

Algorithmique et programmation

Master 2 CCI

TD 2

12 octobre 2007

1 Rappels

1.1 Expressions booléennes

Un booléen est un objet qui peut prendre deux valeurs : VRAI ou FAUX. Il existe trois opérations de base dans le calcul d'expressions booléennes : ET, OU et NON (respectivement `&&`, `||` et `!`, en C).

		b	
		VRAI	FAUX
a	VRAI	VRAI	FAUX
	FAUX	FAUX	FAUX

(a) `a && b`

		b	
		VRAI	FAUX
a	VRAI	VRAI	VRAI
	FAUX	VRAI	FAUX

(b) `a || b`

a	!a
VRAI	FAUX
FAUX	VRAI

(c) `!a`

En C, il n'existe pas de type booléen. À la place, on utilise des entiers (généralement le type `char`, qui ne prend qu'un octet). 0 correspond alors FAUX et toute autre valeur à VRAI. En général, on définit VRAI et FAUX par :

```
#define VRAI (0==0)
#define FAUX (!VRAI)
```

Exemple :

```
if (moyenne < 10)
    printf("Recalé\n");
if ((moyenne >= 10) && (moyenne < 12))
    printf("Passable\n");
if ((moyenne >= 12) && (moyenne < 14))
    printf("Assez bien\n");
if ((moyenne >= 14) && (moyenne < 16))
    printf("Bien\n");
if (moyenne >= 16)
    printf("Très bien\n");
```

1.2 Blocs d'instructions

Il est possible de regrouper des instructions en bloc. En C, les instructions d'un bloc sont délimitées par `{` et `}`.

Exemple :

```
if (1 < 0)
    printf ("Ce message n'est pas affiché\n");
    printf ("Celui-là, l'est\n");
if (1 < 0)
{
    printf ("Ce message n'est pas affiché\n");
    printf ("Celui là, non plus\n");
}
```

Simplifier les expressions booléennes suivantes :

1. $(\text{!}faux) \parallel faux$
2. $vrai \ \&\& \ ((\text{!}vrai) \parallel vrai)$
3. $(\text{!}((vrai \ \&\& \ faux) \parallel vrai)) \parallel faux$

2.2 XOR

L'opération booléenne XOR (OU EXCLUSIF) est définie de la manière suivante :

$$a \ XOR \ b \ = \ (a \ \&\& \ \text{!}b) \parallel (\text{!}a \ \&\& \ b)$$

Écrire la table de vérité de XOR.

2.3 Maximum

Écrire un programme C qui calcule le maximum de trois entiers a, b et c.

2.4 Signe

Écrire un programme C qui affiche le signe du produit de deux entiers a et b, sans faire la multiplication !

Compléter ce programme pour qu'il affiche le signe de l'addition des deux mêmes entiers, sans faire la somme !

Remarque : utilisez la fonction `double fabs(double x)` qui donne la valeur absolue de la variable x

2.5 Parité d'un nombre

Écrire un programme C qui teste si un nombre est pair.

2.6 Année bissextile

Si l'année A n'est pas divisible par 4, alors, elle n'est pas bissextile.

Si A est divisible par 4, l'année est bissextile sauf si A est divisible par 100 et pas par 400.

Écrire un programme qui détermine si une année est bissextile ou non.

2.7 Polynômes du 2ème degré

Écrire un programme qui affiche les solutions éventuelles d'une équation de la forme $ax^2 + bx + c = 0$