Langace C - Travaux Dirigés n° 1

— Licence Informatique —

▶ **Exercice 1 (Structure minimale et compilation)**

- Écrire le fichier source complet d'un programme permettant à l'utilisateur d'entrer un caractère quelconque au clavier et affichant ce caractère et son code ASCII avant de passer à la ligne.
- Écrire la ligne de compilation permettant de construire un fichier exécutable `firstprgm.exe` à partir du source `firstprgm.c`. Le compilateur devra afficher toutes les erreurs et *warnings*.

▶ **Exercice 2 (Boucle, arrêt de boucle, instruction conditionnelle)**

Perfectionner le programme précédent pour :

- qu'il précise si le caractère entré est une lettre minuscule, majuscule, un chiffre, ou un autre type de caractère.
- qu'il répète la saisie et l'affichage d'un caractère tant que celui-ci n'est pas `<ESC>`.

▶ **Exercice 3 (Évaluation d'expression)**

Que font les fonctions `biz1` et `biz2` suivantes ?

```
int biz1(int a)
{
  return a*((a>0)-(a<0));
}

int biz2(int a, int b)
{
  if (a+=b)
    return a;
  return b;
}
```

▶ **Exercice 4 (Boucle)**

Écrire une fonction qui lit des entiers au clavier jusqu'à ce que l'on entre un caractère autre qu'un entier, puis qui calcule la somme, la moyenne, le minimum et le maximum de tous les entiers entrés.

▶ **Exercice 5 (Coefficient binômiaux)** On donne le programme suivant censé calculer les coefficients du binôme définis par

$$C_n^p = \frac{n!}{(n-p)! \times p!}.$$

```
#include <stdio.h>

int F(int n)
{
  int i,f=1;
  for (i=1;i<n;i++)
    f*=i;
  return f;
}
```

```
int C1(int n,int p)
{
  return F(n)/(F(n-p)*F(p));
}

int main(void)
{
  int n,p;

  scanf("%d%d",&n,&p);
  if (n<p) printf("C(%d,%d) non défini\n",n,p);
  else     printf("C(%d,%d)=%d\n",n,p,C(n,p));

  return 0;
}
```

Que penser de cette méthode de calcul ? Quelle est sa complexité ? Quelle est la valeur maximale calculable ? Proposer d'autres méthodes (non récursives) plus économiques et permettant de calculer des valeurs de C_n^p beaucoup plus grandes.