

TP MAPLE N°4 : autour des séquences, listes et ensembles

1 Cours : trois nouveaux types de variable

1.1 Séquence

Une séquence est un type particulier que peut manipuler Maple : c'est une suite ordonnée d'objets séparés par une virgule.

Par exemple, `s := a, 2, genou, ln(34), entorse` est une séquence de 5 éléments.

On accède par exemple à son 3ième élément en tapant `s[3]`. On peut aussi en extraire une sous-séquence en tapant `s[2..4]`.

L'instruction `seq` permet de construire des séquences de nombres. Par exemple `t := seq(2*k+1, k=0..5)` renvoie la séquence des impairs de 1 à 11.

On peut «concaténer» deux séquences pour en fabriquer une troisième. Par exemple, `a := s, t` ou `b := t, s`. On remarque donc que la concaténation n'est pas commutative. Il y a un élément neutre, la séquence vide notée `NULL`.

On peut fabriquer des séquences par itération : par exemple la suite d'instructions suivante renvoie la séquence des diviseurs du nombre 100.

```
s:=NULL;
for k from 1 to 100 do
  if irem(100,k)=0 then
    s:=s,k;
  fi;
od;
```

1.2 Liste

La liste est un type de variable proche de celui de séquence. C'est une séquence comprise entre deux crochets.

Par exemple, `L := [bidule, sqrt(2), truc, saucisson]` est une liste. Si `s` est une séquence, alors `[s]` est une liste.

On peut en extraire ses éléments avec `op`. Ainsi `op(L)` est une séquence.

On peut connaître son nombre d'éléments avec `nops`. Ainsi `nops(L)` renvoie 4.

On peut extraire son deuxième élément avec `L[2]` et une sous-liste par exemple `L[2..3]`.

On peut rajouter un terme à une liste. Par exemple `m := [op(L), ail]`. Plus généralement, on peut concaténer deux listes : on les transforme en séquences, puis on les concatène comme deux séquences et enfin on retransforme la séquence obtenue en liste. Exemple : si `L1 := [1, 2, 3, 4]` et `L2 := [aa, bb, cc]`, `L3 := [op(L1), op(L2)]`.

La liste vide se note `V := []`. Ainsi `nops(V)` renvoie 0.

On peut transformer des éléments de la liste avec `subsop`. Par exemple `subsop(2=truc, L)` remplace le 2ième élément de la liste `L` par l'élément `truc`. En particulier `subsop(3=NULL, L)` efface le troisième élément de `L`.

On peut aussi faire des opérations coordonnées par coordonnées sur des listes de même taille. Exemples `L1+L2` ou `3*L1`. Enfin on peut aussi appliquer une fonction aux éléments d'une liste : `map(x->6*x+3, L1)`.

On peut extraire des éléments d'une séquence à l'aide de `select`. Exemple : `select(x->x>0, [-5, 2, -4, 3])` renvoie les éléments strictement positifs de `[-5, 2, -4, 3]`, c'est à dire `[2, 3]`. Le $x \rightarrow$ signifie « x tel que ».

1.3 Ensemble

Un ensemble est un autre type de variable qui correspond à la notion mathématique. Les éléments d'un **ensemble** ne sont pas ordonnés. Tous les éléments d'un ensemble sont différents. Un ensemble est une séquence entre deux accolades. `E := {1, 2, truc}`.

Remarquer que `{1, 2, 2, truc} = {1, 2, truc}` puisque les éléments d'un ensemble sont distincts.

L'ensemble vide est notée `vide := {}`

On retrouve les opérations classiques, intersection (`intersect`), réunion (`union`), différence (`minus`).

Maple utilise selon les cas, l'une ou l'autre de ces trois notions qui ne sont pas interchangeables.

Par exemple, la commande `solve` renvoie une séquence de racines. En revanche lorsque on résout un système, on doit rentrer un ensemble d'équations et un ensemble d'inconnues. C'est pourquoi il est important de consulter l'aide de Maple.

2 Exercices

Exercice 1 On considère les deux séquences suivantes : `s := seq (seq(2*k+1,k=1..3), j=1..2)` et `t := seq ({seq(2*k+1,k=1..3)}, j=1..2)`. Quel est le nombre d'éléments de ces deux séquences ? Donner votre réponse par réflexion puis vérifier avec maple.

Exercice 2 (Entiers somme de deux carrés)

1. Écrire la *liste* des nombres qui sont la somme de deux carrés d'entiers a^2 et b^2 avec $0 \leq a, b \leq 5$. Attention aux répétitions, déterminer leur nombre, essayer de donner cette liste dans un ordre croissant.
2. Sélectionner dans la liste précédente les nombres impairs. Quelle semble être leur forme générale ?

Exercice 3 (Nombres premiers)

1. Écrire la *liste* des nombres des nombres premiers qui sont inférieurs à 10000. Quel est leur nombre ? On utilisera la fonction `isprime`
2. Quel est le 943-ième nombre premier ?

Exercice 4 (Graphe d'une suite) Soit (u_n) la suite définie par $u_{n+1} = 3.5u_n(1 - u_n)$ et $u_0 = 0.4$.

1. Créer à l'aide d'une itération, la séquence `s` dont les éléments sont $[k, u_k]$ pour k variant de 0 à 100.
2. Représenter graphiquement la liste `[s]` de points avec la fonction `plot` et l'option `style=point`.

Exercice 5 (change l'ordre) Écrire une procédure qui inverse l'ordre des éléments d'une liste.

Exercice 6 (maximum) Écrire une procédure qui renvoie le plus grand élément d'une liste.

Exercice 7 (Tri) Écrire une procédure qui trie par ordre croissant les éléments d'une liste.

Remarques :

- il existe déjà une telle fonction en Maple nommée `sort` que nous n'utiliserons pas bien sûr pour cet exercice.
- il existe beaucoup d'algorithmes de tris plus ou moins performants : par exemple le tri par sélection, le tri par insertion, le «quicksort» qui illustre une méthode de programmation appelée «diviser pour régner».

Exercice 8 (Parties d'un ensemble) On pose $A = \{1, 2, 3, 4\}$ et B est l'ensemble des cubes inférieurs à 100 d'entiers positifs.

1. Déterminer $A \cup B$ (`union`), $A \cap B$ (`intersect`), $A \cap \{6, 8\}$ et $A \setminus B$ (`minus`).
2. Déterminer avec la fonction `powerset` (charger le package `combinat`) les sous-ensembles de $A = \{1, 2, 3, 4\}$ puis $\mathcal{P}(\emptyset)$ et enfin $\mathcal{P}(\mathcal{P}(\emptyset))$.
3. Écrire toutes les permutations de l'ensemble `{ton, esprit, est troublé}`.

Exercice 9 On considère l'équation différentielle (E) $xy'(x) + (1 - x)y(x) = \frac{xe^x}{x^2 + 1}$.

1. Résoudre (E). Sur quels intervalles avez-vous résolu l'équation différentielle ?
2. Taper `y(2)`, `y(x)`. Que se passe-t-il ? Extraire l'expression de la solution donnée à l'aide de la fonction `op` ou de la fonction `rhs` (right hand side) . Définir alors une fonction f solution de (E) (elle dépend encore d'une constante) à l'aide de la fonction `unapply`.
3. Représenter sur un même graphique plusieurs solutions de (E). On pourra créer une liste de fonctions.
4. Existe-t-il des solutions définies sur \mathbb{R} tout entier ? On pourra écrire un développement limité des solutions avec la fonction `series`.

Exercice 10 (Méthode d'Euler)