JEU DE DRIBBLE BINAIRE

Jean-Claude FONTAINE

1. PROPOSITION DE JEU

Il existe sur la disquette 5 IPT de Pâques 1985, le programme DRIBBLE. L'écran figure une partie de football. Un joueur est en possession du ballon représenté par un 0 (zéro). Pour marquer le but, il doit atteindre un nombre tiré aléatoirement par l'ordinateur et qui se trouve dans la cage du gardien. Pour cela, il doit seulement ajouter 1 ou multiplier par 2 le nombre précédemment obtenu.

Après un certain nombre de parties, j'ai proposé aux élèves d'essayer de trouver une stratégie permettant de gagner en un nombre minimum de coups. Pour commencer, nous avons simulé des exemples au tableau.

1.1. Exemple : Atteindre 100

```
0
+1=1
*2=2
*2=4
*2=8
*2=16
*2=32
*2=64
+1=65
+1=66
.....
.....
+1=99
+1=100
```

Dans cet exemple, il faut 42 coups pour atteindre 100 !!!

Problème : Essayer d'atteindre 100 en utilisant moins de coups que dans cet exemple.

1.2. Étude des solutions proposées

Nous avons comparé les différentes solutions proposées par les élèves. Nous avons obtenu une grande variété de réponses. Les élèves donnaient le nombre de coups nécessaires pour atteindre 100. Comme certains d'entre-eux sont arrivés très vite à des solutions en 13, voire 10 coups, les autres ont essayé d'améliorer leur performance. Une gamine a trouvé en 9 coups, et à partir de ce moment là, ce fut l'objectif à atteindre.

Analyse rapide de la solution la plus performante :

0,+1,*2,+1,*2,*2,*2,+1,*2,*2=100 (neuf coups).

2. AUTRES RECHERCHES

2.1. Propositions de recherche

127	128
0	0
+1=1	+1=1
*2=2	*2=2
+1=3	*2=4
*2=6	*2=8
+1=7	*2=16
*2=14	*2=32
+1=15	*2=64
*2=30	*2=128
+1=31	Solution en 8 coups.
*2=62	
+1=63	
*2=126	
+1=127	

Solution en 13 coups.

Question : Pourquoi faut-il moins de coups pour atteindre 128 que 127 ?

Les élèves constatent que pour atteindre 128 nous avons toujours multiplié par deux, ce qui est la façon la plus rapide d'atteindre un nombre dans le jeu de DRIBBLE.

128=2*2*2*2*2*2*2 ou 2^2

128 est une puissance de 2.

Nous en avons déduit que les puissances de deux seraient les nombres les plus vite atteints :

$4=2^2$, en trois coups,
$8=2^3$, en quatre coups,
$16=2^4$, en cinq coups,
$32=2^5$, en six coups,
$64=2^6$, en sept coups,
$128=2^7$, en huit coups,
$256=2^8$, en neuf coups,
$512=2^9$, en dix coups,
$1024=2^{10}$, en onze coups.

Nous avons demandé aux élèves d'apprendre rapidement cette suite de nombres car elle est très utilisée en informatique.

3. ALGORITHME DE RÉSOLUTION

3.1. Analyse des exemples

-Libres remarques des élèves.

A ce moment du travail, les élèves étaient d'accord pour dire qu'il fallait partir du nombre à atteindre pour trouver, à l'envers, la suite des opérations à effectuer.

Quelles sont les données du problème ?

- le nombre de départ :0

- le nombre à atteindre tiré au hasard par l'ordinateur.

- la possibilité d'appliquer *2 ou +1 au nombre précédent.

Quel est le moyen le plus rapide d'atteindre la cible ?

- en multipliant toujours par 2.

Si nous multiplions toujours par deux sans tenir compte du nombre à atteindre, nous risquons de nous trouver dans une situation impossible. Laquelle ?

- le nombre obtenu sera plus grand que le nombre à atteindre.

A quel nombre faut-il donc arriver pour que, en le multipliant par deux, on ne dépasse pas le nombre à atteindre ?

- s'il est pair : nombre à atteindre divisé par deux (ex :128/2=64).
- s'il est impair : (nombre à atteindre - 1) puis divisé par deux (ex :(127-1)/2=63).

Il nous faut maintenant atteindre ce nombre intermédiaire. Quelle stratégie allons-nous employer ?

- c'est la même façon d'agir : 64/2=32 et (63-1)/2=31.

- et ainsi de suite jusqu'à ce que nous obtenions zéro.

3.2. Ecriture de l'algorithme

Situation de départ :

Données : Nombre de départ 0

Nombre à atteindre : NA

Contraintes : Opérations autorisées :+1 ou *2

Atteindre NA en un minimum d'opérations.

Situation finale : Obtenir la suite des résultats intermédiaires pour gagner à tous les coups au jeu de DRIBBLE.

Première version de l'algorithme

- lire NA
- s'il est pair alors le diviser par deux.
- s'il est impair alors enlever 1.
- le nombre ainsi obtenu est le nouveau nombre à atteindre.
- afficher NA
- recommencer tant que NA est différent de zéro.

Deuxième version de l'algorithme

- lire NA
- si NA est pair alors NA reçoit NA/2
  sinon NA reçoit NA-1
- afficher NA
- recommencer tant que NA est différent de zéro.

Troisième version de l'algorithme

Comment la machine va-t-elle reconnaître qu'un nombre est pair ?

```
DEBUT
   |  LIRE NA
   |  TANT QUE NA <> 0 FAIRE
   |      DEBUT
   |         |  SI NA EST TERMINE PAR 0, 2,4,6,8
   |         |                        ALORS NA=NA/2
   |         |                        SINON NA=NA-1
   |         |  AFFICHER NA
   |       FIN
   |  AFFICHER 0
FIN
```

Cet algorithme nous a semblé difficile à programmer en logo.

Nous avons cherché s'il n'y avait pas une autre façon pour reconnaître si NA était pair.

Quatrième version de l'algorithme

```
DEBUT
   |   LIRE NA
   |   TANT QUE NA <> 0 FAIRE
   |       DEBUT
   |          | Q=ENT(NA/2)
   |          | R=NA-(Q*2)
   |          | SI R=0
   |          |    ALORS NA=NA/2
   |          |    SINON NA=NA-1
   |          | AFFICHER NA
   |        FIN
   |   AFFICHER 0
FIN
```

Notre idée fut alors de programmer l'ordinateur afin qu'il fasse tous les calculs intermédiaires, ce qui nous assurerait la victoire à chaque partie.

4.PROGRAMMATION LOGO

Première version

```
POUR DRIBBLE :NA
SI EGAL? :NA 0 [STOP]
DONNE "Q ENT(:NA / 2)
DONNE "R :NA - (:Q * 2)
SI EGAL? :R 0 [DONNE "NA :NA / 2] [DONNE "NA :NA - 1]
EC :NA
DRIBBLE :NA
FIN
```

Ce programme affiche les résultats intermédiaires en partant du nombre à atteindre. Il faut donc lire de bas en haut sur l'écran. Pour obtenir une écriture de haut en bas, il faut que l'ordinateur empile les résultats intermédiaires puis les affiche en commençant par le sommet de la pile.

Deuxième version

```
POUR DRIBBLE :NA
SI EGAL? :NA 0 [STOP]
DONNE "Q ENT(:NA / 2)
DONNE "R :NA - (:Q * 2)
SI EGAL? :R 0 [DONNE "NA :NA / 2] [DONNE "NA :NA - 1]
DRIBBLE :NA
EC :NA
FIN
```

La conclusion de ce travail est que l'intérêt pour le jeu de DRIBBLE a fortement diminué. Heureusement qu'il y avait l'option ternaire, et nous sommes repartis pour un nouvel algorithme.

J.-C. Fontaine

Instituteur animateur
en informatique dans
les Bouches du Rhône.