

Le jeu de Juniper Green

Boris LAVAL et Olivier SICARD

IREM de la Réunion

3 Mai 2017

Aux cycles 3 et 4

- Multiples et diviseurs des nombres d'usage courant. Critères de divisibilité (2, 3, 4, 5, 9, 10).
- Déterminer si un entier est ou n'est pas multiple ou diviseur d'un autre entier.
Simplifier une fraction donnée pour la rendre irréductible.
Division euclidienne (quotient, reste).
Multiples et diviseurs.
Notion de nombres premiers.

Place de l'arithmétique dans les programmes

En terminale S

- Divisibilité et congruences dans \mathbb{Z} .
- Division euclidienne
- PGCD de deux entiers et entiers premiers entre eux.
- Théorème de Bézout et théorème de Gauss.
- Nombres premiers et existence et unicité de la décomposition en produit de facteurs premiers.

En Bts Sio

Division euclidienne : quotient, reste, existence, unicité.
Nombres premiers, décomposition en produit de produit de facteurs premiers et facteurs premiers, entiers premiers entre eux, PGCD Sujet d'examen

ORIGINES ET RÈGLES

Le jeu a été créé par Richard Porteous, enseignant à l'école de Juniper Green, auquel il doit son nom. A l'origine le jeu de Juniper green se joue à deux, avec les nombres entiers de 1 à 20.

ORIGINES ET RÈGLES

Le jeu a été créé par Richard Porteous, enseignant à l'école de Juniper Green, auquel il doit son nom. A l'origine le jeu de Juniper green se joue à deux, avec les nombres entiers de 1 à 20.

Les règles du jeu sont plutôt simples :

- Règle 1 : Le premier joueur choisi un nombre pair entre 1 et 20.

ORIGINES ET RÈGLES

Le jeu a été créé par Richard Porteous, enseignant à l'école de Juniper Green, auquel il doit son nom. A l'origine le jeu de Juniper green se joue à deux, avec les nombres entiers de 1 à 20.

Les règles du jeu sont plutôt simples :

- Règle 1 : Le premier joueur choisi un nombre pair entre 1 et 20.
- Règle 2 : Une fois qu'un nombre a été choisi il ne pourra plus jamais être utilisé.

ORIGINES ET RÈGLES

Le jeu a été créé par Richard Porteous, enseignant à l'école de Juniper Green, auquel il doit son nom. A l'origine le jeu de Juniper green se joue à deux, avec les nombres entiers de 1 à 20.

Les règles du jeu sont plutôt simples :

- Règle 1 : Le premier joueur choisi un nombre pair entre 1 et 20.
- Règle 2 : Une fois qu'un nombre a été choisi il ne pourra plus jamais être utilisé.
- Règle 3 : Chaque joueur doit choisir un nombre qui est soit un diviseur soit un multiple du nombre précédent.

ORIGINES ET RÈGLES

Le jeu a été créé par Richard Porteous, enseignant à l'école de Juniper Green, auquel il doit son nom. A l'origine le jeu de Juniper green se joue à deux, avec les nombres entiers de 1 à 20.

Les règles du jeu sont plutôt simples :

- Règle 1 : Le premier joueur choisi un nombre pair entre 1 et 20.
- Règle 2 : Une fois qu'un nombre a été choisi il ne pourra plus jamais être utilisé.
- Règle 3 : Chaque joueur doit choisir un nombre qui est soit un diviseur soit un multiple du nombre précédent.
- Règle 4 : Le joueur qui ne peut plus jouer perd la partie.

ORIGINES ET RÈGLES

Le jeu a été créé par Richard Porteous, enseignant à l'école de Juniper Green, auquel il doit son nom. A l'origine le jeu de Juniper green se joue à deux, avec les nombres entiers de 1 à 20.

Les règles du jeu sont plutôt simples :

- Règle 1 : Le premier joueur choisi un nombre pair entre 1 et 20.
- Règle 2 : Une fois qu'un nombre a été choisi il ne pourra plus jamais être utilisé.
- Règle 3 : Chaque joueur doit choisir un nombre qui est soit un diviseur soit un multiple du nombre précédent.
- Règle 4 : Le joueur qui ne peut plus jouer perd la partie.

Faisons une partie

RÉFLEXIONS MATHÉMATIQUES

LISTE DES COUPS POSSIBLES

coup précédent	coups possibles																			
1		2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
2	1			4		6		8		10		12		14		16		18		20
3	1					6			9			12			15			18		
4	1	2						8				12				16				20
5	1									10					15					
6	1	2	3									12						18		
7	1													14						
8	1	2		4												16				
9	1		3															18		
10	1	2			5															20
11	1																			
12	1	2	3	4		6														
13	1																			
14	1	2						7												
15	1		3		5															
16	1	2		4				8												
17	1																			
18	1	2	3			6			9											
19	1																			
20	1	2		4	5					10										

RÉFLEXIONS MATHÉMATIQUES

- Celui qui joue un nombre premier assez grand (11, 13, 17, 19) gagne la partie.

RÉFLEXIONS MATHÉMATIQUES

- Celui qui joue un nombre premier assez grand (11, 13, 17, 19) gagne la partie. C'est pour cela que l'on interdit les nombres impairs au premier coup !

RÉFLEXIONS MATHÉMATIQUES

- Celui qui joue un nombre premier assez grand (11, 13, 17, 19) gagne la partie. C'est pour cela que l'on interdit les nombres impairs au premier coup !
- Celui qui joue 1 perd, car alors le joueur d'après peut jouer ce qu'il veut et il se réfère à la ligne du dessus.

RÉFLEXIONS MATHÉMATIQUES

- Celui qui joue un nombre premier assez grand (11, 13, 17, 19) gagne la partie. C'est pour cela que l'on interdit les nombres impairs au premier coup !
- Celui qui joue 1 perd, car alors le joueur d'après peut jouer ce qu'il veut et il se réfère à la ligne du dessus.
- Celui qui joue 14 perd car l'autre joue 7 et oblige le premier à jouer 1.

UNE IA HUMAINE

```
def IA3(x):
    if x==1:
        res=11      # ici l'IA gagne
    elif x==14:
        res=7       # c'est une strategie gagnante
    elif len(coup_dispo[x])==1:
        res=coup_dispo[x][0]
    elif len(coup_dispo[x])==2:
        res=coup_dispo[x][1]
    elif len(coup_dispo[x])>2:
        l=len(coup_dispo[x])
        listres=[]
        for i in range(1):
            if coup_dispo[x][i]!=1:
                if coup_dispo[x][i]!=14: # interdit à l'IA de jouer 1 et 14
                    listres.append(coup_dispo[x][i])
        lmin=20
        for coup in listres:
            if len(coup_dispo[coup])<=lmin:
                lmin=len(coup_dispo[coup])
        listchoix=[]
        for coup in listres:
            if len(coup_dispo[coup])==lmin:      # l'IA choisi Le nombre qui offre le moins de choix possibles à son adversaire
                listchoix.append(coup)
        res=listchoix[floor(random()*len(listchoix))]
    else:
        res=0      # ici l'IA a perdu
    return res
```

Explorer toutes les voies et optimiser sa chance de gagner

```
joueur : 20 5 3 6
ordi : 10 15 18 12
vos coups possibles sont [1, 2, 4]
le joueur choisi un nombre?2
[20, 10, 5, 15, 3, 18, 6, 12, 2, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 4, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 4, 8, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 4, 8, 16, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 4, 16, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 4, 16, 8, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 8, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 8, 4, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 8, 16, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 8, 16, 4, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 14, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 14, 7, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 16, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 16, 4, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 16, 4, 8, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 16, 8, 1]
[20, 10, 5, 15, 3, 18, 6, 12, 2, 16, 8, 4, 1]
joueur : 20 5 3 6 2
ordi : 10 15 18 12 4
```

UNE IA INHUMAINE

Faire jouer 1 et choisir un grand nombre premier

```
Fonction IA():entier
  Si nombreDeCoupPossible()=0 alors
    retourner 0 # l'IA perd
  SinonSi nombreDeCoupPossible()=1 alors
    retourner le seul coup possible
  Sinon
    Si dernierCoup()=1 alors
      retourner le plus grand nombre premier
    Sinon
      profondeurMax=20
      retourner meilleurCoup(profondeurMax)
  FinSi
FinSi
```

UNE IA INHUMAINE

Une approche probabiliste

joueur : 20 5

ordi : 10 15

vos coups possibles sont [1, 3]

le joueur choisi un nombre?3

1 : 0.0

6 : 0.48333333333333334

9 : 0.4789915966386555

12 : 0.5163934426229508

18 : 0.5210084033613446

l'IA choisit 18

UNE IA INHUMAINE

Variable Globale

```
voie : liste d'entier # Par exemple [8,2,6,3,9,1]
voieExplo : liste de listes d'entiers
```

```
Fonction meilleurCoup(profondeurMax : entier):entier
  exploration(profondeurMax,0) # on actualise voieExplo
  déduire les frequences de victoire des coups possibles
  déterminer le coup retenu, coup de forte fréquence
  retourner coupRetenu
```

UNE IA Récursive

Variable Globale

voieExplo : liste de liste d'entier

Procédure exploration(profondeurMax,profondeur)

Si profondeur<profondeurMax alors

Si dernierCoup(voie)=1 alors

ajouter voie à voieExplo

Sinon

Pour coup parcourant les coups possibles

Ajouter coup à la voie

exploration(forceIA,profondeur+1)

enlever coup à la voie

FinPour

FinSi

FinSi

TAUX DE WIN DU JOUEUR 1

		JOUEUR 2				
taux de win du joueur 1 pour nbmax=20		IA0 - full random	IA1 - humaine	IA2 - semi expert humain	IA 3	IA4 inhumaine
JOUEUR 1	IA0	47,70%	47,70%	41,70%	52,60%	
	IA1	47,70%	44,60%	41,70%	52,40%	
	IA2	55,10%	55,10%	51,70%	61,40%	
	IA3	90,20%	90,20%	90,20%	100%	80%
	IA4				80%	66%

JOUER A JUNIPER GREEN TOUT SEUL

On peut jouer seul à juniper green.

JOUER A JUNIPER GREEN TOUT SEUL

On peut jouer seul à juniper green.

Le but est alors de faire la plus longue suite de Juniper possible.

Recherche d'un plus grand vol avec la récursivité

Duée expo:0.018 Nombre le plus grand:10 Taille max de la voie:9
Une voie possible [10, 5, 1, 9, 3, 6, 2, 8, 4]

Duée expo:0.021 Nombre le plus grand:11 Taille max de la voie:9
Une voie possible [10, 5, 1, 9, 3, 6, 2, 8, 4]

Duée expo:0.121 Nombre le plus grand:12 Taille max de la voie:11
Une voie possible [11, 1, 9, 3, 6, 12, 4, 8, 2, 10, 5]

Duée expo:0.130 Nombre le plus grand:13 Taille max de la voie:11
Une voie possible [13, 1, 9, 3, 6, 12, 4, 8, 2, 10, 5]

Duée expo:0.231 Nombre le plus grand:14 Taille max de la voie:12
Une voie possible [14, 7, 1, 9, 3, 6, 12, 4, 8, 2, 10, 5]

Duée expo:0.649 Nombre le plus grand:15 Taille max de la voie:13
Une voie possible [15, 5, 10, 2, 14, 7, 1, 9, 3, 6, 12, 4, 8]

Duée expo:2.474 Nombre le plus grand:16 Taille max de la voie:14
Une voie possible [16, 8, 4, 12, 6, 3, 15, 5, 10, 2, 14, 7, 1, 13]

Duée expo:1.831 Nombre le plus grand:17 Taille max de la voie:14
Une voie possible [17, 1, 16, 8, 4, 12, 6, 3, 15, 5, 10, 2, 14, 7]

Duée expo:7.776 Nombre le plus grand:18 Taille max de la voie:16
Une voie possible [17, 1, 16, 8, 4, 12, 6, 18, 9, 3, 15, 5, 10, 2, 14, 7]

Duée expo:8.094 Nombre le plus grand:19 Taille max de la voie:16
Une voie possible [19, 1, 16, 8, 4, 12, 6, 18, 9, 3, 15, 5, 10, 2, 14, 7]

Duée expo:37.719 Nombre le plus grand:20 Taille max de la voie : 17
Une voie possible [20, 10, 5, 15, 3, 9, 18, 6, 12, 4, 16, 8, 2, 14, 7, 1, 19]

- M. Raphaël Frontera : activité de calcul au collège
frontera.fr/spip.php?article162
- Pour jouer seul :
troumad.org/Math/juniper_green.php
- Le matou matheux
matoumatheux.ac-rennes.fr/tous/jeuxnombre/jeuGreen.htm
- Olivier Lebourg :
www.apmep.fr/Le-jeu-de-Juniper-Green
- Didier Müller : Juniper et Python
www.nymphomath.ch/pj/juniper/chapitre5.pdf