Université Rennes 1 Année 2013–2014
Département de Mathématiques
M1, module ALGB

# TP 1

Utilisation de Sage pour le calcul formel

A la fin du TP, il faudra rendre votre copie :
**enregistrez une dernière fois votre feuille de travail dans un fichier nommé**

TP1_votrenom_votreprenom.sagews

**et déposez ce fichier dans le cours ALGB de votre ENT sous** mes cours en ligne.

***Un seul dépôt possible au plus tard 15 min après la fin du TP. Vous ne pourrez pas déposer un deuxième fichier.***

Si vous n'avez pas terminé le TP pendant la séance, vous devez le finir en autonomie mais sans dépôt possible.
Rappel : pour sauvegarder et récupérer un fichier dans Sage, il faut

(1) Cliquer sur le bouton vert "Save".
(2) Cliquer sur Files (en haut à gauche).
(3) Placez la souris sur le fichier correspondant à votre feuille de travail et cliquez sur "Download file".

1. Utilisation des variables symboliques.

(1) Construire l'expression symbolique $P = (u^2+1)(u^2-2)$.
(2) Développer P.
(3) Évaluer P en 2.
(4) Définir et afficher l'expression symbolique $\frac{x^2+2x+1}{x+1}$.
(5) Simplifier l'expression précédente.
(6) Même question avec $\sqrt{x^2}$.

2. Premières fonctions.

(1) Écrire une procédure récursive `fact_rec` permettant de calculer $n!$.
(2) Écrire une version itérative `fact_it`.
(3) A l'aide de la commande `%timeit`, comparer les temps de calcul pour 50! et 500!. Comparer également avec la fonction `factorial` de `sage`.

**3.** Utilisation des listes.

(1) Créer la liste $L$ des entiers de 1 à 10.
(2) Construire la fonction $f : x \mapsto 2x + 1$.
(3) En utilisant la commande `map`, construire la liste $K$ des 10 premiers entiers positifs impairs.
(4) En utilisant une boucle `for`, calculer le produit scalaire de $L$ et $K$.
(5) Écrire une fonction `prodscal` qui calcule le produit scalaire de 2 listes. Cette fonction devra être commentée et traiter les cas pathologiques (listes de tailles différentes ou vides en entrée).
(6) Construire la liste des nombres premiers inférieurs à 100:
   (a) avec une boucle `for` et la commande `is_prime`,
   (b) avec la commande `filter`,
   (c) avec une boucle `while` et la commande `next_prime`.
(7) En déduire le produit des nombres premiers inférieurs à 100.

**4.** Manipulation de polynômes.

(1) Définir $P = x^4 + 3x^2 + 2$ comme un polynôme symbolique et le factoriser.
(2) Définir `QX` l'anneau des polynômes en la variable $X$ définis sur $\mathbb{Q}$.
(3) Définir $P$ sur `QX` et le factoriser.
(4) Définir $P$ comme un polynôme à coefficients dans le corps fini $\mathbb{F}_{43} (= \mathbb{Z}/43\mathbb{Z})$ et le factoriser.
(5) Définir $P$ comme un polynôme à coefficients dans le corps fini $\mathbb{F}_{47}$ et le factoriser.
(6) Définir $P$ comme un polynôme à coefficients dans le corps fini $\mathbb{F}_{41}$ et le factoriser.
(7) Écrire une fonction qui place les coefficients d'un polynôme dans une liste.

(8) Écrire une fonction qui calcule le polynôme dérivé d'un polynôme. La tester sur $P$.
(9) Définir `ZT` l'anneau des polynômes en la variable $T$ définis sur $\mathbb{Z}$.
(10) Calculer le pgcd de $Q = T^3 - 31T - 30 \in \mathbb{Z}[T]$ et de sa dérivé. Que pouvez vous en dire sur la multiplicité des racines de $Q$.
(11) "Envoyer" $Q$ dans $\mathbb{F}_{11}[T]$ et retester.

Exercice facultatif :

On s'intéresse à la suite de Syracuse, définie par

$$u_{n+1} = \begin{cases} \frac{u_n}{2} & \text{si } n \text{ pair} \\ 3u_n + 1 & \text{si } n \text{ impair} \end{cases}$$

(1) Écrire une fonction qui, étant donné $u_n$, calcule $u_{n+1}$.
(2) Construire une liste comprenant les 20 premiers termes de la suite pour $u_0 = 17$
(3) Il est conjecturé que quelque soit $u_0 > 0$, la suite $u_n$ finit toujours par rentrer dans un cycle 1,4,2. Écrire une fonction qui vérifie l'exactitude de cette conjecture et qui retourne le plus petit indice pour lequel la suite vaut 1.
(4) Sachant que la commande `ZZ.random_element(1000)` fournit un entier (`ZZ`) aléatoire entre 0 et 999, calculer l'indice moyen pour lequel la conjecture est vérifiée pour 100 valeurs de $u_0$ tirées aléatoirement entre 1 et 1000.