

Tableaux et fonctions

Exercice 1 – Palindrome

1. Ecrire un programme qui teste si une chaîne de caractères est un palindrome (on rappelle qu'un palindrome est une chaîne qui se lit de la même façon de gauche à droite et de droite à gauche).
2. En déduire un programme qui affiche les années palindromiques de 1 à n. Une année est palindromique si la chaîne de caractères correspondante est un palindrome. On fera un affichage au fur et à mesure (exemples d'années palindromiques : 2, 66, 161, 1001, 1991)

Exercice 2 – Devinez à quoi je pense (c'est +, c'est -) ?

Ecrire un programme qui initialise x à une valeur arbitraire comprise entre 1 et 100 et demande à l'utilisateur de le deviner. Indiquez entre chaque essai si le nombre donné est soit le bon, soit plus grand ou plus petit.

Exercice 3 – Recherche dichotomique

Écrire une fonction itératives (puis récursive) de recherche dichotomique d'un nombre x dans un tableau de n nombres. La fonction renvoie VRAI si l'élément se trouve dans le tableau et FAUX sinon.

Comparer le temps d'exécution entre les deux méthodes.

Exercice 4 - PGCD et PPCM de 2 entiers

Définir les fonctions itérative et récursive du PGCD(a,b) qui calcule le plus grand commun diviseur d de 2 entiers a et b.

Remarques :

- L'algorithme d'Euclide concerne le calcul du plus grand commun diviseur (pgcd) de 2 entiers.
- En déduire une fonction qui calcule le plus petit commun multiple (PPCM) m de 2 entiers a et b.

Le plus grand commun diviseur de 2 entiers a et b peut se calculer en appliquant la relation de récurrence $\text{pgcd}(a, b) = \text{pgcd}(b, a \% b)$ si $b \neq 0$ jusqu'à ce que le reste ($a \% b$) soit nul ($\text{pgcd}(d, 0) = d$ si $d \neq 0$).

$$\text{pgcd}(a, b) = \text{pgcd}(b, a \bmod b) = \dots = \text{pgcd}(d, 0) = d.$$

Remarque : Le calcul du PPCM de deux entiers positifs se déduit du calcul de leur PGCD parce que le produit du PPCM par le PGCD est égal au produit des deux nombres.

Exercice 5 – Coefficients du binôme

- Ecrire une fonction qui calcule le binôme $(a + b)^n$:

$$(a + b)^n = \sum_{k=0}^n C_n^k a^k b^{n-k}$$

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

Propriétés

- 1) $C_n^n = C_n^0 = 1$
 - 2) $C_n^p = C_n^{n-p}$
 - 3) $C_n^p + C_n^{p+1} = C_{n+1}^{p+1}$ ie $C_n^p + C_n^{p-1} = C_{n+1}^p$
 - 4) $C_n^p = \frac{n}{p} C_{n-1}^{p-1}$
- Ecrire un programme qui calcule les premières lignes du triangle de Pascal et qui affiche le résultat à l'écran. Soit le triangle de Pascal jusqu'à l'ordre n : Il s'agit des coefficients du binôme $(a + b)^n$
- ```

1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
1 6 15 20 15 6 1

```
- Le triangle de Pascal se construit comme suit :
- placer dans la première colonne des 1 à chaque ligne, et des 1 à chaque entrée de la diagonale
  - en partant du haut et en descendant, compléter le triangle en sommant deux coefficients adjacents d'une ligne pour produire le coefficient de la ligne inférieure, en dessous du coefficient de droite

## Exercice 6 – Conversion en base b

Un entier n en base b est représenté par une suite de chiffres  $(r_m r_{m-1} \dots r_1 r_0)_b$  où les  $r_i$  sont des chiffres de la base b  $0 \leq r_i < b$ . Ce nombre n a pour valeur On a :  $\sum_{i=0}^m r_i b^i$

Nous choisirons ici de représenter un tel nombre par un tableau  $[(r_m, r_{m-1}, \dots, r_1, r_0)]$

L'algorithme de conversion consiste à diviser successivement le nombre n par la base b ( $n = bq + r$ ) tant que le quotient q n'est pas nul.

L'ordre inverse des restes des différentes divisions (du dernier au premier reste, écrits de gauche à droite) donne la représentation du nombre n en base b.

$$\begin{aligned}n &= q_0b + r_0 = q_0b^1 + r_0b^0 \\n &= (q_1b^1 + r_1)b^1 + r_0b^0 = q_1b^2 + r_1b^1 + r_0b^0 \\n &= (q_2b^1 + r_2)b^2 + r_1b^1 + r_0b^0 = q_2b^3 + r_2b^2 + r_1b^1 + r_0b^0 \\n &= \dots \\n &= 0 \cdot b^{m+1} + r_mb^m + \dots + r_2b^2 + r_1b^1 + r_0b^0 = \sum_{i=0}^{i=m} r_ib^i \\n &= (r_mr_{m-1} \dots r_1r_0)_b\end{aligned}$$

## Exercice 7 – Opérations sur matrices

- Ecrire une fonction qui Lit une matrice M carrée de taille (n,r)
- Ecrire une fonction permettant d'effectuer la multiplication de deux matrices M1 et M2 respectivement de dimensions (n, r) et (r,m). Le résultat de la multiplication sera mémorisé dans une troisième matrice M3 qui sera ensuite affichée.