

# Master CCI

## Langage machine

### Contrôle continu écrit 2007

Durée 1h30 heure, document autorisés, calculatrices et ordinateurs interdits

## 1 Représentation et arithmétique binaires sur les entiers

**Question** : combien de chiffres hexadécimaux faut-il pour représenter en base 16 un entier quelconque codé sur 16 bits ?

Soit l'entier naturel  $x = 11111111_2$ .

**Question** : écrire l'entier naturel  $x+1$  en binaire et en hexadécimal et  $x$  en décimal.

## 2 Traduction des constructeurs algorithmiques

On rappelle les opérateurs booléens en langage C : **&&** : et, **||** : ou, **!** : not/inversion.

On veut traduire en C équivalent à base de **goto** un squelette de code C. Voici un exemple :

```
/* Squelette à traduire
if (c1 || !c2)
    instruction_alors;
else
    instruction_sinon;
```

Exemple c1 : x égal à 0 c2 : x pair \*/
if ( (x==0) || !(x%2 == 0))
 x = x +1;
else
 y = y -2;

L'évaluation de la condition peut être décrite par un arbre de décision guidant le début de la traduction du code en if ... goto.

```
/* Arbre de décision */
          c1
        /   \
  faux   /     \  vrai
        /         \
        c2       instr_alors
      faux  /   \  vrai
        /     \
  instr_alors  instr_sinon
```

```

/* Début de code goto/if...goto équivalent */
/* variante 1 */                      /* variante 2 */
if ( c1) goto alors;                  if ( c1) goto alors;
if (!c2) goto alors;                  if ( c2) goto sinon;
sinon: instruction_sinon;           alors: instruction_alors;
...                                     ...

```

**Question :** Donner un arbre de décision et une traduction complète en **if ... goto** du squelette suivant. Répondre sur la feuille en fin d'énoncé.

```

avant;
if ( c1 || (!c2 && c3))      /* si c1 ou bien si pas c2 et c3 */
{
    a1;
    a2;
}
else
{
    s1;
    s2;
}
apres;

```

### 3 Feuille de réponse

Nom :

Prénom :

Dessiner ici l'arbre de prise de décision :

Compléter le squelette de code C ci-dessous :

- Ajouter le(s) **if ... goto** ou **goto** manquant.
- Définir les **étiquettes** et spécifier les destinations des **goto**.
- Ajouter si nécessaire l'opérateur ! si la condition de saut d'un **if ... goto** doit être inversée.

```
avant;
if ( c1) goto      ;
if ( c2) goto      ;
```

```
s1;
s2;
```

```
a1;
a2;
```

```
apres;
```