Travaux Pratiques d'algorithmique n°1

Cours d'Informatique de Deuxième Année

—DEUG Sciences S3-MI—

Représentation d'entiers

Dans cette première séance, nous abordons les points suivants :

- l'écriture d'un entier naturel en base 2, (puis en base b, $b > 1$),
- la représentation en complément à 2 d'un entier,
- rappels des concepts de base du langage C (instructions, fonctions, tableaux, entrés-sorties, ...)

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;
  for(i=2;i<n+1;i++)
     resultat *= i;
  return resultat;
}
```

On rappelle que l'écriture en base 2 d'un entier naturel n est la suite $bin(n) = (r_k, r_{k-1}, \ldots, r_2, r_1, r_0)$ telle que :

- $n = r_k \times 2^k + r_{k-1} \times 2^{k-1} + \cdots + r_2 \times 2^2 + r_1 \times 2^1 + r_0$, (1)
- $\forall i$, $(0 \leq i \leq k)$, on a $0 \leq r_i < 2$.

▸ **Exercice 2.** *L'algorithme suivant permet d'obtenir la suite $bin(n)$: On effectue la division entière de n par 2. On trouve un quotient q_0 et un reste r_0. On recommence la division avec q_0.*

On arrête le processus dès que l'on trouve un quotient q_k égal à 0. L'écriture de n en base 2 est alors la suite des restes pris à l'envers, $(r_k, r_{k-1}, \ldots, r_2, r_1, r_0)$. Par exemple pour $n = 11$:

$$\begin{array}{rcrcrcl} 11 & = & 5 \times 2 & + & 1 & \longrightarrow & q_0 = 5 \text{ et } r_0 = 1 \\ 5 & = & 2 \times 2 & + & 1 & \longrightarrow & q_1 = 2 \text{ et } r_1 = 1 \\ 2 & = & 1 \times 2 & + & 0 & \longrightarrow & q_2 = 1 \text{ et } r_2 = 0 \\ 1 & = & 0 \times 2 & + & 1 & \longrightarrow & q_3 = 0 \text{ et } r_3 = 1 \end{array}$$

L'écriture de 11 en base 2 est donc 1011.

1. *Donner l'écriture en base 2 de 18 et 23.*
2. *Déduire de l'algorithme ci-dessus une fonction qui, étant donné un nombre naturel n, calcule la suite $bin(n)$ ainsi que sa longueur. La fonction retournera VRAI si l'opération s'est bien passée, FAUX sinon.*

▸ **Exercice 3.** *On veut maintenant réaliser l'opération inverse, c'est-à-dire, calculer n à partir de la suite $bin(n)$. On suppose que la suite est connue et mémorisée dans un tableau de taille effective `taille`. On propose deux méthodes.*

- *Écrire une fonction qui, étant donnés deux entiers non négatifs b et i, calcule b^i. En déduire une fonction qui calcule l'entier n défini en (1) (méthode naïve).*
- *Résoudre le problème en utilisant le schéma de Horner (méthode efficace) :*

$$n = (\cdots((r_k) \times 2 + r_{k-1}) \times 2 + \cdots + r_1) \times 2 + r_0 \ .$$

▸ **Exercice 4.** *Écrire une fonction permettant de saisir la suite $bin(n)$ au clavier. La fonction retournera VRAI si la saisie s'est bien effectuée et FAUX sinon.*
Écrire un programme complet (avec un menu) permettant de tester toutes vos fonctions.

▸ **Exercice 5.** *Refaire les exercices précédents en considérant une base quelconque b, $1 < b \leq 36$. On convient que les lettres majuscules $A, B, \ldots$ représentent les chiffres compris entre 10 et $b - 1$ en base b.*

▸ **Exercice 6.** *On rappelle que la représentation de l'entier signé n en complément à 2 en utilisant `MAX` bits est :*

- *l'écriture en base 2 de n (complétée par des 0 en tête), si $n \geq 0$;*
- *l'écriture en base 2 de $n + 2^{\texttt{MAX}}$, si $n < 0$.*

Dans une telle représentation, le bit le plus à gauche vaut 1 si $n < 0$ et vaut 0 sinon.

- *Écrire une fonction qui calcule la représentation en complément à 2 d'un entier signé n. La fonction retournera VRAI si l'opération s'est bien passée, FAUX sinon.*
- *Écrire une fonction qui effectue l'opération inverse.*