

Cours de Programmation en C – EI-2I

Travaux Pratiques

Objectif(s)

- ★ Rappels de C (types de données, opérateurs)
- ★ Entrées/Sorties
- ★ Boucles et traitement de file

Exercice 1 – FizzBuzz

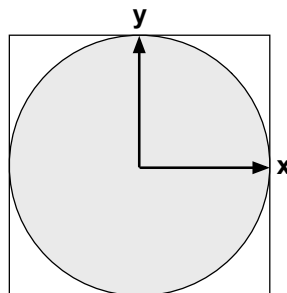
Écrire dans un fichier `fizzbuzz.c` un programme qui affiche les nombres de 1 à 100, sauf:

- si le nombre est un multiple de 3 ou se termine par 3, auquel cas il affiche "Fizz"
- si le nombre est un multiple de 7 ou se termine par 7, auquel cas il affiche "Buzz"
- si le nombre vérifie les 2 propositions, le programme affiche "FizzBuzz"

Le tester, et vérifier que l'on obtient bien (pour les 1ères valeurs) "1 2 Fizz 4 5 Fizz Buzz 8 Fizz 10 11 Fizz Fizz Buzz Fizz 16 Buzz Fizz"

Exercice 2 – Calcul approché de π

On se propose de calculer la valeur de π en utilisant la méthode de Monte-Carlo. Cette méthode consiste à tirer au hasard N points de coordonnées (x,y) , dans un carré de côté 2 et centré en $(0,0)$.



On calcule le nombre total de points qui sont tombés à l'intérieur du disque inscrit dans le carré (c'est-à-dire du disque centré en $(0,0)$ et de rayon 1). On note p le nombre de points à l'intérieur du disque. Le rapport $\frac{p}{N}$ tend (lorsque $N \rightarrow \infty$) vers le rapport des surfaces :

$$\frac{p}{N} \approx \frac{S_{\text{disque}}}{S_{\text{carré}}} = \frac{\pi}{4} \quad (1)$$

Ainsi, pour N suffisamment grand, nous pouvons avoir une approximation de π par $\pi \approx 4 \frac{p}{N}$.

1. Écrire l'équation qui permettent de déterminer si un point de coordonnées (x,y) est dans le cercle de centre $(0,0)$ et de rayon 1.
2. En déduire un programme dans un fichier `pi.c` qui calcule une approximation de π . On utilisera pour cela la fonction `rand`¹.

¹la commande `man 3 rand` vous indiquera comment l'utiliser.

Exercice 3 – Calcul de puissance

1. Écrire dans un fichier `puiss1.c` le programme permettant de calculer et afficher x^k (x est un réel et k un entier positif ou nul) en utilisant une boucle (k multiplications), x et k étant demandés à l'utilisateur.
2. On peut aussi calculer plus efficacement (c.-à-d. avec moins d'opérations) en utilisant le fait que $x^{2p} = (x^2)^p$ et $x^{2p+1} = x.(x^2)^p$. Écrire un nouveau programme `puiss2.c` utilisant cette décomposition.
3. Dans le cas du calcul de x^{18} , combien de tours de boucles fait-on (dans l'une et l'autre version). Dans le cas général (k quelconque) ? Indiquez vos réponses en commentaire.

Exercice 4 – Conversion hexadécimale

On souhaite écrire un programme qui lit un nombre hexadécimal comme une suite de caractères, vérifie sa validité et effectue la conversion décimale. L'itération s'arrêtera quand on lira un caractère ne représentant pas un chiffre hexadécimal, et on calculera la valeur au fur et à mesure.

Écrire dans le fichier `hexa.c` le code C correspondant.

Exercice 5 – Fractions égyptiennes

Toute fraction peut s'écrire comme une somme de fractions ayant 1 comme numérateur. Cette décomposition est appelée décomposition en fractions égyptiennes. En voici un exemple :

$$\frac{7}{8} = \frac{1}{2} + \frac{1}{3} + \frac{1}{24}$$

Écrire un programme `frac.c` prenant en entrée le numérateur a et le dénominateur b d'une fraction et affiche sa décomposition en fractions égyptiennes (cette décomposition n'est pas unique).