

Informatique

Examen — 1^{re} session

Exercice 1

Supposons que le tableau suivant est le dernier tableau d'un simplexe primal correspondant à la résolution d'un problème de maximisation. Fixer les valeurs de a, b, c, d pour que les expressions suivantes soient vraies :

		x_5	x_3
	45	a	11/2
x_2	3	-1/2	1/2
x_4	b	c	-1/2
x_1	5	d	1/2

1. La solution courante est optimale et il existe des solutions optimales multiples
2. La solution de base courante n'est pas une solution de base réalisable
3. La solution courante est dégénérée
4. La solution de base courante est réalisable mais le problème est non borné

Exercice 2

Soit le problème suivant :

$$\begin{aligned}
 \min z &= x_1 - x_2 \\
 \text{s.c.} \quad x_1 + x_2 &\geq 2 \\
 x_1 - 2x_2 &\geq 1 \\
 x_1, x_2 &\geq 0
 \end{aligned}$$

1. Appliquer la phase 1 de l'algorithme dual simplexe afin d'obtenir une base duale admissible pour ce problème.
2. Appliquer la phase 2 de l'algorithme dual simplexe pour achever la résolution du problème. On pourra éventuellement partir de la base duale admissible $\{2, 3\}$ dont le tableau simplexe complet est donné ci-dessous.

z	$\frac{1}{2}$	$-\frac{1}{2}$	0	0	$-\frac{1}{2}$
x_3	$-\frac{5}{2}$	$-\frac{3}{2}$	0	1	$\frac{1}{2}$
x_2	$-\frac{1}{2}$	$-\frac{1}{2}$	1	0	$\frac{1}{2}$

Exercice 3

Soit le problème suivant :

$$\begin{aligned}
 \max z &= x_1 + x_2 + x_3 \\
 \text{s.c.} \quad x_1 + x_2 &\leq 1 & (1) \\
 x_1 + x_3 &\leq 1 & (2) \\
 x_2 + x_3 &\leq 1 & (3) \\
 x_1, x_2, x_3 &= (0, 1) & (4)
 \end{aligned}$$

1. Ce problème est caractéristique dans la littérature de l'optimisation combinatoire ; comment s'appelle-t-il ?
2. Donner une solution optimale
3. Donner une borne duale obtenue par relaxation linéaire (calculs à l'aide de l'algorithme simplexe primal).
4. En quoi l'ajout de la contrainte $x_1 + x_2 + x_3 \leq 1$ à ce problème est-il intéressant ? Expliquer et donner des arguments quantitatifs.

Exercice 4

Soit le problème suivant :

$$\begin{array}{ll} \max z & = 14x_1 + 17x_2 + 15x_3 + 18x_4 + 7x_5 \\ \text{s.c.} & 15x_1 + 23x_2 + 12x_3 + 24x_4 + 7x_5 \leq 53 \\ & x_1, x_2, x_3, x_4, x_5 = (0, 1) \end{array}$$

1. Calculer une borne primale et donner la solution obtenue.
2. Calculer une borne duale et donner la solution obtenue.
3. Compte-tenu du résultat précédent, peut-on espérer améliorer la solution issue de la borne primale ? Justifier.
4. Calculer et donner une solution optimale en utilisant un algorithme de branch and bound

Détailler l'ensemble des calculs réalisés.

Exercice 5

Une raffinerie mélange 5 types d'essence brute pour obtenir deux qualités de carburant : normal et super. Le nombre de barils disponibles par jour, le taux de performance ainsi que le prix par baril est donné pour chaque type d'essence brute dans le tableau suivant :

Type d'essence	Performance	Barils disponibles	Prix / baril (€)
1	70	2000	0.80
2	80	4000	0.90
3	85	4000	0.95
4	90	5000	1.15
5	99	5000	2.00

Le carburant normal doit avoir un taux de performance d'au moins 85 et le super d'au moins 95. Des contrats obligent la raffinerie à produire au moins 8000 barils de super par jour. Mais ils peuvent vendre toute leur production aux prix suivants : 2.85€ par baril de carburant normal et 3.75€ par baril de super. Supposons que le taux de performance est proportionnel au mélange (p.ex. un mélange $\frac{1}{2} - \frac{1}{2}$ d'essences brutes de type 1 et 2 donne un taux de performance de 75). Donner une formulation sous forme de programme linéaire qui maximise le profit de la raffinerie.

Semestre 2

Examen

Conception de Logiciels extensibles

Exercice 1

1. Que permet la réflexion dans un langage de programmation ? en Java ?
2. En quoi est elle utile pour concevoir des logiciels extensibles ?
3. Quels sont ses inconvénients en Java ?

Exercice 2

On souhaite écrire une méthode `testMethod` ayant la signature suivante :

```
public Object testMethod(Object target, Method m, Object source) throws  
Exception
```

Elle retournera le résultat de l'exécution de la méthode `m` sur l'objet `target`. Si `m` possède des paramètres, on cherchera à obtenir chacun d'entre eux en utilisant des méthodes sans paramètres applicables sur l'objet `source`. Par exemple, si `m` nécessite un paramètre de type `String`, un appel de la méthode `toString` sur `source` pourra convenir.

1. Donner en trois lignes maximum, une version résumée de l'algorithme pour cette méthode
2. Écrire le code Java de la méthode et des méthodes auxiliaires

Exercice 3

1. Donner le diagramme de classes du patron de conception Proxy
2. Que permet la classe `Proxy` en Java ?
3. En quoi son utilisation est elle différente de celle du patron de conception Proxy ?

Exercice 4

On souhaite modifier les chaînes de caractères en paramètre et en retour d'appel de méthodes pour certains objets. Les modifications seront déléguées à des classes implémentant une interface `IStringModifier` contenant deux méthodes `String modifyParam(String)` et `String modifyReturn(String)`.

On souhaite encapsuler des objets quelconques dans un proxy devant leur ajouter une interface permettant de leur affecter un `IStringModifier` chargé de réaliser les modifications attendues.

1. Écrire l'interface d'affectation de `IStringModifier`
2. Écrire une méthode création du proxy à partir d'un objet
3. Écrire le code de gestion du proxy (le handler)
4. En utilisant la méthode de création du proxy, écrire le code affectant un `IStringModifier` de votre choix à un objet quelconque

Master1 Informatique
Examen : Compilation
Avril 2013-1^{ère} session – Durée 1 H30
Tous documents autorisés

Question de cours

(nous reprenons dans ces questions les notations du cours)

Etant donnée G0 la grammaire des grammaires suivante :

Règle 1 $S \rightarrow [N. \text{'->'.E. \text{'.'}]. \text{'.'}];$

Règle 2 $N \rightarrow \text{'IDNTER'}$,

Règle 3 $E \rightarrow T.[\text{'+'}.T],$

Règle 4 $T \rightarrow F.[\text{'.'}.F],$

Règle 5 $F \rightarrow \text{'IDNTER' } + \text{'ELTER' } + \text{'(}.E. \text{'')' } + \text{'[}.E. \text{']'} + \text{'(/}.E. \text{'/')};$

1. Positionner les actions sémantiques sur chacune des règles et donner le rôle de chaque action.
2. Quels sont les rôles respectifs de : Action G0 , Action GPL?
3. Donner les Pcodes associés aux mnémoniques suivantes : SUPE, RD, AFF

Problème

Etant donnée la grammaire suivante :

$G \rightarrow S\$$

$S \rightarrow aACb$

$A \rightarrow CA$

$C \rightarrow Cd$

$A \rightarrow a$

$C \rightarrow d$

1. Décrire la table « Shift-Reduce » correspondante
2. Déterminer les configurations (états) de la pile et les poignées lorsqu'on fait l'analyse de la chaîne : adaddb\$
3. Cette grammaire est-elle LL(1) ? justifier.

Semestre 2

Session 2

Master 1 Informatique
Examen Compilation
2^{ème} session juin 2013 – Durée 1h30 heures
Tous documents autorisés

Question de cours

Etant donnée G0 la grammaire des grammaires suivante :

- Règle 1 S -> [N. '>'E. ';'. ' ;',
- Règle 2 N -> 'IDNTER',
- Règle 3 E -> T.['+'T],
- Règle 4 T -> F.[':'F],
- Règle 5 F -> 'IDNTER ' + 'ELTER ' + ' (.E. ') ' + ' ['.E. '] ' + ' (/'.E. '/)';;

Avec les notations habituelles :

[X] ⇔ X.X....X n fois X avec n ≥ 0

(/X/) ⇔ X ou vide

+ ⇔ OU

. ⇔ Concaténation

'X' ⇔ X terminal

X ⇔ X non terminal

- Vérifier que G0 appartient bien à G0

Exercice

Etant donné le programme SOMME suivant :

```
Programme SOMME ;
Var S,I : integer;
    T : array(1..5) of integer;
Debut
    I:= 1;
    S:=0;
    While I <= 5 do
    Debut
        Read(T(I));
        S:= S + T [I];
        I:= I+1;
    Fin
    Write(S)
Fin.
```

1. Donner le P-Code associé à ce programme, pour une machine ayant un seul accumulateur (reprendre les notations du cours).
2. Donner l'état de la pile d'exécution après chaque instruction du programme.