OLYMPIADES MATHÉMATIQUES LYCÉE, PREMIÈRE

ACADÉMIE DE CRÉTEIL 2021

CORRIGÉ DE L'ÉPREUVE

21e OLYMPIADES DE MATHÉMATIQUES

MINISTÈRE DE L'ÉDUCATION NATIONALE

Sujet et Corrigé vous sont présentés par freemaths.fr . . .

CORRECTION OLYMPIADES

EXERCICES ACADÉMIQUES

CRETEIL 2021

Exercice 1 : L'escargot quadrille …

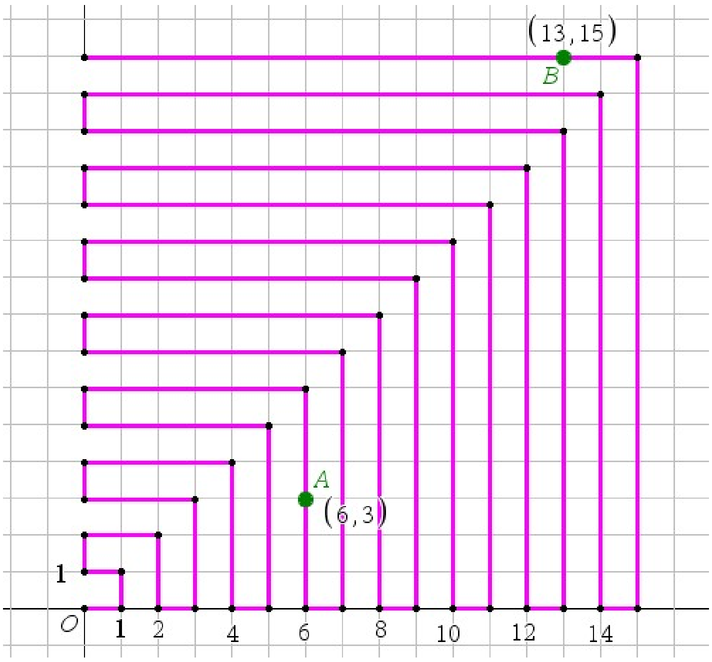
Partie I : Exploration du quadrillage

1. Voici dessinée la trajectoire jusqu'au point de coordonnées $(0\,;15)$.

On peut supposer le plan orienté suivant les quatre points cardinaux. Examinons, pour p entier naturel, la « potence » allant du point de coordonnées $(2p\ ;\ 0)$ au point de coordonnées $(2p+2\ ;\ 0)$. Elle est composée de :

- Un pas vers l'Est (admettons une fois pour toutes qu'un escargot soit capable de « faire un pas » …)
- $(2p+1)$ pas vers le Nord
- $(2p+1)$ pas vers l'Ouest
- Un pas vers le Nord
- $(2p+2)$ pas vers l'Est
- $(2p+2)$ pas vers le Sud

Soit en tout $8p+8$ pas pour parcourir la boucle complète.



Le point de coordonnées $(6\ ;\ 3)$ est situé, sur ce parcours, trois pas avant la fin de la troisième potence.

Or : $D(6\,;0)=8+16+24=48$. Donc : $D(6\,;3)=48-3=45$

La distance $D(6\,;3)$ est égale à 45

2. Pour parvenir au point de coordonnées $(13\ ;\ 15)$, il faut parcourir sept potences pour atteindre le point de coordonnées $(14\ ;\ 0)$ et faire 18 pas de plus :

Or : $D(14;\ 0)=8+16+24+32+40+48+56=8\times(1+2+3+...+7)=8\times 28=224$

On en déduit : $D(13\ ;\ 15)=224+18=242$

La distance $D(13\,;15)$ est égale à 242

3.a. Le point situé à la distance 50 est le deuxième point de la quatrième potence, c'est donc le point de coordonnées $(7\,;1)$.

3.b. On observe que : $D(8\ ;\ 0)=80$ et que $D(10\ ;\ 0)=120$:
Après une distance de 100 unités, l'escargot atteindra le point situé à « mi-potence », soit le point de coordonnées $(0\ ;\ 10)$.

Partie II : Quelques propriétés de la distance parcourue

1.a. On trouve : $D(0\ ;\ 2)=4$; $D(0\ ;\ 4)=16$; $D(0\ ;\ 6)=36$, ce qui amène à la conjecture de l'énoncé :

$$D(0\,;2p)=4p^2$$

NB. Bien que cela ne soit pas demandé par l'énoncé, on peut justifier cette propriété par un *raisonnement par récurrence*.
Les résultats ci-dessus montrent que la propriété « $D(0\,;2p)=4p^2$ » est vérifiée au rang 1 (elle est *initialisée*). Nous n'aurons plus besoin des vérifications suivantes, une suffit.

Observons maintenant pour p entier strictement positif la potence du point de coordonnées $(0\ ;\ 2p)$ au point de coordonnées $(0\ ;\ 2p+2)$. Elle est composée de :

$2p$ pas vers l'Est	+	$2p$ pas vers le Sud	+	Un pas vers l'Est
$(2p+1)$ pas vers le Nord	+	$(2p+1)$ pas vers l'Ouest	+	Un pas vers le Nord

Soit en tout $(8p+4)$ pas pour parcourir à partir de $(0\ ;\ 2p)$ la potence complémentaire.
Par conséquent : $D(0\,;2p+2)=D(0\,;2p)+(8p+4)$

Supposons que la propriété $D(0\ ;\ 2p)=4\,p^2$ soit vérifiée pour un certain rang p entier strictement positif.
Alors : $D(0\ ;\ 2p+2)=4\,p^2+(8p+4)=4(p+1)^2$, la propriété en jeu est vérifiée au rang suivant $(p+1)$. Cette propriété est *héréditaire*.

On peut conclure que, les deux hypothèses d'initialisation et d'hérédité étant vérifiées, la relation « $D(0\,;2p)=4p^2$ » admise par l'énoncé est justifiée pour tout entier p strictement positif.

1.b. Pour continuer à partir du point de coordonnées $(0\,;2p)$, l'escargot effectue $2p$ pas vers l'Est puis $2p$ pas vers le Sud. Il atteint alors le point de coordonnées $(2p\,;0)$ et un pas de plus vers l'Est l'amène au but, au point de coordonnées $(2p+1\,;0)$. On en déduit que : $D(0\,;2p+1)=D(0\,;2p)+4p+1=4p^2+4p+1$.
On reconnaît là le développement d'un carré, celui de $(2p+1)^2$.

Par conséquent : $D(0\,;2p+1)=(2p+1)^2$ **.**

La relation $D(0\,;2p+2)=4(p+1)^2$, quant à elle, a déjà été vue.

2. On remarque que 2021 est un entier voisin d'un carré parfait : en effet $45^2 = 2025$ et ainsi : $2021 = 45^2 - 4$.

L'escargot sera arrivé au point de coordonnées $(45\,;\,0)$ quatre pas plus loin, lorsqu'il aura parcouru une distance égale à 2025 unités.

Il est quatre pas avant ce point (il faut ainsi « reculer » d'un pas vers l'Ouest puis de trois pas vers le Sud).

L'escargot est au point de coordonnées $(44\,;\,3)$ lorsqu'il a parcouru une distance égale à 2021 unités.

Partie III : Points sur une droite

1. Observons ...

Point	A_0	A_1	A_2	A_3	A_4	A_5	A_6	A_7
Dist.	**0**	**5**	**18**	**39**	**68**	**105**	**150**	**203**
saut		s_0	s_1	s_2	s_3	s_4	s_5	s_6
		5	**13**	**21**	**29**	**37**	**45**	**53**
Ecart			$s_1 - s_0$	$s_2 - s_1$	$s_3 - s_2$	$s_4 - s_3$	$s_5 - s_4$	$s_6 - s_5$
			8	**8**	**8**	**8**	**8**	**8**

Conjecture apparente : Pour tout entier naturel *n*, $s_{n+1} - s_n = 8$

2. Le « saut » s_n représente la longueur du parcours allant du point de coordonnées $(n\,;\,2n)$ au point de coordonnées $(n+1\,;\,2n+2)$. Il s'agit d'une « potence » en forme de « gamma », semblable à celles déjà évoquées. On note qu'une potence se déduit de la précédente en ajoutant deux pas à chacune de ses quatre branches. Son parcours est donc plus long de huit pas que le parcours de sa précédente.

Ce qui démontre la conjecture : $s_{n+1} - s_n = 8$

NB. La suite (s_n) est donc une suite arithmétique, de premier terme 5 et de raison 8. On peut en déduire, ce que l'énoncé ne demande pas de faire, que pour tout entier naturel *n*, $s_n = 5 + 8n$

Détaillons d'ailleurs la composition du parcours de la *n*-ième potence :

- n pas vers l'Est, pour atteindre le point de coordonnées $(2n\,;\,2n)$
- $2n$ pas vers le Sud pour atteindre le point de coordonnées $(2n\,;\,0)$
- Un pas vers l'Est pour atteindre le point de coordonnées $(2n+1\,;\,0)$
- $(2n+1)$ pas vers le Nord pour atteindre le point de coordonnées $(2n+1\,;\,2n+1)$
- $(2n+1)$ pas vers l'Ouest pour atteindre le point de coordonnées $(0\,;\,2p+1)$
- Un pas vers le Nord pour atteindre le point de coordonnées $(0\,;\,2n+2)$
- $(n+1)$ pas vers l'Ouest pour atteindre le point cible

En tout, on obtient bien : $n + 2n + 1 + (2n+1) + (2n+1) + 1 + (n+1) = 8n + 5$ pas.

Partie IV : Oh ! Des fractions ...

Les parties précédentes ont mis en évidence trois choses :

- L'escargot visite une fois et une seule chaque point à coordonnées entières naturelles du plan.
- L'application D associe à chaque point à coordonnées entières naturelles du plan un nombre entier, sa « distance à l'origine selon l'escargot ».
- Inversement, la donnée d'un entier naturel n détermine un point et un seul à coordonnées entières naturelles $(x\ ;\ y)$ dont la « distance à l'origine selon l'escargot » est égale à n.

Par construction, l'application $(x\ ;\ y) \in \mathrm{N} \times \mathrm{N} \xrightarrow{D} D(x\ ;\ y) \in N$ est ainsi une bijection de l'ensemble $\mathbb{N}\times\mathbb{N}$ sur $\mathbb{N}$.

L'application réciproque D^{-1} qui, à chaque entier n associe le point de coordonnées $(x\ ;\ y)$ dont la « distance à l'origine selon l'escargot » est égale à n permet de numéroter l'ensemble des couples de nombres entiers positifs ou nuls.

On peut dès lors effectuer la construction suivante :

$$n \in N \xrightarrow{D^{-1}} D^{-1}(n) = (x\ ;\ y) \in \mathrm{N} \times \mathrm{N} \xrightarrow{T} (x\ ;\ y+1) \in \mathrm{N} \times \mathrm{N}^*$$

Cette fois, on numérote non plus $\mathbb{N}\times\mathbb{N}$ mais $\mathbb{N}\times\mathbb{N}^*$.
Le premier nombre x est positif ou nul alors que le deuxième est toujours strictement positif, il peut servir le cas échéant de dénominateur.

En associant à $(x\ ;\ y+1) \in \mathrm{N} \times \mathrm{N}^*$ la fraction $\dfrac{x}{y+1}$, on numérote les fractions.

Inconvénient : les fractions $\dfrac{1}{2}\ ;\dfrac{2}{4}\ ;\dfrac{3}{6}\ ;...$ numérotées différemment représentent un même nombre rationnel, de même que, en général, toutes les fractions équivalentes, numérotées différemment, représentent un même nombre rationnel.

Il reste à faire la convention suivante : Si les deux entiers x et $y+1$ sont *premiers entre eux,* alors on tient compte de la numérotation que l'on affecte au nombre rationnel $\dfrac{x}{y+1}$. Sinon, on ne tient pas compte de la numérotation, que l'on reporte sur le couple suivant.

Ainsi, seules les fractions irréductibles seront numérotées (on ne numérote pas les fractions qui lui sont équivalentes, représentant le même nombre rationnel). Il s'en déduit une numérotation de tous les rationnels positifs.

Complément (hors énoncé) : le coin Python

Ci-dessous, le programme « **cargol** » est affecté de deux arguments *a* et *b*. Il est destiné à calculer la distance à parcourir pour atteindre le point de coordonnées $(a\,;b)$. L'instruction facultative « print(**z**) » fait afficher toutes les positions occupées par l'escargot.
Cette instruction peut être effacée pour des parcours longs (on ne garde que « **return d** »).

```
>>> def cargol(a,b):
        x=1
        y=0
        z=[x,y]
        d=1
        while z!=[a,b]:
                if x==y:
                        if x%2==0:
                                y=y-1
                        else:
                                x=x-1
                elif x>y:
                        if y>0:
                                y=y-(-1)**x
                        else:
                                if x%2==0:
                                        x=x+1
                                else:
                                        y=1
                else:
                        if x>0:
                                x=x+(-1)**y
                        else:
                                if y%2==0:
                                        x=1
                                else:
                                        y=y+1
                z=[x,y]
                d=d+1
                print(z)
        return d
```

Voici un exemple d'exécution de **cargol** :

```
>>> cargol(3,2)
[1, 1]
[0, 1]
[0, 2]
[1, 2]
[2, 2]
[2, 1]
[2, 0]
[3, 0]
[3, 1]
[3, 2]
11
```

Variante du précédent, le programme « **cargoldist** » ci-dessous est affecté d'un argument *n*. Il est destiné à déterminer quel est le point où se trouve l'escargot lorsqu'il a parcouru une distance égale à *n*.

On note que dans les deux programmes, l'escargot part du point $(1\,;0)$ et non de l'origine, c'est pourquoi la distance **d** est initialisée à 1. Ceci pour des raisons de compatibilité avec les conditions « **if** ».

```
>>> def cargoldist(n):
        x=1
        y=0
        z=[x,y]
        d=1
        while d<n:
                if x==y:
                        if x%2==0:
                                y=y-1
                        else:
                                x=x-1
                elif x>y:
                        if y>0:
                                y=y-(-1)**x
                        else:
                                if x%2==0:
                                        x=x+1
                                else:
                                        y=1
                else:
                        if x>0:
                                x=x+(-1)**y
                        else:
                                if y%2==0:
                                        x=1
                                else:
                                        y=y+1
                z=[x,y]
                d=d+1
        return z
```

On reconnaît en exécutant **cargoldist** quelques-uns des résultats obtenus au fil du problème :

```
>>> cargoldist(11)
[3, 2]
>>> cargoldist(50)
[7, 1]
>>> cargoldist(100)
[0, 10]
>>> cargoldist(242)
[13, 15]
>>> cargoldist(2021)
[44, 3]
```

La difficulté est d'écrire des instructions conditionnelles permettant de « guider l'escargot » correctement d'un point au suivant. La liste d'instructions choisie ici est probablement améliorable.

Exercice 2 : Les jeux de Nim

Partie I : Avec des cailloux, le dernier perd ...

1.a. Si $r = 1$, Amélie est obligée de prélever le dernier caillou, elle perd.

1.b. **Il y a une stratégie gagnante dans chacun des cas :**

- Si $r = 2$, Amélie doit prélever un caillou pour gagner.
- Si $r = 3$, Amélie doit prélever deux cailloux pour gagner.
- Si $r = 4$, Amélie doit prélever trois cailloux pour gagner.

1.c. Amélie est obligée de prélever 1, 2 ou 3 cailloux. S'il reste 5 cailloux au moment où elle joue, Amélie est obligée de laisser à Balthasar 4, 3 ou 2 cailloux, tous cas de figure du **1.b** où le joueur qui a la main gagne en adoptant la « meilleure stratégie possible ».

Amélie ne peut pas gagner.

2.a. **Les nombres perdants sont 1, 5, 9, 13 et 17.**

En effet, s'il reste 17 cailloux, quel que soit le nombre de cailloux prélevés, le joueur qui a la main va laisser 14, 15 ou 16 cailloux. Son adversaire lui en laissera 13. Au tour suivant, le joueur laissera 10, 11 ou 12 cailloux, son adversaire lui en laissera 9. Au tour suivant, le joueur laissera 6, 7 ou 8 cailloux et son adversaire lui en laissera 5 qui est un nombre perdant d'après **1.c**.

2.b. Il existe une stratégie gagnante pour un joueur si, au moment où il laisse la main, le nombre de cailloux restants est de la forme $4n + 1$ où n est un entier strictement positif. Suivant que son adversaire prendra 1, 2 ou 3 cailloux, ce joueur prendra 3, 2 ou un seul caillou. Au total, 4 cailloux auront été pris au cours de ce tour et il restera $4(n - 1) + 1$ cailloux. Au bout d'un nombre fini de tours, il ne restera plus qu'un caillou que son adversaire sera obligé de prendre. Par exemple, s'il y a 20 cailloux, on commence par en prendre 3.

S'il y a 20 cailloux et si Amélie commence la partie, elle est sûre de gagner en prélevant trois cailloux.

3. Si on peut prélever de 1 à 5 cailloux, il existe une stratégie gagnante pour un joueur si, au moment où il laisse la main, le nombre de cailloux restants est de la forme $6n+1$ où n est un entier strictement positif. Suivant que son adversaire prendra 1, 2, 3, 4 ou 5 cailloux, ce joueur prendra 5, 4, 3, 2 ou un seul caillou. Au total, 6 cailloux auront été pris au cours de ce tour et il restera $6(n-1)+1$ cailloux. Au bout d'un nombre fini de tours, il ne restera plus qu'un caillou que son adversaire sera obligé de prendre.

Ainsi, pour être sûre de gagner en laissant Balthasar commencer la partie, Amélie doit choisir de jouer avec 25, 31 ou 37 cailloux.

4. De façon générale, si on peut prélever de 1 à p cailloux, il existe une stratégie gagnante pour un joueur si, au moment où il laisse la main, le nombre de cailloux restants est de la forme $k\times(p+1)+1$ où k est un entier strictement positif. Suivant que son adversaire prendra 1, 2, ... ou p cailloux, ce joueur prendra p, $p-1$,..., 2 ou un seul caillou. Au total, $(p+1)$ cailloux auront été prélevés au cours de ce tour et il restera $(k-1)\times(p+1)+1$ cailloux. Au bout d'un nombre fini de tours, il ne restera plus qu'un caillou que son adversaire sera obligé de prendre.

Les nombres perdants sont les nombres de la forme $k\times(p+1)+1$ où k est un entier positif.

<table>
<tr>
<td>5. Puisqu'il y a 20 cailloux au début du jeu, l'ordinateur doit commencer par prendre 3 cailloux pour en laisser 17. Ensuite, l'ordinateur prélève toujours le nombre dont la somme avec le choix du joueur est égale à 4. Jusqu'à ce qu'il ne reste qu'un seul caillou (la partie est alors gagnée).</td>
<td>

```
>>> def nimordi():
	print("l'ordi prend trois cailloux")
	n=17
	print("il reste 17 cailloux")
	while n>3:
		x=randint(1,3)
		print("le joueur prend ",x," caillou(x)")
		n=n-x
		print("il reste ",n," caillou(x)")
		o=4-x
		print("l'ordi prend ",o," caillou(x)")
		n=n-o
		print("il reste ",n," caillou(x)")
```

</td>
</tr>
</table>

Voici deux exemples d'exécution de ce programme **nimordi** :

<table>
<tr>
<td>

```
>>> nimordi()
l'ordi prend trois cailloux
il reste 17 cailloux
le joueur prend  3  caillou(x)
il reste  14  caillou(x)
l'ordi prend  1  caillou(x)
il reste  13  caillou(x)
le joueur prend  2  caillou(x)
il reste  11  caillou(x)
l'ordi prend  2  caillou(x)
il reste  9  caillou(x)
le joueur prend  2  caillou(x)
il reste  7  caillou(x)
l'ordi prend  2  caillou(x)
il reste  5  caillou(x)
le joueur prend  3  caillou(x)
il reste  2  caillou(x)
l'ordi prend  1  caillou(x)
il reste  1  caillou(x)
```

</td>
<td>

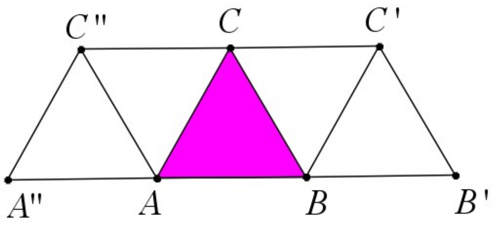
```
>>> nimordi()
l'ordi prend trois cailloux
il reste 17 cailloux
le joueur prend  2  caillou(x)
il reste  15  caillou(x)
l'ordi prend  2  caillou(x)
il reste  13  caillou(x)
le joueur prend  1  caillou(x)
il reste  12  caillou(x)
l'ordi prend  3  caillou(x)
il reste  9  caillou(x)
le joueur prend  1  caillou(x)
il reste  8  caillou(x)
l'ordi prend  3  caillou(x)
il reste  5  caillou(x)
le joueur prend  3  caillou(x)
il reste  2  caillou(x)
l'ordi prend  1  caillou(x)
il reste  1  caillou(x)
```

</td>
</tr>
</table>

Partie II : avec des triangles, le dernier gagne

Examinons, sur la figure ci-dessous que nous avons codée, les possibilités de découpes :

- Si Balthasar découpe suivant le segment [AC] (respectivement, [BC]), Amélie découpe suivant le segment [BC] (respectivement, [AC]) et elle gagne.
- Si Balthasar découpe suivant le segment [BC'] (respectivement, [AC'']), Amélie découpe suivant le segment [AC''] (respectivement, [BC']). Il ne reste que trois triangles, ABC et deux de part et d'autre de ABC. Balthasar est obligé de découper l'un des deux triangles autres que ABC, Amélie découpe l'autre et elle gagne.



Si Amélie adopte la meilleure stratégie, en aucun cas, Balthasar ne peut gagner la partie.

La configuration proposée par l'énoncé est donc perdante.

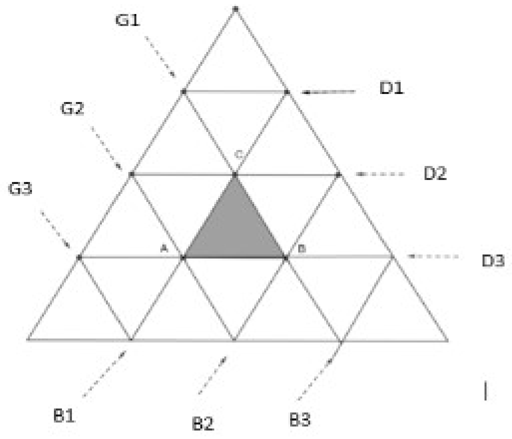
Fin de la correction

Une remarque à propos du « dernier gagnant » de cette partie II

Revenons au début du jeu et examinons les découpes possibles de Balthasar (qui commence) et celles d'Amélie.

Si Balthasar découpe suivant B1 (ou D3 ou G1), alors Amélie découpe suivant B2 (ou D2 ou G2 respectivement) et laisse à Balthasar la configuration perdante étudiée, elle gagne.

Si découpe suivant B2 (ou D2 ou G2), alors Amélie découpe suivant B1 (ou D3 ou G1 respectivement) et laisse à Balthasar la configuration étudiée, elle gagne.



Balthasar doit donc découper un seul triangle, suivant D1 ou B3 ou G3 pour conserver ses chances de gagner (par exemple D1). Amélie fera, semble-t-il, de même. Balthasar découpera le triangle restant et il semble bien qu'il laissera à Amélie une configuration perdante. Nous ne sommes donc pas complètement persuadés que, comme le dit l'énoncé, « le dernier gagne ». Nous laissons au lecteur le soin de se forger sa propre opinion.