PF1 – TD5 : RéVisions

4 décembre 2013

Exercice 1. [La base]

1. Comment s'écrit $(A167B12)_{16}$ en base 2 ? En base 8 ?

   Réponse. *On écrit chaque chiffre hexadécimal sur* 4 *bits en binaire. D'où*

   $$(A167B12)_{16} = (1010\ 0001\ 0110\ 0111\ 1011\ 0001\ 0010)_2$$

   Pour l'avoir en base 8*, on regroupe les bits par paquets de* 3 **en commençant par la droite** *et en ajoutant les* 0 *nécessaires à gauche. On écrit ensuite le chiffre associé à chaque paquet :*

   $$(A167B12)_{16} = (001\ 010\ 000\ 101\ 100\ 111\ 101\ 100\ 010\ 010)_2 = (1205475422)_8$$

2. Comment s'écrit $(FAC)_{17}$ en base 10 ?

   Réponse. *Avec Hörner, on souffre moins :* $(FAC)_{17} = (15 \times 17 + 10) \times 17 + 12 = 4517$

3. Comment s'écrit $(2308)_{10}$ en base 14 ?

   Réponse. *On applique l'algorithme des divisions euclidiennes successives pour trouver chaque chiffre :*

   $$2308 = 14 \times 164 + 12$$
   $$164 = 14 \times 11 + 10$$
   $$11 = 14 \times 0 + 11$$

   Donc le dernier chiffre en base 14 *est le chiffre qui correspond à* 12 *c'est-à-dire* C*. Puis* A*, puis* B*. D'où* $2308 = (BAC)_{14}$*. Attention, c'est différent de* $(111012)_{14} = 579000$ *!*

4. Comment s'écrit $(1001110110111)_2 + (101101110111)_2$ en hexadécimal ?

   Réponse. *Deux approches : soit vous faites l'addition en binaire et convertissez en hexadécimal. Soit vous convertissez en hexadécimal puis faites l'addition en binaire. Deuxième solution (les retenues sont entourées) :*

   $$\begin{array}{ccccc} & & 1 & & \\ & 1 & 3 & B & 7 \\ + & & B & 7 & 7 \\ \hline & 1 & F & 2 & E \end{array}$$

Exercice 2. [Représentant]

1. Quelle est la représentation de -45703 si on le stocke dans un `int` Java ? Que se passe-t-il si on le stocke dans un `short` ?

   Réponse. *On rappelle qu'un* `int` *représente les nombres sur* 32 *bits en complément à deux. Sur* 32 *bits,*

   $$45703 = (0000B287)_{16} = (0000\ 0000\ 0000\ 0000\ 1011\ 0010\ 1000\ 0111)_2$$

   Pour trouver l'opposé, on inverse les bits et on ajoute 1 *:*

   $$-45703 = 1111\ 1111\ 1111\ 1111\ 0100\ 1101\ 0111\ 1001$$

   Les shorts utilisent 16 *bits. On voit que* -45073 *ne peut pas être représenté par un short, donc on aura un overflow.*

2. Que vaut `z` après avoir exécuté le code suivant :

```
byte x = 0xA7;
byte y = −24;
byte z = x && y;
```

Réponse. *En machine x sera représenté par* 1010 0111 *et y par* 1110 1000 *(car* $24 = (0001\ 1000)_2$*). z est un "ET" bit à bit de x et de y d'où z est représenté par* 1010 0000*. Pour trouver sa valeur décimal : on voit que z est un nombre négatif, donc on commence par calculer l'écriture binaire de son opposé en enlevant* 1 *et en inversant les bits. En enlevant* 1 *:* 1001 1111*. En inversant les bits :* $-z = (0110\ 0000)_2$ *d'où* $z = -96$

3. Quelle est la représentation `float` du nombre 15.625 ?

Réponse. *On commence par écrire* 15.625 *en binaire.* $15 = (1111)_2$ *et* $0.625 = (0.101)_2$ *d'où* $15.625 = (1111.101)_2$*. Ensuite, on le normalise (on le met sous la forme* $2^k \times 1.a_1a_2\ldots$*) :* $1111.101 = 1.111101 \times 2^3$*. L'exposant stocké dans le* `float` *sera donc* $e = 127 - 3 = 124 = (00111110)_2$*. La mantisse stockée sera ce qui est derrière la virgule sur* 23 *bits soit* 11110100000000000000000*. D'où, en machine :*

$$0\ 00111110\ 11110100000000000000000$$

Exercice 3. [Petits poids]

1. Combien pèse le fichier obtenu en scannant un document de 4 pouces de large, 6 de long, couleurs vraies (RGB), avec une résolution de 1200dpi ? Et en noir et blanc ?

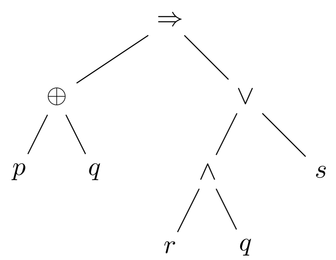
Réponse. *L'image aura* 4×1200 *pixels de large, et* 6×1200 *de long. Chaque pixel est représenté par* 24 *bits soit* 3 *octets en RGB donc un poids* $4 \times 1200 \times 6 \times 1200 \times 3 = 103680000$ *octets. En noir et blanc, chaque pixel est représenté par* 1 *bits donc un poids de* $4 \times 1200 \times 6 \times 1200/8 = 4320000$ *octets.*

2. Combien pèse un enregistrement stereo de 23 minutes, échantillonné à 44kHz, 16 bits ?

Réponse. *Pour chaque canal, chaque seconde de l'enregistrement est représentée par* 44 000 *valeurs codées sur* 16 *bits soit* 2 *octets d'où un poids total de* $23 \times 60 \times 44000 \times 2 \times 2 = 242880000$ *octets.*

Exercice 4. [Logique]

1. Donner l'arbre syntaxique, l'écriture préfixée et l'écriture postfixée de la formule $(p \oplus q) \Rightarrow ((r \wedge q) \vee s)$.



Réponse.
En préfixée : $\Rightarrow \oplus pq \vee \wedge rqs$
En postfixée : $pq \oplus rq \wedge s \vee \Rightarrow$

2. Donner une table de Karnaugh pour cette formule. En déduire une DNF simplifiée.

Réponse. *On donne directement la table de Karnaugh, mais si vous n'êtes pas à l'aise avec cela, écrivez d'abord la table de vérité :*

$rs \backslash pq$	00	01	11	10
00	1	0	1	0
01	1	1	1	1
11	1	1	1	1
10	1	1	1	0

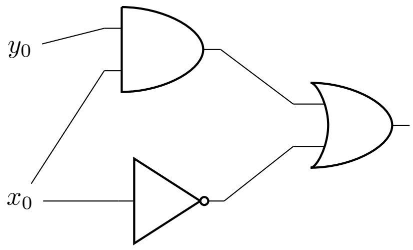
On recouvre les 1 *avec des rectangles de côté une puissance de* 2 *:*

– *la première colonne, qui donne le terme* $\neg p \wedge \neg q$
– *la troisième colonne qui donne le terme* $p \wedge q$
– *la deuxième et la troisième ligne qui donne le terme* s
– *le carré* 2×2 *en bas à droite qui donne le terme* $\neg p \wedge r$
Au final, on obtient : $(\neg p \wedge \neg q) \vee (p \wedge q) \vee s \vee (\neg p \wedge r)$

Exercice 5. [Circuits courts]

1. Donner un circuit $CMP_0(x_0, y_0)$ dont la sortie vaut 1 si $x_0 \leq y_0$ et 0 sinon.

   Réponse. *Soit* $y_0 = 1$ *et dans ce cas* $x_0 \leq y_0$ *quelque soit la valeur de* x_0*. Soit* $y_0 = 0$ *et dans ce cas* $x_0 \leq y_0$ *seulement si* $x_0 = 0$*. Donc on trouve* $CMP_0(x_0, y_0) = (y_0 \vee (\neg x_0 \wedge \neg y_0)) = (y_0 \vee \neg(x_0 \vee y_0))$*. On peut raisonner aussi ainsi : soit* $x_0 = 0$ *et dans ce cas* $x_0 \leq y_0$*. Soit* $x_0 = 1$ *et dans ce cas* $x_0 \leq y_0$ *seulement si* $y_0 = 1$*. D'où* $CMP_0(x_0, y_0) = \neg x_0 \vee (x_0 \wedge y_0)$ *(les deux sont justes !).*



2. À partir d'un circuit $CMP_n(x_0, \ldots, x_n, y_0, \ldots, y_n)$ qui vaut 1 si $x \leq y$ et 0 sinon (où $x = (x_n \ldots x_0)_2$ et $y = (y_n \ldots y_0)_2$), construire le circuit CMP_{n+1}.

   Réponse. *On fait comme en base* 10*. Quand vous comparez deux nombres de même taille, vous regardez d'abord les chiffres de gauche. Si l'un des deux est plus grand, alors vous pouvez conclure. S'ils sont égaux, vous regardez le chiffre suivant. Donc :*

$$CMP_{n+1}(x, y) = (x_{n+1} \leq y_{n+1}) \vee (x_{n+1} = y_{n+1} \wedge CMP_n(x_0, \ldots, x_n, y_0, \ldots, y_n))$$

   Remarquez que $x_{n+1} = y_{n+1}$ *peut être simplement exprimé par* $\neg(x_{n+1} \oplus y_{n+1})$ *et* $x_{n+1} \leq y_{n+1}$ *n'est rien d'autre que* $CMP_0(x_{n+1}, y_{n+1})$*. À vous de dessiner le circuit !*