, $X(z)$, de ce signal ? Préciser son domaine de convergence.
8. On applique ce x_k en entrée du filtre, calculer sa sortie.
9. On applique maintenant à l'entrée du filtre le même signal x_k retardé de d échantillons : $\{x_{k-d}\}_{k\in\mathbb{Z}}$ avec d entier positif. Quelle est la nouvelle sortie du filtre ?

4 Synthèse d'une sinusoïde (4 points)

On note $z_k = \exp[i2\pi\nu kT]$, $\nu \in \mathbb{R}$ fréquence fixée, $T > 0$ période d'échantillonnage fixée, $k \in \mathbb{Z}$.

1. Montrer que la sinusoïde de fréquence ν discrétisée au pas T et sa version déphasée de $\pi/2$ peuvent être générées par une équation récurente de la forme

$$\mathbf{x}_k = \mathbf{R}\mathbf{x}_{k-1} \text{ avec } \mathbf{x}_k^T = [\Re e\,[z_k]\,, \Im m\,[z_k]]$$

   $\mathbf{x}$ est un vecteur réel de taille 2 et $\mathbf{R}$ une matrice de rotation.

2. Dans le même esprit, on peut générer une sinusoïde discrète par

$$s_k = 2\cos(2\pi\nu T)\, s_{k-1} - s_{k-2},\ k \geq 0$$

   en choisissant de manière appropriée les conditions initiales s_0, s_1. Analyser cette équation aux différences à l'aide la transformée en z (stabilité, causalité).

SESSION de RATTRAPAGE JUIN 2011

ECONOMIE - 3 PAGES

ESPAGNOL – 4 PAGES

LOGIQUE POUR L'INFORMATIQUE - 2 PAGES

METHODES NUMERIQUES - 3 PAGES

PROBABILITES APPLIQUEES 1 – 1 PAGE

PROBABILITES APPLIQUEES 2 – 1 PAGE

THEORIE DES LANGUANGES 1 – 2 PAGES

THEORIE DES LANGUANGES 2 – 3 PAGES

ENSIMAG 1ère année – 2010/2011

Economie

Devoir de session 2 – Juin 2011

Durée : 2 heures
Documents autorisés.

Travail demandé :

Rédiger un **commentaire libre** du texte ci-joint.

Veillez à structurer votre travail (introduction, développement, conclusion) et à soigner la rédaction.

La difficile mesure du bien-être des populations

Thomas Baïetto - Le Monde, 25 mai 2011

L'Organisation de coopération et de développement économiques (OCDE) a mis en ligne, mardi 24 mai, un indicateur destiné à mesurer le bien-être des habitants de ses trente-quatre pays membres. Cet indicateur, qui regroupe dix-neuf variables réparties en onze critères, comme le logement ou l'éducation, est la première traduction concrète du rapport Stiglitz-Sen-Fitoussi de 2009.

Commandé par Nicolas Sarkozy à l'économiste américain Joseph Stiglitz, ce rapport avait pour objectif de réfléchir aux limites du baromètre privilégié qu'est le produit intérieur brut (PIB : somme des valeurs ajoutées produites en un an par l'économie d'un pays). *"Le PIB, ce n'est pas un indicateur suffisant"* pour mesurer l'évolution d'une société, explique Jean-Paul Fitoussi, professeur d'économie et coauteur du rapport. Un tremblement de terre et les reconstructions qu'il implique, une augmentation de la violence qui pousse les gens à s'équiper de systèmes d'alarmes, *"tout cela accroît le PIB mais sûrement pas le bien-être"*, constate-t-il. Même en matière de revenus, *"lorsque l'on vous dit que la croissance est de 3 %, eh bien, elle n'est de 3 % pour personne, elle est de 3 % en moyenne"*. Dans ses conclusions, le rapport de 2009 préconisait de modifier les systèmes de mesures statistiques mais également de réfléchir à de nouveaux indicateurs pour compléter le PIB. C'est dans cette optique que s'inscrit le projet présenté par l'OCDE.

L'INDICATEUR DE L'OCDE, "UN PREMIER ESSAI D'APPLICATION"

Partant des recommandations du rapport, l'OCDE établit une liste de onze critères : logement, revenu, emploi, communauté, éducation, environnement, gouvernance, santé, satisfaction, sécurité et l'équilibre vie professionnelle-vie familiale. Ces critères, notés de 1 à 10, comprennent plusieurs indicateurs, renvoyant à des mesures statistiques objectives, comme le taux d'emploi ou l'espérance de vie, et subjectives, comme la satisfaction, établie à partir de sondages. Le poids accordé à chaque critère dépend de l'internaute, qui peut lui donner une importance de 1 à 5 pour établir son indicateur personnel. Pour Jean-Claude Fitoussi, *"c'est un premier essai d'application ;* [...] *pour l'instant, on tâtonne"*.

Jean Gadrey, professeur d'économie et animateur du Forum pour de nouveaux indicateurs de richesse (FAIR), salue *"une bonne initiative"*, mais se montre réservé sur certains points. *"C'est un indicateur orienté vers le bien-être individuel, et pas du tout vers ce que l'on pourrait appeler la 'qualité d'une société'"*, regrette-t-il. Selon lui, cet indicateur manque de mesures de *"santé sociale"*, comme le pourcentage de personnes pauvres, de personnes couvertes par l'assurance maladie ou encore, le revenu des dix pour cent les plus riches divisés par le revenu des dix pour cent les plus pauvres, un indicateur d'inégalité utilisé par les Nations unies. Ces manques expliquent en partie les mauvais résultats de la France, et les bons résultats des Etats-Unis : *"Si on avait tenu compte d'indicateurs plus sociaux, la France serait remontée [dans le classement]"*, constate-t-il. Autre réserve, la faible place réservée à l'écologie, qui n'est représentée que par une seule des dix-neuf variables : la pollution de l'air en ville.

"UN THÈME QUI FAIT FLORÈS PARMI LES DÉCIDEURS POLITIQUES"

A court terme, les effets politiques de ce type d'indicateur restent limités. Mais *"les décideurs politiques s'aperçoivent qu'ils sont parfois en porte-à-faux avec la société quand ils se 'vantent' d'une croissance du PIB élevée alors qu'une très grande fraction de la société n'en bénéficie pas"*, estime Jean-Paul Fitoussi. *"Cela va influencer leur discours"*, nuance Jean Gadrey, qui compte davantage sur le débat public pour faire avancer cette question. Du côté des statisticiens, Stefan Lollivier, directeur de la direction des statistiques démographiques et sociales de l'Insee, explique que la réflexion sur les indicateurs de bien-être avance : *"c'est un thème qui fait florès parmi les statisticiens et les décideurs politiques"*.

L'Insee s'est associé à Eurostat, l'organisme statistique de la Commission européenne, et à l'OCDE pour mettre en place de nouvelles mesures. En effet, l'indicateur de l'OCDE se fonde sur des données déjà existantes, auxquelles s'ajoutent quelques enquêtes d'opinion. *"Nous sommes plus ambitieux côté européen, parce que nous voulons modifier le système de mesures. C'est-à-dire modifier les enquêtes européennes pour les adapter aux recommandations du rapport Stiglitz et pour que la statistique publique ajoute des questions qui n'existent pas forcément, notamment sur la sociabilité, les liens entre vie familiale et vie professionnelle"*, détaille M. Lollivier. L'Insee expérimente une collecte où différentes activités, comme travailler ou s'occuper des enfants, sont évaluées sur une échelle de 1 à 10, allant de l'agréable au désagréable. L'institut cherche également à identifier les populations fragiles, qui cumulent les indicateurs de mauvaise qualité de vie.

L'autre chantier majeur, c'est la pondération des différents critères et leur agrégation. Le rapport Stiglitz était très réticent à l'idée de proposer des indicateurs synthétiques. *"Nous n'aurons jamais de pondération satisfaisante, c'est clair et net, parce qu'il y a autant d'avis que de personnes"*, reconnaît Stefan Lollivier. Mais, *"parce que en termes de communication, c'est un message qui passe beaucoup mieux"*, un indicateur synthétique lui semble tout de même nécessaire. *"L'OCDE a fait un très bon travail, mais ce n'est vraiment qu'un début*, résume Jean-Paul Fitoussi. *Avant que nous aboutissions à quelque chose d'achevé, il faudra du temps."*

Ensimag 1A Espagnol Juin 2011 Session de Rattrapage Durée : 2 h

NOM, Prénom...

1. COMPLETA EL TEXTO CON LOS VERBOS EN PRESENTE DE INDICATIVO Y RESPONDE A LAS PREGUNTAS

El 22% de los universitarios (reconocer) ________________ que no (leer) ________________ nunca un libro

Los libros han acompañado al hombre a lo largo de la Historia, como un compañero fiel y un mudo testigo de guerras y conquistas. Son una fuente de conocimiento, (abrir) ________________ puertas y (ocupar) ________________ nuestras estanterías. **Pero, ¿los (leer) ________________ nosotros?**

Según el Ministerio de Cultura, el 22% de los universitarios no (leer) ________________ nunca. La cifra (variar) ________________ según la fuente. Por ejemplo, la Fundación BBVA (estimar) ________________ que **el 13% no leyó ningún libro durante el último año y el 18%, de uno a dos**. Mientras, el Barómetro de Hábitos de Lectura (decir) ________________ que es el sector joven -entre 10 y 13 años- el que más libros (devorar) ________________ "El hábito (reforzarse) ________________ con la edad, pero lo llamativo es que se (leer) ________________ más cuando ya se han finalizado los estudios que durante los estudios", explica Antonio María Ávila, director ejecutivo de la Federación.

La evolución en la última década es, no obstante, positiva. El índice de lectura se ha incrementando progresivamente **en los últimos 10 años,** pero nuestro país (encontrarse) ________________ muy alejado de la media europea. Los líderes son los países nórdicos, tras los que vienen Alemania, Francia y Reino Unido. España (situarse) ________________ a la cola, incluso los países del Este, recién llegados a la UE, (disponer) ________________ de un índice de lectura mayor que el nuestro". Algunas ciudades, como Madrid o Barcelona, sí (encontrarse) ________________ al nivel europeo, pero en comunidades como Galicia, Andalucía o Murcia, la situación es "mejorable".

¿Se fomenta la lectura? En Cuba, existe un **Festival Universitario del Libro y la Lectura** para seducir a los jóvenes. En España, también se han lanzado campañas para animar a lectura desde instituciones públicas y privadas. Como en el Metro, donde los carteles (proponer) ________________ a los jóvenes visitar las bibliotecas.

El perfil del lector español (poseer) ________________ nombre de mujer. Según la FGEE, quienes más (leer) ________________ son las mujeres de entre 30 y 45 años, con estudios universitarios y residentes en una zona urbana. Las motivaciones (modificarse) ________________ según la edad. Los jóvenes de 14 a 24 años leyeron su último libro por estudios; de 45 a 54 años, para mejorar su nivel cultural y, los mayores de 65, por consulta.

La Asociación de Editores de Diarios Españoles (alertar) ________________ de un alejamiento de los universitarios hacia la prensa, mientras (crecer) ________________ el número de lectores con estudios superiores. Los diarios deportivos son los preferidos por los jóvenes.

El Mundo 28/11/2007

Preguntas

1. ¿Estás de acuerdo con el texto? ¿Piensas que los jovenes están perdiendo el hábito de leer? ¿Sí? ¿No? ¿Por qué? (10 líneas)

....

....

....

....

....

....

....

....

....

....

....

2. ¿Cuál es o debería ser el lugar de la cultura general en la formación de un ingeniero? (10 líneas)

....

....

....

....

....

....

....

....

....

....

....

2. ELIGE <u>UNA</u> DE LAS SIGUIENTES INFORMACIONES Y HAZ UNA CRONICA (UTILIZANDO EL PASADO) EN LA QUE QUE RESPONDAS A LAS PREGUNTAS: ¿QUIÉN? ¿CÓMO? ¿CUÁNDO? ¿DÓNDE? ¿POR QUÉ?...

- Un hombre abre fuego en un centro comercial, matando a seis personas

....

....

....

....

- Tres detenidos por atracar y disparar a una mujer que conocieron por Internet

....

....

....

....

- Fingía ser un modelo adolescente para contactar con 380 menores

....

....

....

....

3. PON "SER" O "ESTAR" EN LOS ESPACIOS EN BLANCO Y CONJUGA EL VERBO EN EL TIEMPO PROPUESTO

1. María (passé simple)............................. en Paris para la promoción del nuevo perfume
2. La casa de Pedro (futuro).................................... muy grande
3. No te bebas el té (presente) .... muy caliente
4. La paella (imperfecto)................................. muy buena
5. Nosotros (passé composé) .... agotados todo el día a causa del tráfico
6. Creo que vosotros (condicional).......................... enfadados por esa razón tan estúpida
7. El examen (presente)............................... más fácil de lo que pensaba.
8. Los enfermos del hospital (passé simple)................................ bien atendidos

4. QUÉ DIRIAS A UNA PERSONA QUE... (UTILIZA EL CONDICIONAL CON VERBOS DIFERENTES PARA CADA FRASE)

- Tiene problemas para dormir (3 consejos con verbos diferentes)

- Quiere perder tres kilos de peso (3 consejos con verbos diferentes)

- Que no ha preparado el examen de matemáticas del viernes (3 consejos con verbos diferentes)

5. EXPRESION ESCRITA

El año académico se ha terminado. Escribe una carta a un amigo en la que cuentes algunas de las cosas que has aprendido a nivel humano y/o académico en la ENSIMAG (pasado), qué haces en este momento (presente) y cuáles son tus proyectos para el año que viene (futuro). *(25 lineas)*

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

EXAMEN DE "LOGIQUE POUR L'INFORMATIQUE"

Ensimag 1ère Année - juin 2011

DURÉE: 1 heure et demie.

SEULS DOCUMENTS AUTORISÉS: Poly "Une approche simple de la logique pour l'Informatique et la formalisation de l'inférence" et notes personnelles.

- S.V.P. **avant de commencer** lisez les 7 points ci-dessous.

- Cet examen comporte 6 questions. Toutes les questions sont indépendantes.
- L'ordre des questions n'a, en principe, aucun rapport avec leur difficulté.
- *S.V.P. rédigez vos réponses de façon similaire à celle des corrigés du poly.*
- S.V.P. **séparez** (par une ligne horizontale de la largeur de la feuille) et **numérotez clairement** les réponses aux différentes questions.
- **S.V.P.** respectez le nombre de lignes suggérées pour les réponses.
- Les réponses à l'aide des méthodes ***autres*** que celles demandées explicitement seront ***ignorées***.
- On tiendra compte dans la correction de la clarté, de la simplicité et de la concision des réponses ainsi que de la qualité de la présentation.

Question 1

Supposons une logique que nous appelerons $\mathcal{L}$ dans laquelle on peut écrire des formules du type :

$\{x \mid x \in \mathbb{N} \wedge x \neq 0 \wedge x \neq 1 \wedge x \neq 2 \ldots\}$

Un théorème de compacité obtenu en changeant dans la première ligne de la page 73 du poly LP par $\mathcal{L}$, en est-il toujours un dans $\mathcal{L}$?

Répondre par OUI ou par NON en donnant une justification d'au maximum 3 lignes.

Question 2

Considérer le raisonnement suivant :

Aucun C n'est B
Tous les B sont A

Quelques A sont non C

Après l'avoir formalisé en utilisant la traduction suggérée en Remarque 7 de page 26 du poly, pouvez-vous ?

a) Essayer de vérifier sa correction ***en utilisant la méthode des tableaux sémantiques*** (on vous demande de construire l'arbre correspondant jusqu'à une profondeur de 3).

b) S'il n'est pas correct, pouvez-vous utiliser l'information fournie par l'arbre pour ajouter une hypothse qui le rendrait correct ? Laquelle ?

c) Commentez. ***Maximum 3 lignes.***

Question 3

Pouvez-vous ***en utilisant la méthode de résolution*** dire si les ensembles de clauses ci-dessous sont insatisfaisables ? Comme d'habitude, a, b dénotent des constantes, x, y des variables, P un symbole de prédicat.

a) $\{P(x, b), \quad \neg P(a, x)\}$

b) $\{P(x, f(x)), \quad \neg P(g(y), y)\}$

Question 4

Pouvez-vous montrer l'insatisfaisabilité de l'ensemble de clauses S de l'exercice 39, page 81 du poly ***en utilisant la méthode de résolution*** avec *stratégie unitaire* (définie en exercice 40, page 81) ?

Question 5

Pensez-vous qu'il est possible de prouver l'insatisfaisabilité de l'ensemble de clauses de l'exemple 25, pages 68 et 69 du poly ***en utilisant la méthode de résolution*** avec *stratégie d'entrée* (voir exemple 29, page 81) ? ***Justifiez. Maximum 3 lignes.***

Question 6

On veut définir dans $\mathbb{N}$ *l'ensemble des nombres premiers* avec une fbf de la L1O.
Sachant que le domaine du discours est $D = \mathbb{N}$:

Complèter (i.e. remplacer les '...') l'expression ci-dessous (où x, y, z dénotent des variables) pour la transformer en la fbf de la L1O demandée et donner la signification des symboles de fonction ('a', 'f' et 'g') et du symbole de prédicat ('E').

$\neg E(x, f(a)) \ \wedge \ \forall y \forall z \, [E(x, g(y, z)) \Rightarrow ...]$

Fin-Examen

Examen de la session 2 — Juin 2011

Durée : 3h.

Les seuls documents autorisés sont les notes du cours et des travaux dirigés de méthodes numériques. Les calculatrices sont interdites.

La qualité de la rédaction et de la présentation sera prise en compte pour l'évaluation des copies.

Exercice I

$\mathcal{M}_n(\mathbb{C})$ désigne l'espace vectoriel des matrices (n, n) à coefficients dans $\mathbb{C}$.
$\|.\|$ est une norme sur $\mathcal{M}_n(\mathbb{C})$, supposée de plus *sous-multiplicative*, c'est-à-dire:

$$\forall A, B \in \mathcal{M}_n(\mathbb{C}), \qquad \|AB\| \leq \|A\| \cdot \|B\|$$

Soit $A \in \mathcal{M}_n(\mathbb{C})$ une matrice inversible.
On perturbe la matrice A en $(A + \Delta A)$. On désire savoir alors si la matrice perturbée $(A + \Delta A)$ est encore inversible et si oui, si $(A + \Delta A)^{-1}$ diffère beaucoup de A^{-1}.

1. Montrer que si $S \in \mathcal{M}_n(\mathbb{C})$ vérifie $\|S\| < 1$, alors la matrice $(I + S)$ est inversible et on a

$$(I + S)^{-1} = \sum_{k=0}^{+\infty} (-1)^k S^k = I - S + S^2 \ldots .$$

2. En déduire que si $\|\Delta A\| < \frac{1}{\|A^{-1}\|}$ alors $(A + \Delta A)$ est inversible.

3. Sous l'hypothèse de la question précédente $\left(\|\Delta A\| < \frac{1}{\|A^{-1}\|}\right)$ montrer que:

$$\frac{\|(A + \Delta A)^{-1} - A^{-1}\|}{\|(A + \Delta A)^{-1}\|} \leq \kappa(A) \cdot \frac{\|\Delta A\|}{\|A\|}$$

avec $\kappa(A) = \|A\| \cdot \|A^{-1}\|$.

4. Qu'en déduisez-vous quant au lien entre le conditionnement de A et le fait que $(A + \Delta A)^{-1}$ diffère peu ou beaucoup de A^{-1} ?

Exercice II

Soit le système linéaire:

$$Ax = b$$

avec

$$A = \begin{pmatrix} a & b & 0 \\ b & a & c \\ 0 & c & a \end{pmatrix}, \quad a, b, c \in \mathbb{R}, \quad a \neq 0.$$

1. Ecire la matrice M_J de l'itération de Jacobi et donner des conditions nécessaires et suffisantes sur a, b, c pour que la méthode converge.

2. Etudier la CV de la méthode de Gauss-Seidel suivant les valeurs des paramètres a, b, c.

Exercice III

Etant donné une matrice symétrique définie positive $A \in M_n(\mathbb{R})$ et $b \in \mathbb{R}^n$, on souhaite résoudre le système linéaire

$$A\,x = b, \quad x \in \mathbb{R}^n, \tag{1}$$

en utilisant une méthode itérative. Pour cela on décompose la matrice A en $A = M - N$ avec M inversible, et on considère la suite $(x_k)_{k \geq 0}$ dans $\mathbb{R}^n$ définie par

$$M\,x_{k+1} = N\,x_k + b, \quad x_0 \in \mathbb{R}^n. \tag{2}$$

Dans la suite de l'exercice on munit $\mathbb{R}^n$ de la norme $\|x\| = \sqrt{x^t A x}$, et $M_n(\mathbb{R})$ de la norme induite associée.

1. Montrer que $M^T + N$ est symétrique.

2. Etant donné $v \in \mathbb{R}^n$ et $w = M^{-1}Av$, montrer les égalités

$$\|M^{-1}Nv\|^2 = \|v - w\|^2 = \|v\|^2 - w^T(M^T + N)w.$$

3. Montrer que si $M^T + N$ est symétrique définie positive alors $\|M^{-1}N\| < 1$.

4. En déduire que si $M^T + N$ est symétrique définie positive alors la suite (x_k) converge vers x.

5. On décompose la matrice A en $A = L + D + U$, où D est la matrice diagonale telle que $d_{ii} = a_{ii}$ et

$$L = \begin{pmatrix} 0 & \cdots & \cdots & 0 \\ a_{21} & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ a_{n1} & \cdots & a_{n,n-1} & 0 \end{pmatrix}, \quad U = \begin{pmatrix} 0 & a_{12} & \cdots & a_{1n} \\ \vdots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & a_{n-1,n} \\ 0 & \cdots & \cdots & 0 \end{pmatrix}.$$

On considère la méthode SOR définie par

$$(D + \omega L)\,x_{k+1} = [(1-\omega)D - \omega U]\,x_k + \omega b, \quad x_0 \in \mathbb{R}^n \tag{3}$$

où $\omega > 0$ est le paramètre de relaxation. En utilisant la question **4**, montrer que si $0 < \omega < 2$ alors la méthode SOR converge.

Exercice IV

Soit $A \in M_n(\mathbb{R})$ une matrice symétrique définie positive. On cherche à résoudre le système linéaire

$$A^2 x = b, \tag{4}$$

avec $b, x \in \mathbb{R}^n$.

1. On considère une première méthode qui consiste à calculer A^2 puis résoudre (4) par la méthode de Cholesky. Donner un équivalent du coût lorsque $n \to +\infty$ du nombre d'opérations arithmétiques réalisées avec cet algorithme.

2. Montrer qu'on peut résoudre (4) sans calculer A^2, en utilisant la factorisation de Cholesky de A. Donner un équivalent du coût de cet algorithme lorsque $n \to +\infty$, et comparer son efficacité à celle de la méthode précédente lorsque n est grand.

Ensimag 1re année **2010-2011**

Probabilités Appliquées 1
Session 2

Durée 1h30 - Documents et calculatrices non autorisés.

1. On tire une variable aléatoire entière, N, à valeurs dans $\{1, 2, 3\}$, telle que

$$\mathrm{P}(N = 1) = \frac{1}{2}, \quad \mathrm{P}(N = 2) = \frac{1}{3}, \quad \mathrm{P}(N = 3) = \frac{1}{6},$$

   et on définit $X = U^{1/N}$, où U est une variable de loi uniforme sur $(0, 1)$ indépendante de N. Déterminer la fonction de répartition, puis la densité de la loi de X. Calculer l'espérance de X.

2. Soit N_1 et N_2 deux variables aléatoires indépendantes de loi géométrique de paramètre p. Déterminer la loi de la somme $N_1 + N_2$, ainsi que l'espérance et la variance de cette somme.

3. On dispose d'un dé équilibré à 6 faces, que l'on peut lancer autant de fois qu'on le souhaite. Décrire un algorithme permettant, à partir de ce dé, de simuler le résultat d'un dé équilibré à 15 faces. Combien de lancers sont nécessaires en moyenne pour obtenir le résultat ?

Ensimag 1re année **2010-2011**

Probabilités Appliquées 2
Session 2

Durée 1h30 - Documents et calculatrices non autorisés.

Exercice I - On considère un couple (X, Y) de variables aléatoires de densité conjointe

$$\forall (x, y) \in \mathbb{R}^2, \quad f(x, y) = ce^{-x}\mathbb{1}_D(x, y),$$

où c est une constante positive et $D = \{(x, y) \in \mathbb{R}^2 \; ; \; 0 < y < x\}$.

1) Déterminer la valeur de c, la loi de la variable Y, la densité de la loi conditionnelle de X sachant $Y = y$ pour tout $y \in \mathbb{R}_+$, et l'espérance conditionnelle $E[X|Y]$.
2) Écrire un algorithme de simulation d'un couple de densité f.
3) Calculer $E[XY]$.
4) On pose $Z = Y/X$. Démontrer que Z est de loi uniforme sur $(0, 1)$.

Exercice II - Proposer un algorithme de simulation d'un couple gaussien (X, Y) de moyenne nulle et de matrice de covariance

$$K = \begin{pmatrix} 7 & -\sqrt{3} \\ -\sqrt{3} & 5 \end{pmatrix}.$$

Prouver la validité de l'algorithme de simulation proposé. On suppose disposer d'un générateur aléatoire de loi $\mathrm{N}(0, 1)$.

Grenoble InP - Ensimag 1ère année Juin 2011

Théorie des langages 1 – Session 2

Durée : 2h.
Documents : tous documents autorisés.

Nb : le barème est donné à titre indicatif ; la rigueur des preuves et la clarté de la rédaction seront prises en compte dans la notation.

Exercice 1 - Automates (5 points)

On considère L, l'ensemble des mots sur le vocabulaire $\{a, b\}$ contenant au moins deux occurrences de la sous-chaîne aa séparées par un b.

Par exemple :

- $aabaa$, $babaabbaa$ et $bbaabbabaab$ sont dans L ;
- aa, $baaaa$, $baab$ et $baabab$ ne sont pas dans L.

▷ **Question 1** *(2 points) Donner une expression régulière représentant L en construisant un automate reconnaissant L puis en résolvant le système d'équations associé.*

▷ **Question 2** *(3 points) Donner un automate déterministe minimal qui reconnaît L en détaillant les étapes de construction.*

Exercice 2 - Langages réguliers (10 points)

Etant donné un vocabulaire V, et un langage $L \subseteq V^*$, on définit le *langage des préfixes de L* par :

$$\mathrm{Pre}(L) = \{u \in V^* \mid \exists v \in V^* : uv \in L\}.$$

Autrement dit, le langage des préfixes de L est constitué des mots qui sont des préfixes de mots de L.

▷ **Question 1** *(3 points) On pose $V = \{a, b\}$. A quoi correspond* $\mathrm{Pre}(L)$ *pour les langages L suivants :*

- $L = \{$*mots de V^* contenant deux 'a' consécutifs*$\}$
- $L = \{xy \mid x \in \{a\}^*, y \in V^*\}$
- $L = \{$*mots qui commencent par 'ab'*$\}$

▷ **Question 2** *(3 points) Soit $A = (Q, V, \delta, I, F)$ un automate* quelconque *reconnaissant un langage L. Construire à partir de A un automate qui reconnaît* $\mathrm{Pre}(L)$.

▷ **Question 3** *(1 point) Appliquer la construction de la question précédente à l'automate suivant : $A = (Q, \{a, b, c\}, \{q_0\}, \delta, \{q_6\})$, où $Q = \{q_0, \ldots, q_4\}$, et la relation de transition δ est définie dans le tableau ci-dessous :*

δ	a	b
q_0	q_3	q_1
q_1	q_2	q_4
q_2	q_2	q_2
q_3	q_3	q_3
q_4	q_3	q_2

Ensimag 1ère année — Juillet 2011

Théorie des Langages 2

Durée : 2h.
Documents : tous documents autorisés.

Exercice 1 (6 points)

▷ **Question 1 (1 point)**
Rappeler la définition des ensembles récursifs et celle des ensembles récursivement énumérables.

▷ **Question 2 (2 points)**
Soit $L_1 = \{a^{2n} : n \geq 0\}$. Montrer que L_1 est récursif.

▷ **Question 3 (3 points)**
Soit $L_2 = \{x \in \mathbb{N} : MTU(x, x) \text{ s'arrête}\}$. Montrer que L_2 n'est pas récursif. L_2 est-il récursivement énumérable ?

Exercice 2 (14 points)

Les grammaires étendues sont une notation plus compacte pour les grammaires hors-contexte. Elles permettent de décrire dans les parties droites de règles des parties optionnelles ou itérées.

Dans la définition classique des grammaires une partie droite de règle est une chaîne de l'ensemble $L_D = (V_T \cup V_N)^*$. Dans une grammaire étendue le langage des parties droites de règles est défini par les constructions suivantes :

(1)	$\epsilon \in L_D$	
(2)	si $X \in (V_T \cup V_N)$ alors $X \in L_D$	
(3)	si $w \in L_D$ alors $[w] \in L_D$	w : partie optionnelle
(4)	si $w \in L_D$ alors $\{w\} \in L_D$	w : partie itérée
(5)	si $w_1 \in L_D$ et $w_2 \in L_D$ alors $w_1 w_2 \in L_D$	

Les symboles [,], { et } sont des méta-symboles, i.e. des symboles appartenant au langage de description des règles.

On peut construire de manière systématique une grammaire " classique " à partir d'une grammaire étendue en appliquant (récursivement) les transformations suivantes :

1. Remplacer dans les parties droites de règles chaque sous-terme de la forme $[w]$ par un nouveau symbole non-terminal X et ajouter les deux règles :

   $X \rightarrow \epsilon \qquad X \rightarrow w$

2. Remplacer dans les parties droites de règles chaque sous-terme $\{w\}$ par un nouveau symbole Y et ajouter les deux règles :

   $Y \rightarrow \epsilon \qquad Y \rightarrow wY$

On notera $\mathcal{T}(G)$ la grammaire classique associée à la grammaire étendue G. On dira qu'une grammaire étendue G est LL(1) si et seulement si sa transformée $\mathcal{T}(G)$ est LL(1).

▷ **Question 1 (3 points).**
Soit la grammaire étendue G suivante :

1	suite-inst	$\rightarrow$	inst SI
1	SI	$\rightarrow$	suite-inst
2	SI	$\rightarrow$	ϵ
3	inst	$\rightarrow$	**idf :=** exp **case ;**
4	inst	$\rightarrow$	**case** exp **of** lch **end case ;**
5	lch	$\rightarrow$	choix {choix} [autre]
6	autre	$\rightarrow$	**when others** $\Rightarrow$ suite-inst
7	choix	$\rightarrow$	**when const** $\Rightarrow$ suite-inst

Le non-terminal suite-inst est l'axiome et les éléments du vocabulaire terminal sont notés en gras. $V_T = \{\mathbf{idf}, :=, \mathbf{case}, \mathbf{of}, \mathbf{end}, ;, \mathbf{when}, \mathbf{const}, \mathbf{others}\}$. Le non-terminal exp n'est pas défini ici.

Donner la grammaire transformée $\mathcal{T}(G)$ associée à G. Montrer que la grammaire $\mathcal{T}(G)$ n'est pas LL(1).

▷ **Question 2 (3 points).**
Proposer une grammaire LL(1) définissant le même langage que $\mathcal{T}(G)$. Justifier votre réponse.

▷ **Question 3 (4 points).**
Pour des questions d'optimisation du code généré par un compilateur, nous voulons calculer pour chaque instruction **case** le nombre de choix (sans inclure le cas others) et la présence ou l'absence d'un **when others**. Ecrire la procédure d'analyse LL(1) qui reconnait une liste de choix (non-terminal lch) et calcule les informations précédentes. On supposera écrites les procédures d'analyse suite-inst et choix, ainsi que la procédure `lire-mot` qui a pour effet de lire un élément du vocabulaire terminal. On utilisera la variable globale `mot` qui contiendra le dernier mot lu. On écrira toute autre procédure d'analyse nécessaire en fonction de la grammaire proposée en question 2.

▷ **Question 4 (2 points).**
On s'intéresse maintenant à déterminer le caractère LL(1) d'une grammaire directement sur les grammaires étendues. On rappelle qu'on calcule l'ensemble des directeurs d'une règle par la formule suivante :

$$Dir(A \rightarrow w) = Premier(w) \bigcup_{w \Rightarrow^* \epsilon} Suivant(A)$$

1. Expliquer comment calculer $w \Rightarrow^* \epsilon$ pour w de la forme $w_1[w_2]w_3$ avec w_1, w_2 et w_3 des chaînes de L_D.
2. Expliquer pourquoi une grammaire étendue dont une partie droite de règle contient une occurrence de la forme $[w]$ (ou $\{w\}$), avec $w \Rightarrow^* \epsilon$, ne peut pas être LL(1).

▷ **Question 5 (2 points).**

Pour les grammaires étendues les conditions sur les ensembles de directeurs sont les mêmes que pour les grammaires classiques. Par contre des conditions internes sont introduites pour les règles contenant en partie droite des occurrences de la forme $[w]$ ou $\{w\}$.

Soit une règle de la forme $A \rightarrow w_1[w_2]w_3$. En se basant sur les transformations données au début de ce sujet, donner des conditions portant sur les chaînes w_2, w_3 et A, garantissant le caractère LL(1) de la grammaire étendue. Même question pour une règle de la forme $A \rightarrow w_1\{w_2\}w_3$.