IREM 2011-2012
Résolution de problèmes
Etienne Mann
etienne.mann@univ-montp2.fr

Histoire d'oeufs

Je tiens tout d'abord à vous remercier d'avoir posé ce problème à vos classes. Félicitez vos élèves pour leur recherche et donnez-leur rendez-vous l'année prochaine! J'étais très content d'entendre que certains élèves ont très bien accroché au problème et qu'ils se sont passionnés pour les œufs allant jusqu'à envahir les discussions familiales et à discuter (voire parfois débattre vivement) avec leurs copains pour les convaincre de leurs idées.

Le problème : Monsieur Paul Hayet fabrique des œufs en céramique qui sont tous identiques. Il voudrait tester la solidité des œufs. Pour cela, il dispose d'une échelle de 100 barreaux. Pour tester la résistance d'un œuf, il le laisse tomber de la hauteur d'un barreau et il regarde s'il s'est cassé ou non. Il voudrait déterminer le barreau le plus haut où les œufs ne se cassent pas. Quelle est la meilleure stratégie pour faire le moins de tests possibles ?

La relance : En général, vos premières stratégies minimisent le nombre de tests mais elles utilisent beaucoup d'œufs. Par soucis d'économie, Paul Hayet ne veut utiliser que **2 œufs**. Un de ses collaborateurs a trouvé une **stratégie en au plus** 19 tests. Je sais qu'il existe une meilleure stratégie, pouvez-vous la trouver ?

1. Solution pour une infinité d'œufs

En faisant une dichotomie, nous arrivons à faire 7 essais.

Théorème 1.1. — *Cette stratégie minimise le nombre de tests.*

Preuve. — A chaque test, nous codons 0 si l'œuf est cassé et 1 sinon. Ainsi après n tests nous avons récupéré une suite 0 et de 1 de longueur n c'est-à-dire un nombre en binaire de taille n.
Dans le problème, nous voulons pouvoir identifier 101 niveaux (de 0 à 100) avec une stratégie. Pour cela, nous disposons de 2^n nombres binaires. Avoir une stratégie en n tests signifie qu'au pire nous sommes capable d'identifier le barreau en n coups c'est-à-dire de relier n'importe quel nombre binaire de taille n à un barreau ; plus précisément, de définir une application surjective de $\varphi : \{0,1\}^n \to \{0,\dots,100\}$. La surjectivité implique que le cardinal de départ est plus grand que 101. Le plus petit n possible est 7 car $2^7 = 128$. Ainsi, une stratégie doit avoir au moins 7 tests. Remarquons que cette même stratégie marche aussi pour une échelle ayant entre 64 et 127 barreaux. □

Remarque 1.2. — Si nous appliquons la dichotomie, le codage proposé dans la démonstration ci-dessus correspond à l'écriture binaire de numéro du barreau. Plus précisément

$$\begin{aligned} \varphi : \{0,1\}^7 &\longrightarrow \{0,\dots,127\} \\ 0\,000\,000 &\longmapsto 0 \\ 0\,000\,001 &\longmapsto 1 \\ 1\,111\,111 &\longmapsto 127 \end{aligned}$$

2. Solution pour 2 œufs

D'abord nous remarquons qu'une stratégie en 19 tests se fait en plaçant par exemple les œufs tous les 10 barreaux (11, 12 et 13 marchent aussi) s'il ne cassent pas. Le cas le pire est donc le barreau 99. Nous faisons 10 tests $(10, 20, \dots, 100)$ et au 100-ième il casse. Nous redescendons alors au 91 puis 92 jusqu'au 99.
Heuristiquement, certains groupes ont remarqué que pour améliorer cette stratégie il fallait sauter plus de barreaux au début et moins vers la fin c'est-à-dire qu'il faudrait modifier le pas à chaque test...c'était la bonne intuition.

Théorème 2.1. — *Il existe une stratégie en* 14 *tests et elle minimise le nombre de tests.*

Preuve:. — La preuve que nous présentons est assez rigolote car nous trouvons une stratégie et nous démontrons dans le même temps qu'elle est minimale.
Pour fixer les idées, remarquons le fait suivant.
Fait: *Si l'œuf ne casse pas au barreau p et casse au barreau $p+k$. On doit placer le* 2*-ième œuf au barreau $p+1$, puis $p+2$ puis le monter* 1 *en* 1*. Au final on aura fait au pire $k-1$ tests avec le* 2*-ième œuf.*
Notons m le nombre minimal de tests. Nous ne le connaissons pas à priori mais nous verrons comment le déterminer dans la suite.

1. Comme nous avons m tests, nous pouvons toujours mettre le première œuf sur le m-ième barreau. S'il casse, nous le mettons au 1-er barreau puis 2-ième barreau....jusqu'au $m-1$-ième (nous appliquons le fait avec $p=0$ et $k=m$). Dans le cas le pire (c'est-à-dire s'il ne casse pas au 1-er, 2-ième, $\dots$,au $m-1$-ième) nous aurons fait m tests.
2. S'il ne casse pas, nous n'aurons plus que $(m-1)$ essais car nous n'a déjà fait 1 test au m-ième barreau. Du coup, nous le mettons au $m+(m-1)$-ième barreau. S'il casse nous appliquons le fait ci-dessus (avec $p=m$ et $k=m-1$) et nous aurons fait au pire m tests.
3. S'il ne casse pas, nous n'avons plus que $m-2$ tests, du coup nous le mettons au $m+(m-1)+(m-2)$-ième barreau. S'il casse, nous appliquons le fait ci-dessus (avec $p=m+m-1$ et $k=m-2$) et nous aurons fait au pire m test.

Nous continuons comme ça.
Au final, l'algorithme ci-dessus converge c'est-à-dire que nous pouvons tester tous les barreaux si m vérifie $m+(m-1)+\dots+1 \geq 100$. Le plus petit entier m qui le vérifie est $m=14$. □