Mathématica Cours 7

Le texte écrit en police `Typewriter` correspond au langage mathématica

Tant que

`While[`$c, i_1; i_2; \cdots ; i_n$`]` signifie : tant que la condition c est vérifiée, faire les instructions $i_1, i_2, \cdots , i_n$: il n'y a qu'une virgule dans un While !

1. Taper le programme suivant, valider $f(1)$ et commenter :

```
f[x_]:=Module[{a=x},
While[a<10,Print["a=",a," essaye encore"];a=2a]
;a]
```

{a=x} signifie que a est une variable locale initialisée à la valeur x

Le problème $3n + 1$

Soit x un entier naturel non nul, on considère le programme suivant :

$n = x$ (initialisation)
tant que $n \neq 1$ faire :
Si n pair, $n \leftarrow n/2$
Si n impair, $n \leftarrow 3n + 1$

$n \leftarrow n/2$ signifie que n est remplacé par $n/2$, ce que l'on note `n=n/2` en informatique : le nouveau n écrase l'ancien
La conjecture de Syracuse dit que le programme termine toujours, c'est à dire que n finit toujours par valoir 1 : elle n'est toujours pas démontrée à ce jour !

2. Vérifier la conjecture de Syracuse pour $x = 7$ sans utiliser Mathématica

3. Taper une fonction d'entrée x et qui affiche les différentes valeurs de n en utilisant `Print` et la tester sur des exemples (utiliser la commande `EvenQ` pour les questions de parité)

Le procédé dichotomique

4. Soient g la fonction définie par $g(x) = cos(x) - x$, rentrer la fonction g puis tracer sa courbe représentative, x variant variant de 0 et $\pi/2$

5. On admet que g s'annule en un réel α, faites plusieurs zooms sur le graphique précédent pour trouver deux réels notés a et b vérifiant $a \leq \alpha \leq b$ et $b - a \leq 0,001$

Plus généralement, soit $(a, b) \in \mathbb{R}^2$ avec $a < b$ et f une fonction continue définie de $[a, b]$ vers $\mathbb{R}$
On suppose que f change de signe entre a et b, ce qui se traduit mathématiquement par l'inégalité $f(a)f(b) \leq 0$, f s'annule donc d'après le théorème des valeurs intermédiaires

(éventuellement plusieurs fois)

Soit ε un réel > 0 donné, le but de la suite est de donner un encadrement d'un réel α vérifiant $f(\alpha) = 0$ à la précision ε en utilisant le procédé dichotomique

tant que $b - a > \varepsilon$

calculer $c = (a + b)/2$
si $f(a)f(c) \leq 0$ remplacer $[a, b]$ par $[a, c]$ sinon remplacer $[a, b]$ par $[c, b]$

Noter que a et b sont modifiés au cours du programme. On appelle longueur du segment $[a, b]$ le nombre $b - a$. D'une étape à l'autre, $[a, b]$ est remplacé par le segment $[a, c]$ ou $[c, b]$ qui est d'une longueur deux fois plus petite : à la fin de la $n$ième étape, la longueur du segment $[a, b]$ est donc de $(b_0 - a_0)/2^n$, a_0 et b_0 désignant les valeurs initiales de a et b

Si $f(a)f(c) \leq 0$ alors f s'annule entre a et c sinon f s'annule entre c et b : à la fin, c'est à dire dès que $b - a \leq \varepsilon$, f s'annule en un réel α vérifiant $a \leq \alpha \leq b$ et $b - a \leq \varepsilon$

6. Ecrire une fonction dicho d'entrée f, a, b, ε qui renvoie un segment $[a, b]$ contenant un réel α tel que $f(\alpha) = 0$ (a et b étant modifiés au cours du programme), $[a, b]$ vérifiant $b - a \leq \varepsilon$: pour cela, on évitera d'entrer la fonction `dicho[f_,a_,b_,eps_]` car alors a et b sont des entrées que l'on ne peut pas modifier
On présentera plutôt a et b comme des variables locales du module sur lesquelles on agit en les initialisant aux véritables valeurs de a et b (nommées $a1$ et $b1$) en procédant comme suit :

```
dicho[f_,a1_,b1_,eps_]:=
Module[{a=a1,b=b1,c},···
```

7. Reprendre la question 5 grâce à la fonction dicho

Densité d'une partie de $\mathbb{R}$

Soit $A = \{\sqrt{n} - \sqrt{p}\,(n, p) \in \mathbb{N}^2\}$, on admet que A est une partie dense dans $\mathbb{R}$, c'est à dire :
$\forall (a, b) \in \mathbb{R}^2, a < b \Rightarrow \exists x \in A, a < x < b$
Soit donc $a = 3,141$ et $b = 3,142$, on cherche n et $p \in \mathbb{N}$ tels que $x = \sqrt{n} - \sqrt{p}$ vérifie $a < x < b$

On va pour cela faire dérouler les entiers naturels (n, p) en notant $s = n + p$
s, initialisée à 0, varie de un en un (ce que l'on note `s=s+1` en informatique)
s étant fixé, n varie de 0 à s (on utilisera donc un `Do` pour n), p s'en déduit par la formule $p = s - n$

On note B la variable initialisée à F et qui vaut V dès que $x = \sqrt{n} - \sqrt{p}$ vérifie $a < x < b$, B s'appelle un booléen

8. Programmer le calcul de n et p de cette façon, la structure générale sera donc :

```
s=0;B=F;
While[B==F,···;s=s+1]
```

Le "et" logique est noté &&