Chapitre 1
Programmation récursive

Exercices

Exercices d'application

TD d'informatique du Lycée Louis Legrand – Jean-Pierre Becirspahic

`http://info-llg.fr/`

Savoirs et compétences :

❐ *Alg – C15 : Récursivité : avantages et inconvénients.*

Exercice 1

On considère la fonction récursive suivante :

■ **Python**
```python
def f(n) :
    if n>100 :
        return n-10
    return f(f(n+11))
```

Question *Prouver sa terminaison lorsque $n \in \mathbb{N}$ et déterminer ce qu'elle calcule (sans utiliser l'interpréteur de commande).*

Exercice 2

Question *Prouver la terminaison de la fonction G de Hofstadter, définie sur $\mathbb{N}$ de la façon suivante :*

■ **Python**
```python
def g(n) :
    if n== 0 :
        return 0
    return n-g(g(n-1))
```

Exercice 3

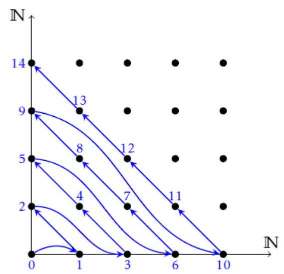
Question *Écrire une fonction récursive qui calcule a^n en exploitant la relation : $a^n = a^{n/2} \times a^{n/2}$.*

Question *Écrire une fonction qui utilise de plus la remarque suivante : $n/2 = \begin{cases} n/2 & \text{si } n \text{ est pair} \\ n/2+1 & \text{sinon} \end{cases}$.*

Question *Effectuer le nombre de multiplications effectuées dans les deux cas.*

Exercice 4

On démontre que sur l'ensemble $\mathbb{N} \times \mathbb{N}$ est dénombrable en numérotant chaque couple $(x, y) \in \mathbb{N}^2$ suivant le procédé suggéré par la figure ci-dessous.



Question *Rédiger une fonction récursive qui retourne le numéro du point de coordonnées (x, y).*

Question *Rédiger la fonction réciproque, là encore de façon récursive.*

Exercice 5

On suppose donné un tableau $t[0,..,n-1]$ (contenant au moins trois éléments) qui possède la propriété suivante : $t_0 \geq t_1$ et $t_{n-2} \leq t_{n-1}$. Soit $k \in [|1, n-2|]$; on dit que t_k est un minimum local lorsque $t_k \leq t_{k-1}$ et $t_k \leq t_{k+1}$.

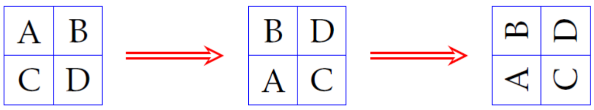
Question *Justifier l'existence d'un minimum local dans t.*

Question *Il est facile de déterminer un minimum local en coût linéaire : il suffit de procéder à un parcours de tableau. Pourriez-vous trouver un algorithme récursif qui en trouve un en réduisant le coût logarithmique ?*

Exercice 6

Les processeurs graphiques possèdent en général une fonction de bas niveau appelée *blit* (ou transfert de bloc) qui copie rapidement un bloc rectangulaire d'une image d'un endroit à un autre.

L'objectif de cet exercice est de faire tourner une image carrée de $n \times n$ pixels de 90°dans le sens direct en adoptant une stratégie récursive : découper l'image en quatre blocs de tailles $n/2 \times n/2$, déplacer chacun des ces blocs à sa position finale à l'aide de 5 *blits*, puis faire tourner récursivement chacun de ces blocs.



On supposera dans tout l'exercice que n est une puissance de 2.

Question *Exprimer en fonction de n le nombre de fois que la fonction blit est utilisée.*

Question *Quel est le coût total de cet algorithme lorsque le coût d'un blit d'un bloc $k \times k$ est en $\mathcal{O}(n^2)$?*

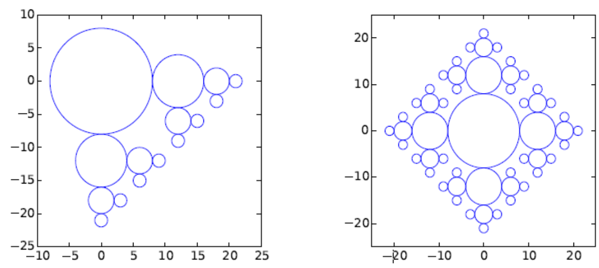
Question *Et lorsque ce coût est en $\mathcal{O}(n)$?*

Question *En supposant qu'une image est représentée par une matrice numpy $n \times n$, rédiger une fonction qui adopte cette démarche pour effectuer une rotation de 90°dans le sens direct (on simulera un blit par la copie d'une partie de la matrice vers une autre en décrivant ces parties par le slicing).*

Exercice 7

On suppose disposer d'une fonction *circle([x, y], r)* qui trace à l'écran un cercle de centre $(x; y)$ de rayon r.

Question *Définir deux fonctions récursives permettant de tracer les dessins présentés figure suivante (chaque cercle est de rayon moitié moindre qu'à la génération précédente).*



Question On suppose disposer d'une fonction `polygon((xa, ya), (xb, yb), (xc, yc))` qui trace le triangle plein dont les sommets ont pour coordonnées $(xa; ya)$, $(xb; yb)$, $(xc; yc)$.

Question *Définir une fonction récursive permettant le tracé présenté figure suivante (tous les triangles sont équilatéraux).*