Examen de seconde session

Durée : 2 heures

Partie Matlab

Seize jetons numérotés de 1 à 16 sont disposés aléatoirement sur une grille carrée, face visible. Par exemple :

4	14	1	7
13	8	12	16
3	9	6	15
2	5	11	10

On se donne : un numéro de jeton `num`,
un nombre de jetons `nb`,

tous deux compris entre 1 et 16, et l'on doit ramasser au plus `nb` jetons contigus, en partant du jeton numéro `num` et en progressant à chaque fois d'une case horizontalement ou verticalement. Quelle somme maximale peut-on réaliser ainsi ?

Par exemple, dans la disposition ci-dessus, avec `num` $= 14$, `nb` $= 5$, la somme maximale est 65, obtenue en ramassant successivement les jetons n° 14, 8, 12, 16 et 15.

On voudrait écrire en Matlab un programme qui crée et affiche le tableau des jetons, permet d'introduire au clavier les valeurs de `num` *et* `nb`*, puis calcule et affiche la somme maximale correspondante.*

1. Taper l'instruction suivante et décrire ce qu'elle fait :

```
>> T = reshape(randperm(16),[4,4])
```

2. Taper l'instruction suivante et dire à quoi elle sert :

```
>> [i,j] = find(T==12)
```

3. Ecrire une fonction `voisins(num,T)` renvoyant dans un vecteur les numéros des jetons voisins de `num` dans T (dans les directions horizontales et verticales). Par exemple, pour le tableau T ci-dessus, `voisins(9,T)` renvoie `[6,8,3,5]`, `voisins(2,T)` renvoie `[5,3]` (l'ordre des voisins n'a pas d'importance).

4. On veut écrire une fonction récursive `somme_max(num,nb,T)` qui donne la somme maximale qu'on peut obtenir en ramassant au plus `nb` jetons contigus dans T, à partir du jeton numéro `num`.
   a) Que doit renvoyer la fonction lorsque `nb` $= 1$?
   b) Pour `nb` > 1, comment obtenir la valeur de `somme_max(num,nb,T)` à partir des valeurs de `somme_max(numvoisin,nb-1,T)`, où `numvoisin` décrit l'ensemble des jetons voisins du jeton `num` ?
   c) En déduire la fonction `somme_max`.

TSVP

Partie Maple

Exercice 1

On a un certain nombre de cailloux disposés en un ou plusieurs tas. Partant de cette disposition, on prend un caillou dans chacun des tas et, avec les cailloux ainsi prélevés, on forme un nouveau tas ; cela donne une nouvelle disposition. On recommence à partir de cette disposition, ainsi de suite ...

1. On modélise une disposition de cailloux par une liste d'entiers non nuls représentant le nombre de cailloux de chaque tas. Ecrire une procédure `suivante(L)` qui, à partir d'une disposition L, renvoie la disposition suivante. Exemple d'exécution :

```
> suivante([6,4,1,7]);
                        [5,3,6,4]
```

2. Programmer le calcul des dispositions successives obtenues à partir d'une disposition initiale choisie. Que constate-t-on si le nombre de cailloux est un nombre triangulaire, c'est-à-dire de la forme $\frac{k(k+1)}{2}$?

Exercice 2 (*approximation d'intégrale*)

1. Soit P_5 le $5^{\text{ème}}$ *polynôme de Legendre* (noté `P(5,x)` dans la librairie `orthopoly`). Donner l'expression de P_5.

2. Soient $z_1, \ldots, z_5$ les zéros de P_5. Pour $i = 1$ à 5, on pose :

$$w_i = \frac{2}{(1 - z_i^2) P_5'(z_i)^2} \qquad (P_5' = \text{dérivée de } P_5).$$

Pour toute fonction $f : [-1, 1] \to \mathbf{R}$, on pose : $J(f) = \sum_{i=1}^{5} w_i f(z_i)$. Programmer la fonction J.

3. Comparer numériquement $J(f)$ et $\int_{-1}^{1} f(x)\, dx$ pour $f(x) = \mathrm{e}^{x \cos x}$.

4. Soit $P(x) = \sum_{i=0}^{10} a_i x^i$ un polynôme arbitraire de degré ≤ 10. Que vaut la différence $J(P) - \int_{-1}^{1} P(x)\, dx$?