TD/TP 3 : Boucles

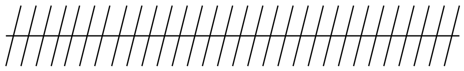
Gilles Peskine

18 octobre 2002

1 Premières boucles

(a) Écrire un programme qui demande à l'utilisateur d'entrer des nombres. Lorsque l'utilisateur entre un caractère autre qu'un chiffre ou un espace, le programme doit afficher la somme des nombres entrés et se terminer.

(b) Écrire un programme qui dessine la figure suivante.



(c) Écrire un programme qui demande un entier n et calcule la somme des carrés des entiers de 1 à n.

2 Polygones réguliers

Écrire un programme qui lit un nombre entier n et dessine un polygone régulier à n côtés. Utiliser au choix la bibliothèque `graphics` directement ou la tortue de l'exercice 6 du TP 2.

Note : avant de se lancer dans le programme, il faudra savoir calculer les coordonnées des sommets. Les fonctions mathématiques habituelles `sin`, `cos`, etc. sont déclarées dans `math.h`.

3 Suite de Syracuse

On définit la fonction f sur les nombres entiers naturels par :

$$\begin{aligned} f(x) &= \frac{x}{2} \qquad \text{si } x \text{ est pair} \\ f(x) &= 3x+1 \end{aligned}$$

La suite de Syracuse est définie de la manière suivante : on choisit un nombre entier strictement positif u_0, et $u_{n+1} = f(u_n)$ pour tout $n \geq 0$. Écrire un programme qui lit u_0 et affiche les valeurs successives u_0, u_1, u_2, etc. Le programme devra se terminer si $u_n = 1$.

Note : on conjecture que quel que soit le nombre entier $u_0 > 0$, il existe un entier n tel que $u_n = 1$. Mais on ne sait pas le démontrer...

4 Graphe d'une fonction

Écrire une fonction qui dessine le graphe de la fonction `f` dont le prototype est `double f(double x)`. Utiliser cette fonction dans un programme dans lequel `f` est définie par $f(x) = 1 - x^2$, puis dans lequel f est la fonction sinus.

5 Fonction factorielle

On définit la fonction *factorielle* sur les nombres entiers positifs de la manière suivante :

$$0! = 1 \text{(on lit : « la factorielle de 0 est 1 »)}$$
$$n! = n * (n-1)! \text{pour } n > 0$$

Écrire une fonction qui calcule la factorielle d'un entier, la tester. Que se passe-t-il lorsque n est grand (plus de 13) ?

Afin d'obtenir des valeurs approchées des factorielles de nombres plus élevés, changer le programme pour qu'il utilise le type `double`.[1].

6 Calcul de e

(Exercice destiné à ceux qui aiment les mathématiques.)

Le nombre e est défini comme la limite de la somme infinie

$$e = \frac{1}{0!} + \frac{1}{1!} + \frac{1}{2!} + \ldots + \frac{1}{n!} + \ldots$$

Utiliser la fonction `factorielle` de l'exercice 5 pour calculer une valeur approchée de e. Pourquoi n'est-ce pas une bonne méthode ? Proposer une manière de calculer l'expression ci-dessus avec la meilleure précision possible.

[1] Le type `double` est analogue au type `float`, mais a deux fois plus chiffres significatifs