Travaux Pratiques d'algorithmique n°1

Cours d'Informatique de Deuxième Année

—DEUG Sciences S3-MI—

Révisions de *C*

L'objet de cette séance de travaux pratiques est le rappel des concepts de base du langage *C*. Nous rappelons entre autres : les tests, les boucles, les fonctions et le passage des paramètres.
Toutes les fonctions programmées devront être commentées.

▸ **Exercice 1.** *Compiler et exécuter le programme suivant. Expliquer son fonctionnement.*

```
#include<stdio.h>

int Mystere(int);

int main(void)
{ int ent, res;

  printf("Entrez un entier: ");
  scanf("%d",&ent);
  res = Mystere(ent);
  printf("%d\n",res);
  return 0;
}
```

```
int Mystere(int n)
{ int resultat = 1;
  int i;

  if ( n == 0 ) return 1;
  if ( n > 0 )
     {
       for(i=2;i<n+1;i++)
          resultat *= i;
       return resultat;
     }
  return -1;
}
```

▸ **Exercice 2. (coefficients binomiaux)** *Nous cherchons à calculer les coefficients binomiaux C_n^p définis par les relations suivantes :*

$$C_n^p = \begin{cases} \dfrac{n!}{p!(n-p)!} & \text{si } 0 \leq p \leq n \ , \\ 0 & \text{sinon.} \end{cases}$$

Écrire la fonction `Binomial` *qui prend comme arguments* `n` *et* `p` *et qui calcule le coefficient binomial C_n^p.*

▸ **Exercice 3.** *Nous cherchons à étudier les propriétés des coefficients binomiaux. Nous nous intéressons à la somme suivante :*

$$\sum_{i=0}^{n} C_n^i \quad .$$

Écrire la fonction `Somme` *qui prend en argument* `n` *et qui calcule cette somme. Essayer cette fonction pour $n = 2, 3, 4, 5, \ldots$ En déduire une évaluation de la somme ci-dessus.*

▸ **Exercice 4.** ***(passage par adresse)*** *Programmer et tester les trois fonctions suivantes. Détailler leurs fonctionnements. Qu'en pensez-vous ?*

```
void swap1(int a, int b)
{
  int tmp;
  tmp = a;
  a   = b;
  b   = tmp;
}
```

```
void swap2(int a, int *b)
{
  int tmp;
  tmp = a;
  a   = *b;
  *b  = tmp;
}
```

```
void swap3(int *a, int *b)
{
  int tmp;
  tmp = *a;
  *a  = *b;
  *b = tmp;
}
```

▸ **Exercice 5.** *On considère le programme suivant :*

```
#include <stdio.h>
int main(void)
{
  int *ad1, *ad2, *ad3;
  int n = 10, p = 20;
  ad1 = &n;
  ad2 = &p;
  *ad1 = 3;
  *ad2 = *ad1 + *ad2 + p;
  printf("n = %d, p =  %d\n",n,p);
  printf("*ad1 = %d, *ad2 =  %d\n",*ad1,*ad2);
  printf("adresse de n=%p\n",&n);
  printf("valeur de ad1=%p\n",ad1);
  printf("adresse de ad1=%p\n",&ad1);
  *ad3 = *ad2;
  printf("*ad2 = %d, *ad3 =  %d\n",*ad2,*ad3);
  return 0;
}
```

Ce programme est syntaxiquement correct mais ne marche pas à l'exécution. Pourquoi ?

▸ **Exercice 6.** *Le but de cet exercice est d'implémenter les opérations de base sur les tableaux.*

- *Écrire une fonction permettant de saisir un tableau d'entiers. Le prototype de la fonction sera* `int SaisirTab(int *t, int *nb)` *où* `t` *est le tableau contenant les entiers et* `nb` *est l'adresse de la variable contenant le nombre d'éléments du tableau. La fonction retournera* 1 *si la saisie s'est réalisée correctement,* 0 *sinon ;*
- *écrire une fonction de prototype* `void AffichageTab(int *t, int nb)` *qui affiche le contenu du tableau* `t` *de taille* `nb` *;*
- *écrire une fonction* `Somme` *qui somme les éléments du tableau.*
- *écrire une fonction* `Moyenne` *qui retourne la moyenne des valeurs contenues dans un tableau* `t` *de taille* `nb`*.*
- *écrire une fonction de somme des éléments d'indice pair, puis une autre qui somme les éléments dont la valeur est paire.*