Outils Mathématiques pour l'Informatique
TP 2 : Arithmétique et Théorie des Nombres

Licence 3 Informatique

Maxime Bros, Christophe Clavier
Xlim, Université de Limoges

Novembre 2019

Pour l'exerice 1, et seulement pour l'exerice 1, vous aurez besoin d'une fonction classique en Théorie des Nombres : la factorisation. Étant donné qu'il est compliqué de coder une fonction de factorisation efficace [1], nous allons utiliser Sagemath qui est un langage construit sur Python et qui contient une telle fonction.

Pour utiliser Sagemath, vous pouvez vous rendre à l'adresse

`https://sagecell.sagemath.org`

et coder directement sur cette page, il suffit ensuite d'évaluer votre code en cliquant sur `Evaluate`.

1 L'indicatrice d'Euler

Vous avez vu en cours la fonction indicatrice d'Euler, notée $\Phi(n)$.

Écrivez une fonction `phi(n)` qui prend en paramètre un entier $n \geq 1$ et qui retourne la valeur $\Phi(n)$.

Remarque : vous aurez besoin de la fonction `factor(n)` de Sagemath qui renvoie un tableau de tableaux [2]. Cette fonction est très simple à comprendre, essayez là sur un petit entier, par exemple 18, pour comprendre exactement ce qu'elle renvoie.

Question 1 : Une fois cette fonction codée, appelez l'enseignant de TP pour qu'il vous donne une valeur à tester.

1. En effet, il s'agit d'un des sujets de projet proposés en Master 2.
2. En Python on parle plutôt de *listes*.

2 L'inverse modulaire

Question 1 : Coder une fonction[3] `EuclideEtendu(a,b)` qui, pour deux entiers a, b supérieurs ou égaux à 1, renvoie des valeurs (u, v) telles que

$$ua + vb = pgcd(a, b).$$

Question 2 : En utilisant la fonction précédente, coder une fonction `inverseMod(a,n)` qui renvoie la valeur $a^{-1} \mod n$ si elle existe, et 0 sinon. Une fois cette fonction codée, appelez l'enseignant de TP pour qu'il vous donne une valeur à tester.

3 Le groupe multiplicatif $\mathbb{Z}_n^*$

Question 1 : Coder une fonction `pgcd(a,b)` qui, pour deux entiers a, b supérieurs ou égaux à 1, renvoie le pgcd de a et b.

Question 2 : Coder une fonction, qui pour un entier $n \leq 50$, affiche la liste de tous les entiers (entre 0 et $n - 1$) qui appartiennent à $\mathbb{Z}_n^*$. Vous afficherez aussi le cardinal de ce groupe, c'est à dire le nombre d'éléments qu'il y a dedans.

Question 3 : Quel lien voyez vous entre le nombre d'éléments dans $\mathbb{Z}_n^*$ et la fonction calculée à l'exercice 1 ?

Question 4 : Compléter la fonction de la question 2 en demandant ensuite à l'utilisateur de choisir un des nombres appartenant au groupe $\mathbb{Z}_n^*$, appelons ce nombre a. Le but de cette question est d'afficher toutes les puissances successives de a dans $\mathbb{Z}_n^*$, c'est-à-dire

$$a^1 \mod n, \quad a^2 \mod n, \quad a^3 \mod n, \quad \ldots$$

L'affichage s'arrête lorsque vous trouvez la plus petite valeur entière k telle que $a^k \mod n$ vaille 1, vous avez alors trouvé *l'ordre multiplicatif de a modulo n*.

Une fois cette fonction codée, appelez l'enseignant de TP pour qu'il vous donne une valeur à tester.

Question 5 : Pour certains groupes $\mathbb{Z}_n^*$, vous arriverez à trouver des entiers a tels que toutes leurs puissances sont en fait le groupe multiplicatif en entier, on parle de racines primitives modulo n.

Donnez quelques exemples de racines primititives modulo un entier $n \leq 50$ de votre choix.

3. L'algorithme a été donné en cours.

Bonus Pour d'autres groupes, vous n'arriverez pas à trouver de racines primitives, donnez des exemples.

Énoncez les critères qui permettent de déterminer à l'avance, si pour une valeur de n fixée, le groupe multiplicatif $\mathbb{Z}_n^*$ contient une telle racine ou non. Comment appelle-t-on un tel groupe ?