

Informatique tronc commun

Algorithmique

Sujet

30 septembre 2006

1 Syntaxe d'un programme en Maple

<pre>p:= proc(x) local a,b,c ; <suite d'instructions> end;</pre>	déclaration d'une procédure p de variable x variables locales de la procédure programme de la procédure le résultat envoyé = celui de la dernière instruction exécutée
<pre>a := <expression></pre>	affectation de la variable locale a
<pre>if <condition> then <suite d'instructions 1> else <suite d'instructions 2> fi;</pre>	instructions conditionnelles
<pre>for i from <debut> by <pas> to <fin> do <suite d'instructions> od;</pre>	boucle for i commence à deb , finit à fin et augmente de pas à chaque itération ($pas=1$ par défaut)
<pre>while <condition> do <suite d'instructions> od;</pre>	boucle while

2 Boucles simples

Exercice 1 Écrire une fonction $ordre(k, n)$ qui renvoie l'ordre d'un élément k dans le groupe multiplicatif $(\mathbb{Z}/n\mathbb{Z})^*$. (On supposera que k est bien inversible modulo n .)

Exercice 2 On représente un polynôme $P \in \mathbb{Z}[X]$ de degré n par le tableau $a[i], i \in [1, n]$ de ses coefficients ($a[i]$ contenant le coefficient de X^i). Écrire une fonction $evaluate(a, x)$, ou $evaluate(a, n, x)$, qui calcule la valeur de P en un point $x \in \mathbb{Z}$, en temps $O(n)$.

3 Algèbre linéaire et suite de Fibonacci

On rappelle que la suite de Fibonacci $(F_n)_{n \in \mathbb{N}}$ est définie par $F_0 = F_1 = 1$ et $\forall n \in \mathbb{N}, F_{n+2} = F_{n+1} + F_n$.

Exercice 3 Écrire une fonction $Fa(n)$ qui calcule récursivement F_n à l'aide de la définition ci-dessus. Calculer F_{30} . Regarder la barre d'état de Maple. Évaluer (ou minorer) la complexité de votre fonction pour rendre compte de ce que vous observez.

Exercice 4 Écrire une fonction $Fb(n)$ de complexité linéaire en n qui calcule itérativement F_n .