LYCÉE SSB MOKNINE

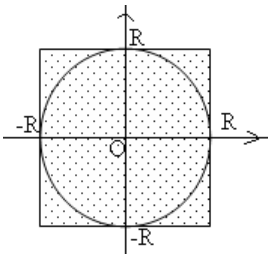
Mr : Karchoud Radhouane **4 SI : Algorithmique et Programmation**

Série 16 (approximation)

Exercice 1 :

On se propose de calculer une valeur approchée de π en utilisant les trois formules suivantes :

Formule1 : (Proportion de points dans un cercle)



Une des méthodes d'approximation de **π** consiste à **remplir au hasard** un carré **Q** de côté **2R** par **n** points. Parmi ces **n** points, il y aura **p** points qui seront à l'intérieur du cercle inscrit dans le carré **Q** (de coté = 2) comme illustré par la figure ci-contre.

En déterminant **p**, on pourra calculer une valeur approchée de **π** vu que le rapport entre le nombre **p** de points à l'intérieur du cercle inscrit et le nombre total **n** de points du carré est une estimation du rapport entre la surface du cercle et la surface du carré **Q**. Nous avons donc :

$p/n \cong \pi R^2 / 4R^2 = \pi/4$ ce qui donne **$\pi \cong 4p/n$**

On rappelle qu'un point M(x, y) est à l'intérieur du cercle si $\sqrt{x^2+y^2}\sqrt{x^2+y^2} \leq R$

NB : Pour générer un nombre réel au hasard entre deux bornes, on utilise la fonction : **uniform(Vi, Vf)**

Exemple :

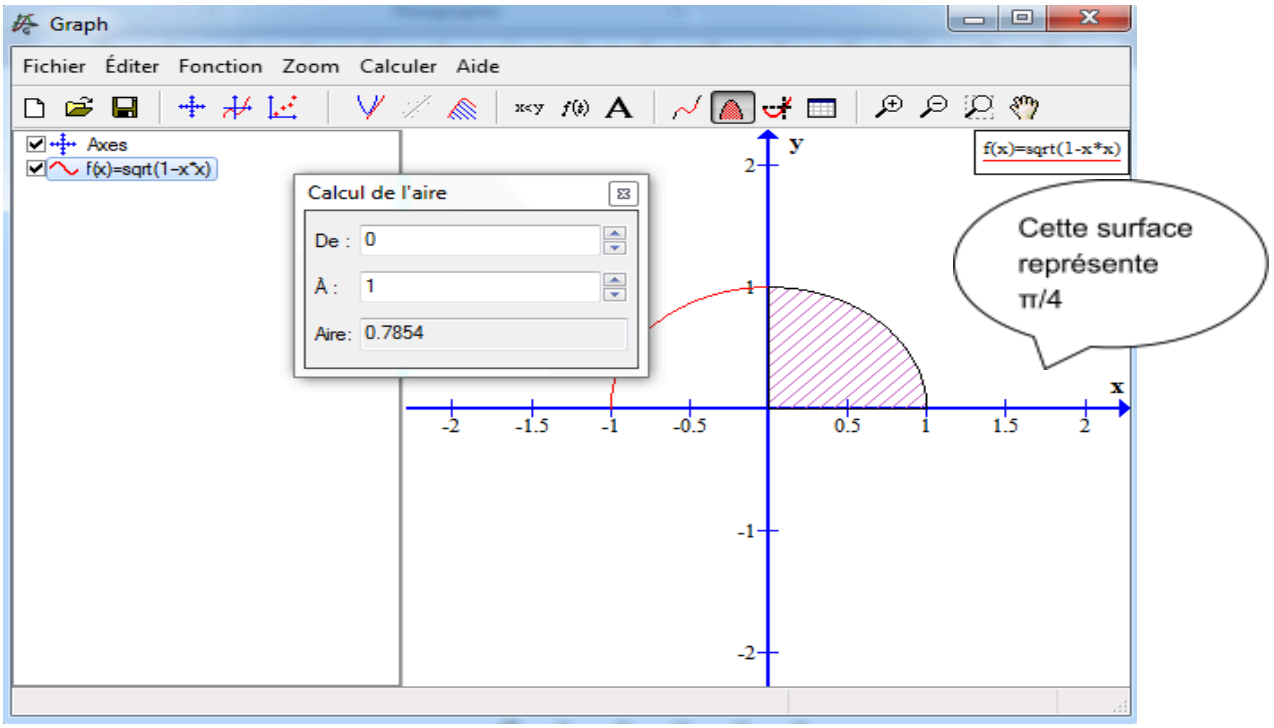
pour choisir un réel au hasard compris entre 3 et 20, on utilise la syntaxe suivante :

x □ uniform(3,20)

Formule2 (La méthode de Simpson)

Sachant que : $\frac{\pi}{4} = \int_0^1 F(x) \frac{\pi}{4} = \int_0^1 F(x)$ dx avec F(x) = $\sqrt{1-x^2}\sqrt{1-x^2}$

Pour calculer une valeur approchée de l'aire de la courbe de la fonction **F,** on utilise la méthode de **Simpson** qui consiste à chercher l'aire délimité par la courbe représentatif de F et l'axe des abscisses entre deux droites qui passent respectivement par les droites F(x) = 0 et F(x) = 1.



Formule3 (La méthode de suite)

Soit une suite définie par les premiers termes suivants :

$$\begin{cases} U_1 = \sqrt{2} \\ U_n = \sqrt{2+U_{n-1}} \quad \text{pour n>=2} \end{cases}$$

On propose la formule suivante : $\frac{\pi}{2}\frac{\pi}{2} = \frac{2}{U_1} * \frac{2}{U_2} * \frac{2}{U_3} * \frac{2}{U_4} * \frac{2}{U_5} * \ldots\ldots\ldots$

Le calcul s'arrête lorsque la différence entre deux valeurs de cette formule devient inférieure ou égale à une valeur très faible eps.

Ecrire un programme modulaire qui permet de :

- Saisir un entier **n** (nombre de **subdivision**) avec **10000 ≤ n ≤ 20000**.
- Saisir un réel **eps** avec $\mathbf{10^{-5} \leq eps \leq 10^{-3}}$.
- Créer et remplir un fichier texte « **pi.txt** » contenant :
  - Votre **nom et prénom** dans la première ligne.
  - Dans les trois autres lignes, on écrit respectivement les résultats suivants :
    - Valeur de **π** par la **formule1. (**En développant une fonction **valeur_pi_cercle(n)** qui permet de retourner une approximation de **π** en utilisant la **formule1).**
    - Valeur de **π** par la **formule2**. (En développant une fonction **valeur_pi_simpson(a,b,n)** qui permet de retourner une valeur approchée de **π** en utilisant la **formule2**).
    - Valeur de **π** par la **formule3. (**En développant une fonction **valeur_pi_suite(eps)** qui utilise la **formule3** pour retourner la valeur approchée de **π** à **eps** près**).**
- Afficher le fichier « **pi.txt** ».

Exemple :

Pour n = 15000 et eps= 0.00001

Le contenu du fichier « pi.txt » est :

```
Foulen ben Foulen
Pi_cercle= 3.1314666666666667
Pi_simpson= 3.14159256520705
Pi_suite= 3.1415914215111997
```

Travail demandé :

1- Ecrire l'algorithme du programme principal en le décomposant en modules.
2- Ecrire les algorithmes des modules envisagés dans votre solution.

NB : dresser les tableaux de déclarations nécessaires.

Exercice 2 :

1- Ecrire un algorithme d'une fonction intitulée **Premier_Suivant** qui prend en paramètre un entier a premier et retourne l'entier premier qui lui suit.

Exemple :

Premier_Suivant (23) retourne 29 (car 29 est l'entier premier qui suit l'entier premier 23).

2- Soit **S** une constante qui est égale à :

$$S = \frac{3}{1!} - \frac{5}{2!} + \frac{7}{3!} - \frac{11}{4!} + \frac{13}{5!} - \frac{17}{6!} + \frac{19}{7!} - \frac{23}{8!} \ldots$$

En utilisant la fonction **Premier_Suivant**, écrire un algorithme d'un module qui permet de déterminer une valeur approchée de **S** à ε près.

Exercice 3 :

On souhaite calculer une valeur approximative de la constante π (pi) en utilisant les formules d'**Isaac**

En mathématiques, si **N** est un nombre décimal, **-N** est appelé l'opposé de **N** (-5 est l'opposé de 5).
En binaire pour obtenir l'opposé de N, on utilise l'une de deux méthodes suivantes :

- **1ere Méthode** : Elle est appelée **complément à 2 de N** : C'est une manière de représenter en binaire (sur 8 bits) les entiers négatifs.
  Cette méthode consiste à **inverser** tous les 1 par 0 et tous les 0 par 1, puis on **ajoute** 1.

Exemple 1 :	**Exemple 2 :**
Le nombre décimal **24** vaut en binaire : 11000	Le nombre décimal **6** vaut en binaire : 110
➔Sur 8 bits on obtient : 00011000	➔Sur 8 bits on obtient : 00000110
➔On inverse les bits : 11100111	➔On inverse les bits : 11111001
➔On ajoute 1 : 11100111+1 = 11101000	➔On ajoute 1 : 11111001+1 = 11111010
Donc -24 en binaire est : **11101000**	*Donc -6 en binaire est :* **11111010**

↳ **N.B.** En binaire 0 + 0 = 0, 1 + 0 = 1 et 1 + 1 =10

- **2ème Méthode** : On garde tous les chiffres 0 depuis la droite jusqu'au premier 1 (compris) puis on inverse tous les suivants (à gauche).

Exemple 1 :	**Exemple 2 :**
Le nombre décimal **24** vaut en binaire : 11000	Le nombre décimal **6** vaut en binaire : 110
- Sur 8 bits on obtient : 00011000	- Sur 8 bits on obtient : 00000110
- Trouver le premier 1 depuis la droite : 00011000	- Trouver le premier 1 depuis la droite : 00000110
- Inverser les restes : 11101000	- Inverser les restes : 11111010
Donc -24 en binaire est : **11101000**	*Donc -6 en binaire est :* **11111010**

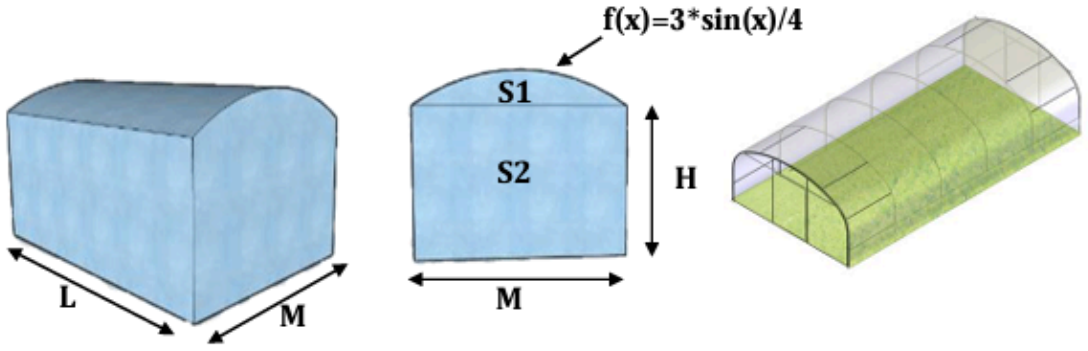
1. Écrire un algorithme une fonction nommée **Méthode1** permettant de déterminer l'opposé d'un nombre décimal donné et ceci en utilisant le principe de 1ere méthode
2. Écrire un algorithme une fonction nommée **Méthode2** permettant de déterminer l'opposé d'un nombre décimal donné et ceci en utilisant le principe de 2ere méthode

Exercice 4

Exercice 5 :

Dans le cadre de l'encouragement à l'utilisation des serres agricoles, l'État fixe des aides (des primes) aux agriculteurs pratiquant ce type d'agriculture. Ces aides incitatives sont valorisées en fonction du volume de serres utilisées par chaque agriculteur.

Une serre agricole est de longueur **L**, de largeur **M**, de hauteur **H**, et sa forme est obtenue par la représentation graphique suivante :



Le volume d'une serre est égal à **(S1+S2) * L**, avec :

- La surface **S1 = H * M**,
- La surface **S2** est obtenue par la représentation graphique de la fonction **f (x) = (3 * sin(x)) / 4** sachant que **x∈ [0..M]**.

Travail demandé :

1) Écrire l'algorithme d'une fonction qui permet de calculer une valeur approchée de **sin(x)** à 10^{-8} prés en utilisant la formule suivante :

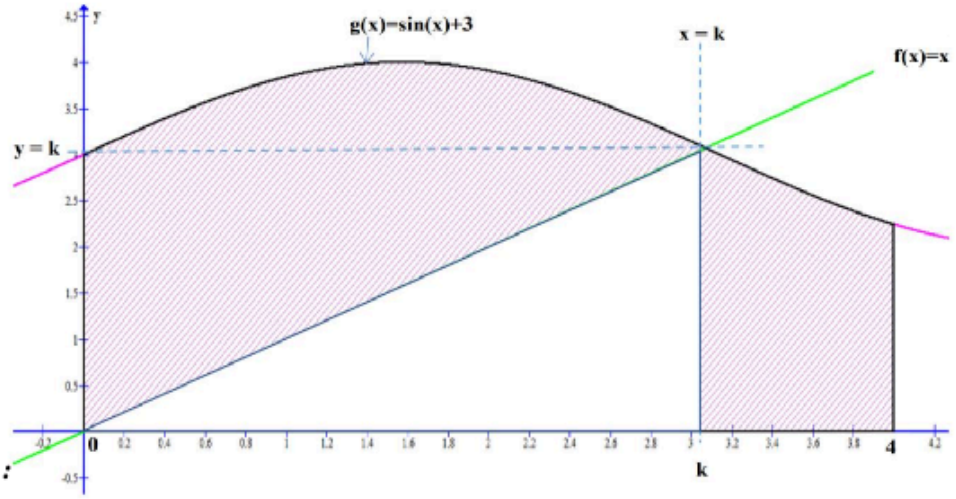
$$\sin(x) = \frac{x}{1!} - \frac{x3}{3!} + \frac{x5}{5!} - \frac{x7}{7!} + \frac{x9}{9!} - \ldots\ldots$$

2) Pour aider les paysans à calculer la valeur de leurs primes, écrire l'algorithme d'un module qui permet de calculer une valeur approchée du volume d'une serre pour un nombre de divisions **N** donné.
N.B. : L, **M**, **H** et **N** sont passés en paramètres.

Exercice 6 :

Soient les deux fonctions **f** et **g** définies comme suit : $\begin{cases} f(x) = x \\ g(x) = \sin(x) + 3 \end{cases}$

Soit les courbes représentatives des deux fonctions **f** et **g** :



Travail demandé :

1- Ecrire un algorithme d'une fonction **calcul (e)** qui permet de calculer une valeur approchée de **k** à epsilon près, tel que **k = sin(k)+3**.

2- Ecrire un algorithme d'une fonction **surface (e)** qui permet de calculer une valeur approchée de la partie hachurée à epsilon près (dans la figure ci-dessus) en utilisant la méthode des rectangles (au point milieu)
NB. e : epsilon est saisie dans le programme appelant