

Projet 2

1 Défi mathématique

1.1 Le carré parfait

On dit qu'un entier positif $x(x \in \mathbb{N})$ est un carré parfait si $\sqrt{x} \in \mathbb{N}$.

Pour un ensemble d'entiers positifs $\{a, a+1, a+2, \dots, b\}$, soit $C(a, b)$ le nombre de sous-ensembles non vides dans lesquels le produit de tous les éléments est un carré parfait.

Par exemple $C(5, 10) = 3$, puisque les produits de tous les éléments de $\{5, 8, 10\}$, $\{5, 8, 9, 10\}$ et $\{9\}$ sont des carrés parfaits, et aucun autre sous-ensemble de $\{5, 6, 7, 8, 9, 10\}$ n'a cette propriété.

1. Vérifier que $C(40, 55) = 15$, et $C(1000, 1234) \bmod 1000000007 = 975523611$.
2. Trouver $C(1000000, 1234567) \bmod 1000000007$.

1.2 Exponentiation efficace

La manière la plus naïve de calculer n^{15} nécessite quatorze multiplications :

$$n \times n \times \dots \times n = n^{15}$$

Mais en utilisant une méthode "binaire" vous pouvez le calculer en six multiplications :

$$\begin{aligned} n \times n &= n^2 \\ n^2 \times n^2 &= n^4 \\ n^4 \times n^4 &= n^8 \\ n^8 \times n^4 &= n^{12} \\ n^{12} \times n^2 &= n^{14} \\ n^{14} \times n &= n^{15} \end{aligned}$$

Cependant il est encore possible de le calculer en seulement cinq multiplications :

$$\begin{aligned} n \times n &= n^2 \\ n^2 \times n &= n^3 \\ n^3 \times n^3 &= n^6 \\ n^6 \times n^6 &= n^{12} \\ n^{12} \times n^3 &= n^{15} \end{aligned}$$

Nous définirons $\text{minProd}(k)$ comme le nombre minimum de multiplications pour calculer n^k ; par exemple $\text{minProd}(15) = 5$.

Pour $1 \leq k \leq 200$, trouvez

$$\sum_{k=1}^{200} \text{minProd}(k)$$

.

2 Jeu de Nim

2.1 Principe du jeu

Ce jeu se joue à deux. On dispose d'un certain nombre de tas (entre 3 et 7) de pierres empilées avec des hauteurs différentes. Par exemple, considérons quatre tas de 3, 8, 7 et 11 pierres par tas. Chaque joueur, à tour de rôle, prend le nombre de pierres qu'il veut (au moins une) dans un des tas. Le dernier joueur qui prend une pierre a perdu. Donc, l'idée du jeu est de forcer l'adversaire à prendre la dernière pierre.

2.2 Déroulement du jeu

2.2.1 Lancement du jeu

Au départ, l'ordinateur demande aux deux joueurs d'introduire leurs noms. Si l'un des deux joueurs a déjà joué au jeu, l'ordinateur doit récupérer le score de la dernière partie jouée ainsi que le meilleur score obtenu par le joueur depuis un fichier de sauvegarde. Dans le cas où il s'agit d'un nouveau joueur, un score et un meilleur score de valeurs nulles lui seront attribués.

L'ordinateur doit maintenant tirer aléatoirement un chiffre entre 3 et 7 représentant le nombre de tas de pierres. Après, il tire autant de chiffre entre 5 et 23 que de tas de pierres. Ces chiffres représentent le nombre de pierres par tas.

L'ordinateur affiche les noms de deux joueurs ainsi que l'état du jeu i.e. les tas et les pierres sont affichés comme suit :

Par exemple, on a 4 tas avec (5, 9, 19, 5) pierres par tas, on doit avoir comme affichage :

```
1| ***** |5
2| ********* |9
3| ***************** |19
4| ***** |5
```

2.2.2 Tour du jeu

L'ordinateur demande à un des deux joueurs de commencer et de jouer le premier tour.

Le joueur doit indiquer le tas et le nombre de pierres à retirer. Pour le faire, il doit entrer une chaîne de caractères de la forme "Numéro du tas" - "Nombre de pierres à retirer". Par exemple, si le joueur introduit : 3-13, cela veut dire que l'ordinateur doit retirer 13 pierres du tas numéro 3.

Après validation du coup l'ordinateur affiche l'état du jeu. Si on reprend l'exemple précédent, l'état du jeu est :

```
1| ***** |5
2| ********* |9
3| ***** |6
4| ***** |5
```

L'ordinateur affiche le nom du joueur suivant et l'invite à jouer son coup pour continuer.

2.2.3 Fin du jeu

Le jeu se termine s'il ne reste qu'une pierre à retirer d'un tas. Le joueur qui doit jouer à cette étape a perdu, et son score est nul. Pour le joueur gagnant, son score dépend du nombre de coups joués pour gagner. Soit nbCoup le nombre de coups joués, le score est calculé avec la formule suivante :

$$score = \sum_{i=1}^{nbCoup} i * (10)^i$$

Si le nouveau score obtenu est plus petit que le meilleur score obtenu par le joueur dans les parties précédentes, la valeur du meilleur score est écrasée par la valeur du nouveau score. La valeur du score de la dernière partie est également mise à jour. Ces informations sont enregistrées localement dans un fichier text ou un fichier pickle.

L'ordinateur affiche à la fin, la liste des noms et scores des 10 meilleurs joueurs et propose une nouvelle partie ou quitter le jeu.

3 Instructions

- Les étudiants doivent répondre au défi mathématique dans une feuille de calcul jupyter.
- Il sera tenu compte de la rédaction de la feuille calcul.
- Il est important de renseigner la feuille de calcul en précisant les noms, prénoms et matricules des étudiants participants à la feuille.
- Le jeu Nim doit être déposé sur un compte github de l'un des étudiants du même groupe.
- L'étudiant en question doit initialiser un dépôt github et inviter ses trois autres camarades à contribuer au projet.
- Le fichier README.md doit comporter toutes les informations relatives aux étudiants intervenants.

A propos du projet

Imad Eddine BOUSBAA (ibousbaa@usthb.dz), année 2018 ;
Ce projet *est* sous licence GNU GENERAL PUBLIC LICENSE.