

Questions de contrôle continu (TD3 - semaine du 14 novembre)

Correction

1) Donner la représentation décimale de l'entier dont la représentation hexadécimale est A5C0. C'est $0 + 12 \cdot 16 + 5 \cdot 16^2 + 10 \cdot 16^3 = 192 + 5 \cdot 256 + 10 \cdot 4096 = 42432$

2) Quelle est la différence entre une variable `x` de type `int` et une variable `y` de type `unsigned long` ? Une différence en taille de stockage et en domaine de valeurs. Les entiers `int` sont des relatifs et utilisent 1 bit pour coder le signe. Le type `unsigned long` est celui d'entiers positifs (`unsigned`) et ils sont a priori plus grand (ou égaux) en taille de stockage que les `int`. Leur domaine exact dépend du nombre de bits utilisés sur la plateforme pour coder un `long`. (Mais dans tous les cas, on pourra coder des entiers positifs plus grands que les `int`, car dans le pire des cas, on utilise le même nombre de bits pour coder la valeur et cependant les `int` ayant en fait un bit de moins à cause du signe, le plus grand `int` positif sera plus petit que le plus grand `long`.)

3) Combien y a-t-il de caractères encodés par le code ASCII ? 256 virtuellement, mais historiquement on utilise seulement les 128 premiers pour ce code "standard" (et les autres 128 sont utilisés de manière moins standard selon les plateformes pour coder des caractères accentués, grecs, etc.). (Les deux réponses 128 et 256 seront acceptées).

4) Quelle est le nom de l'instruction suivante ? `x = x + 25` ; Cette instruction s'appelle une *affectation*.

5) Le programme suivant comporte 3 ou 4 erreurs. Indiquez celles que vous trouvez et corrigez-les.

```
#include <stdlib.h>

int main()
{
    int x ;

    x = 0;
    y = 8;

    x = y + x
    printf("x vaut %s",    x);

    return EXIT_SUCCESS;
}
```

- 1- La variable `y` est utilisée avant d'avoir été déclarée. il faut ajouter `int y`;
- 2- L'instruction `x = y + x` devrait être séparée de la suivante par un point-virgule.
- 3- La fonction `printf` n'a pas été déclarée. Il fallait écrire `#include <stdio.h>`.
- 4- Le format `%s` n'est pas approprié pour afficher `x`, car `x` a été déclaré `int`. (Il fallait utiliser le format `%d` des nombres décimaux).