Prologin 2016

Concours national d'informatique

Épreuve écrite d'algorithmique
Louvain-La-Neuve
Strasbourg

Samedi 20 Février 2016

Let's Go to the finale !

1 Préambule

Bienvenue à **Prologin**. Ce sujet est l'épreuve écrite d'algorithmique et constitue la première des trois parties de votre épreuve régionale. Sa durée est de 3 heures. Par la suite, vous passerez un entretien (20 minutes) et une épreuve de programmation sur machine (4 heures).

Conseils

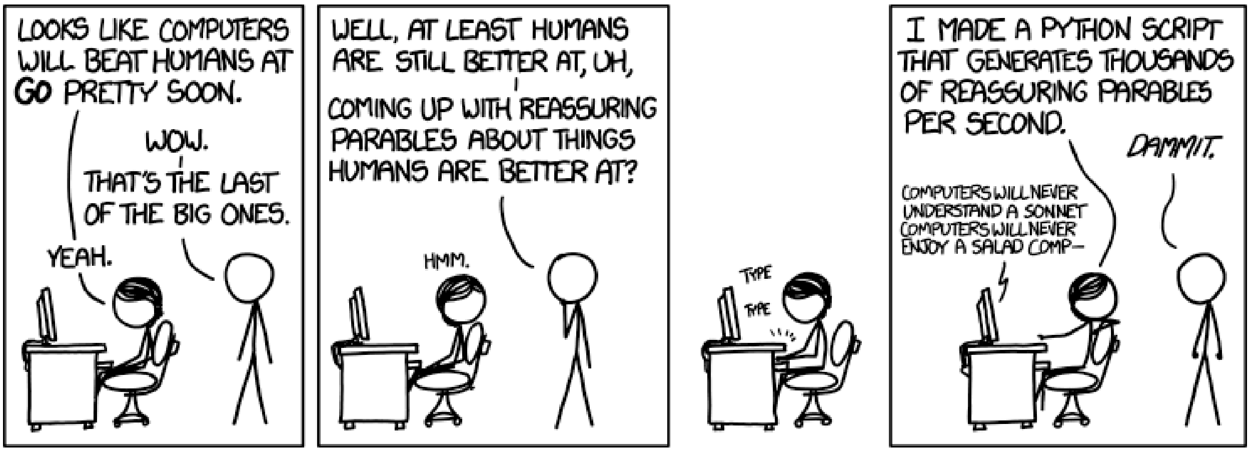
— Lisez bien tout le sujet avant de commencer.
— **Soignez la présentation** de votre copie.
— N'hésitez pas à poser des questions.
— Si vous avez fini en avance, relisez bien.
— N'oubliez pas de passer une bonne journée.

Remarques

— Le barème est donné à titre indicatif uniquement.
— Indiquez lisiblement vos nom et prénom, la ville où vous passez l'épreuve et la date en haut de votre copie.
— Tous les langages sont autorisés, veuillez néanmoins préciser celui que vous utilisez.
— Ce sont des humains qui lisent vos copies : laissez une marge, aérez votre code, ajoutez des commentaires (**seulement** lorsqu'ils sont nécessaires) et évitez au maximum les fautes d'orthographe.
— Le barème récompense les algorithmes les plus efficaces : écrivez des fonctions qui trouvent la solution le plus rapidement possible.
— Si vous trouvez le sujet trop simple, relisez-le, réfléchissez bien, puis dites-le-nous, nous pouvons ajouter des questions plus difficiles.

2 Sujet

Introduction



Du 5 au 9 octobre 2015, AlphaGo a battu Fan Hui, champion d'Europe de Go [1]. Hikaru et Akira, deux jeunes japonais passionés de Go, se rendant bien compte que le Go, c'est nul [2], décident alors de tester d'autres jeux [3] pour en trouver un intéressant.

Dans tout le reste du sujet, Akira commencera toujours. Ils jouent à tour de rôle et le perdant sera toujours celui qui n'a plus de mouvement disponible.

Pour les six premières questions, et uniquement celles-ci, l'efficacité algorithmique de votre réponse ne sera pas prise en compte, uniquement sa correction. Un résumé de chaque jeu est disponible à la fin. Sauf mention contraire, on supposera que les deux protagonistes jouent parfaitement.

Partie 1

Tout d'abord, ils décident d'essayer le jeu de vim [5] : partant d'un tas de n pierres de Go, ils en retirent un nombre strictement positif de leur choix à tour de rôle (le perdant est donc celui qui doit jouer alors qu'il n'a plus de pierre devant lui, puisqu'il faut toujours en retirer au moins une).

Exemple

Supposons que l'on commence avec un tas de 13 pierres :
— Akira commence en enlevant 5 pierres.
— Il reste 8 pierres. Hikaru enlève 3 pierres.
— Il reste 4 pierres. Akira enlève 4 pierres.
— Il reste 0 pierre. Hikaru ne peut pas jouer : Il a perdu. Tristesse !

1. Au Go. Parce que s'il l'avait battu au tennis, on n'en aurait pas parlé
2. Alors qu'en fait, il n'y a pas besoin de ça pour s'en rendre compte
3. D'ailleurs vous avez perdu [4]
4. Vous allez beaucoup perdre aujourd'hui
5. Trois organisateurs ont été sacrifiés pour cette blague. Et ça va continuer

Question 1 (1 point)

> Comment Akira peut-il jouer pour être sûr de gagner ?

Question 2 (1 point)

> Comme nos deux protagonistes sont des abrutis, ils préfèrent en retirer une seule à tour de rôle. Écrire une fonction qui prend n en entrée, et renvoie le nom du gagnant.

Question 3 (3 points)

> Comme nos deux protagonistes sont encore des abrutis, Akira en retire toujours a, et Hikaru toujours h. Écrire une fonction qui prend n, a et h en entrée, et renvoie le nom du gagnant.

Partie 2

Malheureusement, Akira gagne toujours. Hikaru préfère donc jouer au jeu de min : c'est exactement le même que le jeu de vim, si ce n'est que chaque protagoniste a le droit d'enlever au plus 5 pierres par tour.

Hikaru applique toujours **aveuglément** la stratégie suivante :
- Soit a le nombre de pierres qu'a enlevées Akira au tour précédent (Akira commençant, c'est toujours bien défini).
- Hikaru en enlève $6 - a$.

Exemple

Supposons que l'on commence avec un tas de 12 pierres :
- Akira commence en enlevant 5 pierres.
- Il reste 7 pierres. Hikaru enlève 1 pierre.
- Il reste 6 pierres. Akira enlève 2 pierres.
- Il reste 4 pierres. Hikaru enlève 4 pierres.
- Il reste 0 pierre. Akira ne peut pas jouer : Il a perdu. Honte à lui !

Question 4 (3 points)

> Si $n = 42$, alors Hikaru applique cette stratégie. Montrer qu'il gagne toujours.

Question 5 (3 points)

> Montrer que si $n = 10$ et si Hikaru applique cette stratégie, Akira peut gagner. Trouver une autre valeur de n pour laquelle la stratégie ne fonctionne pas non plus.

Question 6 (5 points)

> Décrire un algorithme qui prend comme arguments n et le nombre de pierres maximal que l'on peut enlever à chaque tour, et renvoie le nom du gagnant.

Partie 3

Malheureusement, ces jeux[6] sont un peu trop simples. Hikaru et Akira aimeraient essayer des jeux plus dynamiques, comme par exemple le jeu de max[7].

Les règles du jeu de max sont simplissimes : à chaque tour, chaque protagoniste peut soit enlever une seule pierre, soit en retirer la moitié arrondie au supérieur.

Exemple

Supposons que l'on commence avec un tas de 15 pierres :
- Akira commence en enlevant 8 pierres.
- Il reste 7 pierres. Hikaru enlève 1 pierre.
- Il reste 6 pierres. Akira enlève 3 pierres.
- Il reste 3 pierres. Hikaru enlève 1 pierre.
- Il reste 2 pierres. Akira enlève 1 pierre.
- Il reste 1 pierre. Hikaru enlève 1 pierre.
- Il reste 0 pierre. Akira ne peut pas jouer : Il a perdu. Bien fait !

Question 7 (5 points)

> Trouver un algorithme qui prend en entrée le nombre initial n de pierres, et renvoie le nom du gagnant au jeu de max.

Le problème, c'est qu'Akira, qui est plus prompt à faire des erreurs de calcul que Hikaru, soudoie souvent les juges pour mettre peu de pierres initialement. Pour empécher ça, Hikaru a réussi à faire ajouter la règle suivante au jeu de max : on ne peut pas jouer de coup (et on perd donc) si il reste 42 pierres ou moins. On appelle ce nouveau jeu le jeu de minimax.

Question 8 (3 points)

> Modifier votre algorithme précédent pour tenir compte de ce changement de règle.

Akari, une amie d'Hikaru, veut parier sur le résultat de la partie. Malheureusement, elle vous écoute depuis le début, et sait donc facilement calculer qui va gagner. Pour lui permettre de parier, vous allez donc jouer au jeu de winamax, qui ressemble beaucoup au jeu de minimax, les deux actions possibles étant les suivantes :
- Si il reste strictement plus de 42 pierres, il peut en **rajouter** une
- Tant qu'il reste strictement plus de 42 pierres et qu'il y en a un nombre pair, il peut en enlever le moitié, et ceci autant de fois qu'il le désire à la suite (au moins une fois, et il faut à chaque fois que le nombre de pierres restantes soit pair)

On remarque que, comme au jeu de minimax, si un joueur commence son tour avec moins de 42 pierres restantes, il perd[8].

6. D'ailleurs, vous avez oublié de perdre
7. Ou jeu d'Emacs
8. Et vous aussi

Exemple

Supposons que l'on commence avec un tas de 1337 pierres :

— Akira commence en ajoutant 1 pierre.
— Il reste 1338 pierres. Hikaru enlève 669 pierres.
— Il reste 669 pierres. Akira ajoute 1 pierre.
— Il reste 700 pierres. Hikaru enlève 350 pierre et rejoue.
— Il reste 350 pierres. Hikaru enlève 175 pierres.
— Il reste 175 pierres. Akira ajoute 1 pierre.
— Il reste 176 pierres. Hikaru enlève 88 pierres et rejoue.
— Il reste 88 pierres. Hikaru enlève 44 pierres et décide de ne pas rejouer.
— Il reste 44 pierres. Akira enlève 22 pierres.
— Il reste 22 pierres. Hikaru ne peut pas jouer : Il a perdu. Bien fait !

Question 9 (10 points)

> Écrire un algorithme qui, étant donné n, renvoie le nom du gagnant au jeu de winamax.

Partie 4

Akari s'étant enrichie ainsi, elle a assez d'argent pour acheter deux fois plus de pierres, et propose donc de jouer au jeu de mac [9]. Le principe du jeu de mac est le suivant : on dispose de deux jeux de max en parallèle, et à chaque tour le joueur en cours choisit un des deux jeux, et joue dessus (i.e. en retire une pierre ou en enlève la moitié arrondie au supérieur).

Exemple

Supposons que l'on commence avec deux tas : l'un de 15 pierres et l'autre de 13 pierres :

— Akira commence en choisissant le second tas, de 15 pierres, en enlevant 1 pierre.
— Il reste 14 pierres dans l'un, 13 dans l'autre. Hikaru choisit le premier et en enlève 7 pierres.
— Il reste 7 pierres dans l'un, 13 dans l'autre. Akira choisit le second et en enlève 1 pierre.
— Il reste 7 pierres dans l'un, 12 dans l'autre. Hikaru choisit le second et en enlève 6 pierres.
— Il reste 7 pierres dans l'un, 6 dans l'autre. Akira choisit le second et en enlève 3 pierres.
— Il reste 7 pierres dans l'un, 3 dans l'autre. Hikaru choisit le premier et en enlève 1 pierre.
— Il reste 6 pierres dans l'un, 3 dans l'autre. Akira choisit le premier et en enlève 3 pierres.
— Il reste 3 pierres dans l'un, 3 dans l'autre. Hikaru choisit le premier et en enlève 1 pierre.
— Il reste 2 pierres dans l'un, 3 dans l'autre. Akira choisit le premier et en enlève 1 pierre.
— Il reste 1 pierre dans l'un, 3 dans l'autre. Hikaru choisit le second et en enlève 1 pierre.
— Il reste 1 pierre dans l'un, 2 dans l'autre. Akira choisit le second et en enlève 1 pierre.
— Il reste 1 pierre dans l'un, 1 dans l'autre. Hikaru choisit le premier et en enlève 1 pierre.
— Il reste 0 pierre dans l'un, 1 dans l'autre. Akira choisit le second et en enlève 1 pierre.
— Il reste 0 pierre. Hikaru ne peut pas jouer : Il a perdu. Sûrement la faute du mac.

Question 10 (5 points)

> Ils jouent au jeu de mac avec deux jeux en parallèle, un à n pierres et l'autre à m pierres. Décrire un algorithme qui à n et m renvoie le nom du gagnant.

9. Elle n'a cependant pas les moyens de s'en acheter un pour de vrai

Question 11 (3 points)

> On peut s'intéresser au jeu de mac dans le cas particulier où il y a autant de pierres dans chacun des deux jeux. Écrire une fonction qui au nombre de pierres initial associe le nom du gagnant. Elle doit s'exécuter en temps constant.

Alors que le (jeu de) mac est lent et inintéressant, (le jeu de) tux est lui un jeu génial. Le principe du jeu de tux est le suivant :

- À chaque tour, le joueur en cours choisit un tas non vide, et joue sur celui-ci
- Jouer sur un tas consiste à effectuer l'une des deux actions suivantes :
  - Enlever une pierre de ce tas
  - Si il y a un nombre pair de pierres sur ce tas, le diviser par deux en le remplaçant par deux tas de tailles égales (i.e. remplacer un tas à $2n$ pierres par deux tas de n pierres).

Un joueur perd donc lorsque, au début de son tour, tous les tas sont vides.

Exemple

Supposons que l'on commence avec un tas de 6 pierres :

- Akira commence en le séparant en deux tas de 3 pierres.
- Il reste deux tas de 3 pierres. Hikaru enlève 1 pierre du premier.
- Il reste un tas de 3 pierres, et un autre de 2. Akira enlève 1 pierre au premier.
- Il reste deux tas de 2 pierres. Hikaru enlève 1 pierre du premier.
- Il reste un tas de 2 pierres, et un autre de 1 pierre. Hikaru enlève l'unique pierre du second.
- Il reste un tas de 2 pierres. Akira le sépare en deux tas de 1 pierres.
- Il reste deux tas de 1 pierre. Hikaru enlève l'unique pierre du second.
- Il reste un tas de 1 pierre. Akira le détruit.
- Il reste 0 pierre. Hikaru ne peut pas jouer : Il a perdu. Bouh !

Question 12 (10 points)

> Supposons qu'il n'y a initialement qu'un tas de n pierres. Écrire un algorithme qui prend en entrée n, et renvoie le nom du gagnant.

Plutôt qu'utiliser son argent pour acheter des pierres, Akari a en fait préféré acheter un générateur magique de pierres, pour jouer au jeu de TeX : C'est exactement le jeu de tux, si ce n'est que, cette fois-ci, un entier k est fixé au début de la partie, et quand un joueur est en présence d'un tas de taille n, il peut soit retirer une pierre, soit, si n est pair, faire apparaître k tas de taille $\frac{n}{2}$.

Supposons que l'on commence avec un tas de 4 pierres et que $k = 3$

— Akira commence en le séparant en trois tas de 2 pierres.
— Il reste trois tas de 2 pierres. Hikaru enlève 1 pierre du premier.
— Il reste deux tas de 2 pierres, et un autre de 1. Akira enlève 1 pierre à ce dernier.
— Il reste deux tas de 2 pierres. Hikaru sépare le premier en trois tas de 1 pierre.
— Il reste trois tas de 1 pierre, et un autre de 2 pierre. Hikaru sépare le dernier en trois tas de 1 pierre
— Il reste six tas de 1 pierre. Akira détruit le premier.
— Il reste cinq tas de 1 pierre. Hikaru détruit le premier.
— Il reste quatre tas de 1 pierre. Akira détruit le premier.
— Il reste trois tas de 1 pierre. Hikaru détruit le premier.
— Il reste deux tas de 1 pierre. Akira détruit le premier.
— Il reste un tas de 1 pierre. Hikaru détruit le premier.
— Il reste 0 pierre. Akira ne peut pas jouer : Il a perdu. Les règles sont trop compliquées ?

Question 13 (20 points)

> Je pense que vous avez compris la question.

Partie Bonus

Question 14 (5 points)

> Dans le jeu winamax de la question 9, que se passerait-il si l'on ne pouvait diviser le nombre de pierres par deux qu'au plus une fois par tour ?

À Gotopia, il existe n pays. Dans chaque pays i, il existe n_i clubs de Go, et x_i canards. Gotopia est tel que le premier club du i^{me} pays de Gotopia, noté $c_{i,1}$, possède $x_i + 1$ pierres de Go, le second, noté $c_{i,2}$, $x_i + 2$ pierres de Go, le j^{me}, noté $c_{i,j}$, possède $x_i + j$ pierres, et le dernier, noté c_{i,n_i}, possède $x_i + n_i$ pierres.

Hikaru et Akira, qui sont devenus, par votre aide, dictateurs de Gotopia [10], peuvent à chaque tour choisir un club, et retirer autant de pierres qu'il leur sied d'icelui.

Question 15 (50 points)

> Il ne peut rester qu'un dictateur à Gotopia. Lequel ?

Question 16 (-1 point)

> Quel est le meilleur jeu : le Go ou les Échecs ?

10. Et en ont profité pour vous faire pendre. Longue histoire

Résumé des différents jeux

— Akira commence toujours
— La partie commence avec un tas de n pierres
— Ils jouent à tour de rôle
— Le premier à ne pas pouvoir jouer perd

Jeu de vim

— À chaque tour, le joueur en cours retire un nombre de pierres de son choix

Jeu de min

— À chaque tour, le joueur en cours retire un nombre de pierres de son choix inférieur à k

Jeu de max

À chaque tour, le joueur en cours peut :
— retirer une pierre
— ou garder la moitié des pierres arrondie à l'inférieur

Jeu de minimax

À chaque tour, si il y a strictement plus de 42 pierres, le joueur en cours peut :
— retirer une pierre
— ou garder la moitié des pierres arrondi à l'inférieur

Jeu de winamax

À chaque tour, si il y a strictement plus de 42 pierres, le joueur en cours peut :
— ajouter une pierre
— ou, si le nombre de pierres est pair, enlever la moitié des pierres, et faire ceci continuellement tant que c'est possible et que le joueur le désire

Jeu de mac

Il y a deux tas.À chaque tour, le joueur en cours peut choisir un tas à $n > 0$ pierres et :
— retirer une pierre de ce tas
— ou, si le nombre de pierres de ce tas est pair, enlever $\frac{n}{2}$ pierres d'icelui

Jeu de tux

Il y a deux tas.À chaque tour, le joueur en cours peut choisir un tas à $n > 0$ pierres et :
— retirer une pierre de ce tas
— ou, si le nombre de pierres de ce tas est pair, et le séparer en deux tas de $\frac{n}{2}$ pierres

Jeu de TeX

Il y a deux tas.À chaque tour, le joueur en cours peut choisir un tas à $n > 0$ pierres et :
— retirer une pierre de ce tas
— ou, si le nombre de pierres de ce tas est pair, et le remplacer par k tas de $\frac{n}{2}$ pierres