Suicide collectif

Nadège Albrespic, Alexia Dorkel, Guillaume Rincel, Rémi Courtin, Thomas Barillot, Thomas Perelló, Florent Imbert

2017-2018

Établissements : Universités de Bordeaux et La Rochelle

Chercheurs : Marie-Line Chabanol et Gilles Bailly-Maître

Résumé :
Des soldats se tuent mutuellement suivant un procédé détaillé dans la suite. Dans cet article on donne des algorithmes pour déterminer le survivant, ainsi que des formules permettant de trouver le survivant en fonction du nombre de soldats.

Remerciements : La confrérie des matheux, l'université de Bordeaux, l'UF de mathématiques et interactions de l'université de Bordeaux, l'UF Informatique de l'université de Bordeaux, le département de Mathématiques de l'université de La Rochelle

1 Présentation du sujet

1.1 Enoncé

Il y a bien longtemps, le soldat Cléombrote fut fait prisonnier avec ses camarades par l'armée athénienne. Or à Sparte être fait prisonnier était un grand déshonneur, et la règle était que les soldats devaient se tuer mutuellement. Ils procédaient ainsi : ils se mettaient en cercle ; l'un d'entre eux (1) tuait son voisin de gauche (2), puis le voisin suivant (3) tuait son voisin de gauche, et ainsi de suite, en faisant autant de tours que nécessaire pour qu'il n'en reste plus qu'un, qui devait alors se suicider. Mais Cléombrote ne veut pas mourir. Où doit-il se placer pour être le dernier survivant ? Et si la règle est différente ? Par exemple, si à chaque fois on tue ses deux voisins de gauche ? Ou si on saute quelqu'un à chaque fois ? (1 tue 2 puis 4 tue 5, etc).

Dans cet article, on donne des algorithmes pour déterminer le survivant, ainsi que des formules permettant de trouver le survivant en fonction du nombre de soldats. On a également traité le cas où on tue à chaque fois k soldats.

1.2 Notations

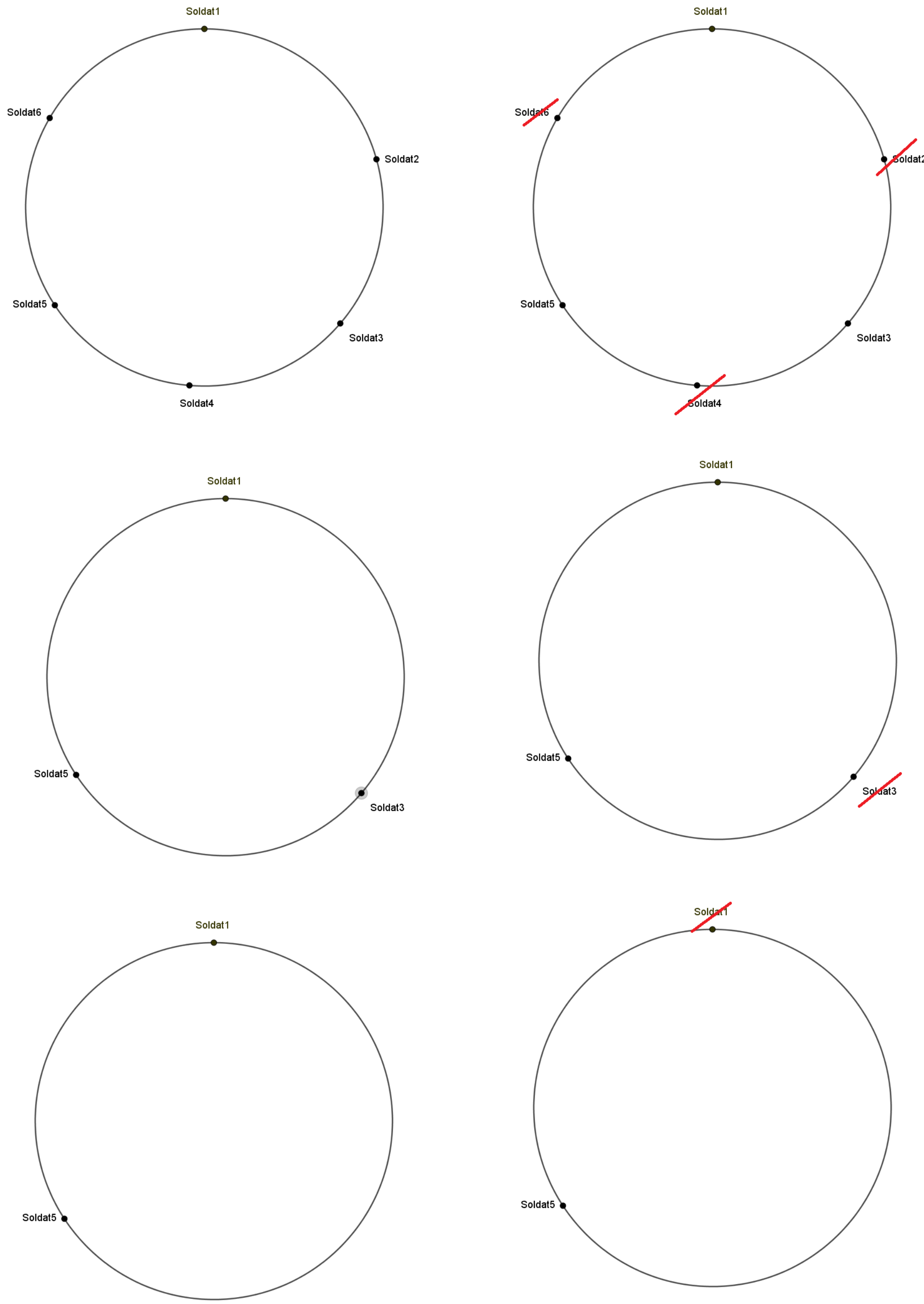
On note n le nombre de soldats. Les soldats sont numérotés par les entiers de 1 à n, 1 étant celui qui commence à tuer. Le numéro 2 est attribué à son voisin de gauche, puis 3 au voisin de gauche du numéro 2, etc.

On note $S(n)$ le numéro du survivant dans un groupe de n.

L'état de notre cercle de soldat est modélisé par une liste ordonnée des numéros des soldats encore en vie, où le premier numéro de la liste est celui du soldat dont c'est le tour de tuer, et le suivant son voisin de gauche, etc.

Exemple : au départ, le cercle de 6 soldats est représenté par la liste (1, 2, 3, 4, 5, 6).
Après que 1 ait tué 2, on obtient la liste (3, 4, 5, 6, 1), puis, 3 tuant 4, (5, 6, 1, 3), puis (1, 3, 5), puis (5, 1), puis (5) et 5 est donc le survivant.

Illustration en image :



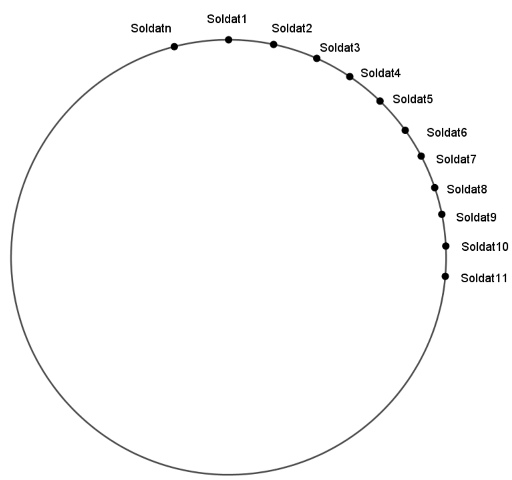
2 Démarche

Dans un premier temps, pour avoir une idée de la suite de nombres qu'on étudiait, on en a calculé les premières valeurs, d'abord à la main, puis par ordinateur. Il a fallu pour cela traduire le sujet sous la forme d'algorithme.

2.1 Algorithmes

2.1.1 Traduction immédiate de l'énoncé

Le cercle de soldats est représenté par une liste d'entiers (lesdits entiers étant les numéros affectés aux soldats). Le premier soldat de la liste est celui dont c'est le tour, le suivant est son voisin de gauche, etc. : au début de l'algorithme, la liste est donc $(1, 2, \ldots, n)$. Puisqu'il s'agit d'un cercle, le voisin de gauche du dernier de la liste (ici n) est le premier (ici 1).
Pour passer d'une liste $(a, b, c, \ldots, m)$ à la liste qui représente l'état suivant : $(c, \ldots, m, a)$, on extrait les deux premiers éléments de la liste, et seul le premier est replacé dans la liste, à la fin (le deuxième est tué).



On obtient l'algorithme suivant :

```
Algorithme survivant(n) :
      L ← (1, 2, ..., n)
      Tant que L a au moins deux éléments faire:
            tueur  ← extraire_premier_élément(L)
            tué ← extraire_premier_élément(L)
            ajouter tueur à la fin de L
      Fin Tant que
      retourner L[1]
```

Cet algorithme se termine en $n-1$ étapes puisqu'au bout de $n-1$ meurtres, il reste bien un seul soldat. Calculer les n premiers termes de la suite $(S(n))$ se fait donc en un temps quadratique[1].

2.1.2 Relation de récurrence vérifiée par la suite $(\mathbf{S(n)})$

En partant de la liste $(1, 2, \ldots, n)$, si on effectue un seul meurtre, on obtient la liste $L = (3, 4, \ldots, n, 1)$, qui contient $n-1$ soldats. Si on connaît $S(n-1)$, alors le survivant sera le $S(n-1)$-ième soldat de notre

1. C'est-à dire que le nombre d'opérations à faire pour calculer ces n premiers termes est de l'ordre de n^2. Plus précisément, il faut $\frac{n(n-1)}{2}$ opérations (la somme des $n-1$ premiers entiers).

nouvelle liste L. On en déduit que

$$S(n) = \begin{cases} S(n-1)+2 & si \quad S(n-1) < n-1 \\ 1 & si \quad S(n-1) = n-1 \end{cases}$$

Cette relation reliant $S(n)$ et $S(n-1)$ (appelée relation de récurrence) donne un algorithme simple pour calculer la liste des n premiers termes de la suite [1] :

```
Algorithme survivant_jusque(n) :
    L ← (1)
    s ← 1
    Pour i de 1 à n faire :
        si  s < i − 1
            s ← s + 2
        sinon
            s ← 1
        ajouter s à la fin de L
    retourner L
```

Ce nouvel algorithme calcule les n premiers termes de la suite en $n-1$ opérations : c'est plus efficace que le précédent. On peut également utiliser cette relation de récurrence pour remplir un tableur avec les valeurs de la suite.

2.2 Table de valeurs

Ces algorithmes permettent de construire une table de valeurs de notre suite $S(n)$.

n	S
1	1
2	1
3	3
4	1
5	3
6	5
7	7
8	1
9	3
10	5
11	7
12	9
13	11
14	13
15	15
16	1
17	3
18	5
19	7
20	9
21	11
22	13
23	15
24	17
25	19
26	21

2.3 Conjectures basées sur cette table

La suite $S(n)$ semble être une énumération des nombres impairs qui recommence à 1 pour certaines valeurs de n.

En y regardant de plus près, on remarque que la suite reprend à 1 lorsque n vaut $1, 2, 4, 8, 16, 32, 64 \ldots$: Il semblerait que ce soit lorsque n est une puissance de 2.

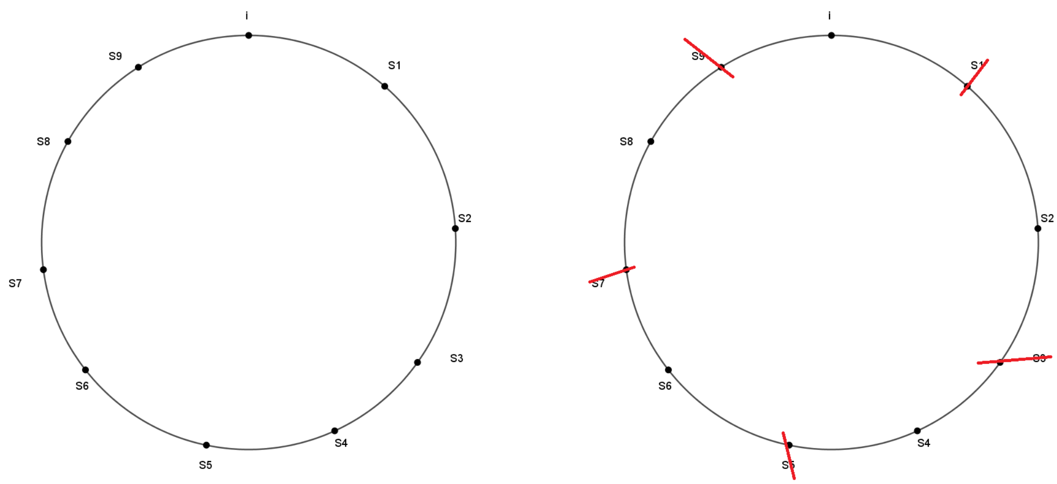
3 Résultats

3.1 Problème initial

Lemme 0.1 :
Soient n le nombre de soldats encore vivants, i le numéro du soldat dont c'est le tour.
Si n est pair et $k = n/2$, ce sera à nouveau le tour du soldat i après k tours et il restera k soldats vivants.

Preuve :
On re-numérote les soldats de façon à ce que le soldat i ait maintenant le numéro 1.
1 tue 2, 3 tue 4, ..., $2k-1$ tue $2k$, c'est de nouveau le tour de 1.

Exemple :



Ici il y a $n = 10 = 2 \times 5$ soldats (donc $k = 5$), c'est initialement le tour du soldat i et après 5 meurtres, c'est de nouveau son tour, et il reste 5 soldats.

Lemme 0.2 :
Soient n le nombre de soldats encore vivants et i le numéro du soldat dont c'est le tour. Si n est une puissance de 2, alors l'ultime survivant est le soldat i.

Preuve : On pose $n = 2^p$. On applique p fois le lemme 0.1 [2] : il reste un soldat et c'est le soldat i.

Théorème 0.3 :
Soit n le nombre de soldats dans le cercle initial. n peut s'écrire de façon unique $2^p + r$ avec p et r entiers naturels tels que $r < 2^p$. Par convention, le premier soldat qui tue son voisin porte le numéro 1. Alors, le survivant sera le soldat numéro $2r + 1$.

Preuve :
On ne fera pas ici la preuve de l'existence et de l'unicité de cette écriture.
Après r tours, c'est le tour du soldat $2r + 1$ (par récurrence : après 0 tour c'est le tour du soldat $1 = 2 \times 0 + 1$; et si c'est le tour de $2i+1$ après i tours, au $i+1$-ième tour $2i+1$ tue $2i+2$ et c'est le tour de $2i + 3 = 2(i + 1) + 1$, tant que $2i + 3 \leq n$. En l'occurrence on a imposé $r < 2^p$ donc $2r < 2^p + r = n$, $2r + 1 \leq n$).
On a alors tué r soldats (un à chaque tour) donc il en reste 2^p. D'après le lemme 0.2, le survivant sera donc le soldat $2r + 1$.

Exemple : Pour $n = 42$ (il y a donc 42 soldats).
On cherche la plus grande puissance de 2 inférieure à 42 ; listons les puissances de 2 : 1, 2, 4, 8, 16, 32, 64... la puissance de 2 la plus grande inférieure à 42 est $32 = 2^5$.
$42 = 2^5 + 10$: ici $p = 5$ et $r = 10$ donc, d'après le théorème 0.3, le survivant est le soldat numéro $2r + 1 = 2 \times 10 + 1 = 21$.

Expression générale des termes de la suite $(\mathbf{S(n)})$ (pour ceux qui connaissent le logarithme et la partie entière) :
On a vu que si $n = 2^p + r$ avec p et r entiers naturels tels que $r < 2^p$, alors $S(n) = 2r + 1$.
Cela signifie que p est alors la plus grande puissance de 2 inférieure à n : $2^p \leq n < 2^{p+1}$. En passant au logarithme népérien dans tous les membres de cette inégalité, on obtient $p \ln(2) \leq \ln(n) < (p+1)\ln(2)$.
On divise alors tous les membres par $\ln(2)$: $p \leq \frac{\ln(n)}{\ln(2)} < p + 1$.
p est donc l'entier qui vient immédiatement avant $\frac{\ln(n)}{\ln(2)}$: on l'appelle partie entière de $\frac{\ln(n)}{\ln(2)}$ et on le note $\lfloor \frac{\ln(n)}{\ln(2)} \rfloor$. Ainsi $p = \lfloor \frac{\ln(n)}{\ln(2)} \rfloor$, et donc

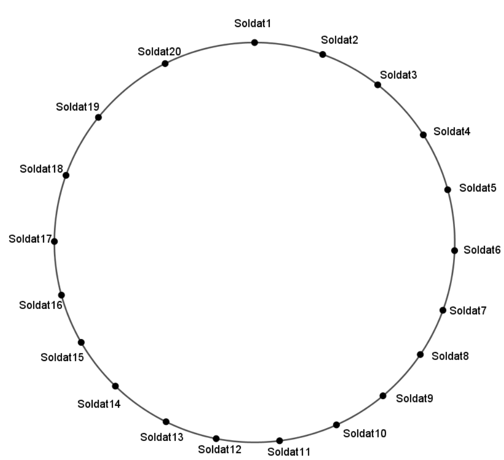
$$S(n) = 2\left(n - 2^{\lfloor \frac{\ln(n)}{\ln(2)} \rfloor}\right) + 1$$

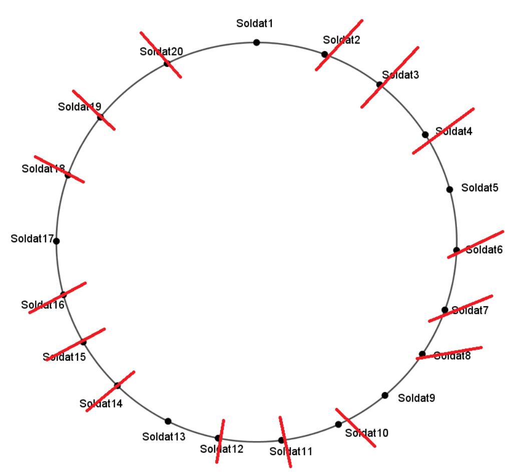
Lien avec l'écriture en base 2 :
On écrit n en base 2. Le numéro du survivant est obtenu en faisant une permutation circulaire des chiffres vers la gauche.
Exemple :
$189 = 128 + 32 + 16 + 8 + 4 + 1$ s'écrit 10111101 en base 2. $S(189)$ s'écrit donc 1111011 en base 2 qui correspond à 123 en base 10 [3].

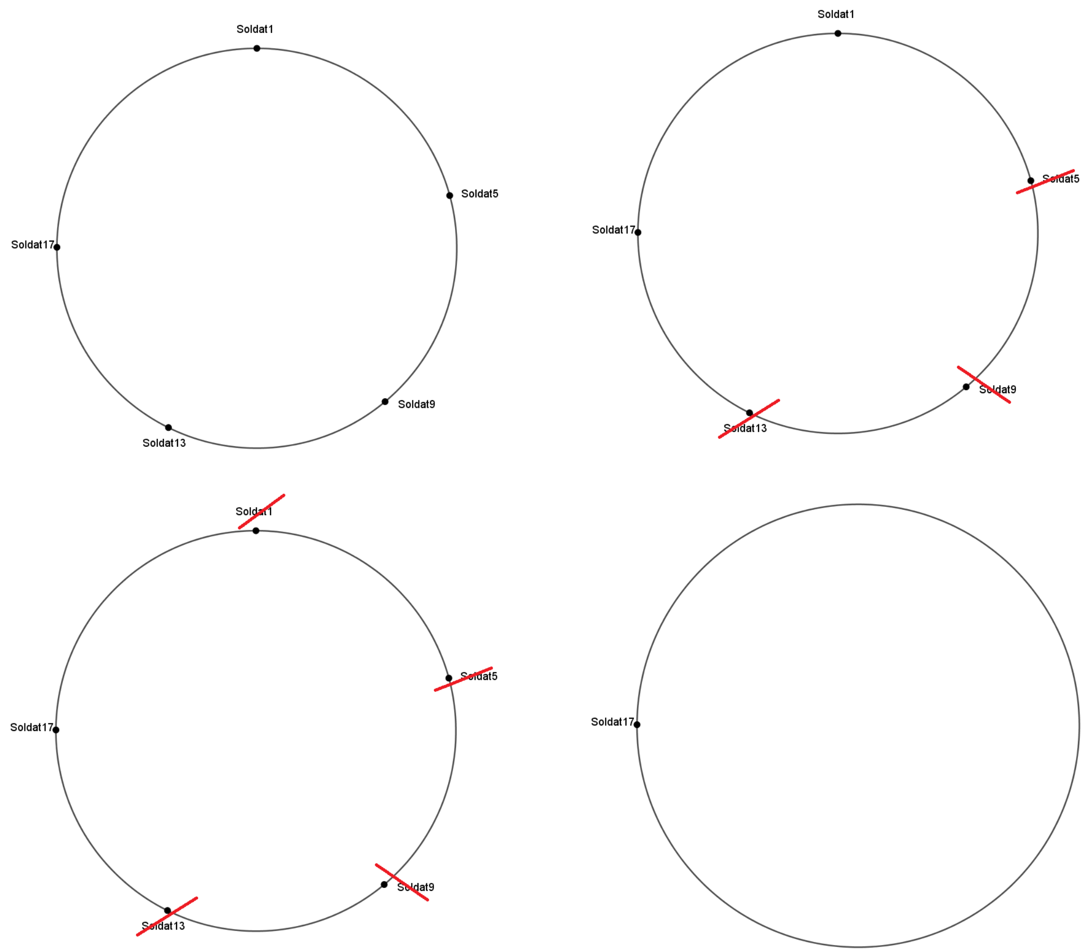
En effet on a vu que le survivant est $2r + 1$ lorsque $n = 2^p + r$ où p est la plus grande puissance de 2 inférieure à n. Donc r s'obtient à partir de l'écriture de n en enlevant le 1 de poids le plus fort (i.e. le plus à gauche). Pour notre exemple, r s'écrit 111101. Multiplier par 2 en base 2 revient à décaler tous les chiffres d'un cran vers la gauche (similairement à la multiplication par 10 en base 10) : ici $2r = 1111010$ et finalement $2r + 1 = 1111011$.

3.2 Généralisation

On étudie maintenant la même situation, mais au lieu de tuer seulement son voisin de gauche, le soldat dont c'est le tour tue k soldats sur sa gauche ($k > 1$).
On note maintenant $S(n, k)$ le numéro du survivant dans un groupe de n soldats lorsque les soldats sont tués k par k.
Exemple :
On se place dans le cas où $n = 20$ et $k = 3$, c'est-à-dire qu'il y a en tout 20 soldats et que le soldat qui a le couteau tue ses trois voisins de gauche.







3.2.1 Généralisation des algorithmes

On modifie les algorithmes précédemment écrits pour tenir compte de notre nouveau paramètre k.

Algorithme survivant(n, k) [4] :

$L \leftarrow (1, 2, \ldots, n)$
Tant que L a plus de $k+1$ éléments faire :
  tueur $\leftarrow$ extraire_premier_élément(L)
  Répéter k fois :
    tué $\leftarrow$ extraire_premier_élément(L)
  ajouter tueur à la fin de L
Fin Tant que
retourner $L[1]$

Comme précédemment, on peut trouver une relation de récurrence vérifiée par la suite $S(n, k)$. En partant de la liste $(1, 2, \ldots, n)$, (en supposant $n > k$), si on effectue un seul meurtre de k personnes, on obtient la liste $L = (k+2, k+3, \ldots n, 1)$, qui contient $n-k$ soldats. Si on connaît $S(n-k, k)$, alors le survivant $S(n, k)$ sera le $S(n-k, k)$-ième soldat de notre liste L. On en déduit que

$$S(n) = \left\{ \begin{array}{rll} S(n-k,k)+k+1 & si & S(n-k) < n-k \\ 1 & si & S(n-k,k) = n-k \end{array} \right.$$

Cette fois, on n'a pas exprimé $S(n)$ en fonction de $S(n-1)$ mais en fonction de $S(n-k)$: la relation de récurrence obtenue définit donc k sous-suites de la forme $(S(mk, k))$, $(S(mk+1, k))$, ..., $(S(mk+(k-1), k))$.

```
Algorithme survivant_jusque(n, k) :
    L ← (1, 1, ..., 1)   //liste de taille k
    s ← (1, 1, ..., 1)
    Pour i de 1 à n par pas de k faire :
        Pour j de 1 à k faire :
        si L[i] < i + j
            s[i] ← s[i] + k
        sinon
            s[i] ← 1
        ajouter les éléments de s à la fin de L
    retourner L
```

3.2.2 Table des valeurs

n	$S(n,1)$	$S(n,2)$	$S(n,3)$	$S(n,4)$
1	1	**1**	**1**	**1**
2	1	**1**	**1**	**1**
3	3	**1**	**1**	**1**
4	1	4	**1**	**1**
5	3	4	5	**1**
6	5	**1**	5	6
7	7	7	5	6
8	1	4	**1**	6
9	3	**1**	9	6
10	5	7	9	**1**
11	7	4	5	11
12	9	10	**1**	11
13	11	7	13	11
14	13	13	9	6
15	15	10	5	**1**
16	1	16	**1**	16
17	3	13	13	16
18	5	**1**	9	11
19	7	16	5	6
20	9	4	17	**1**
21	11	19	13	21
22	13	7	9	16
23	15	22	21	11
24	17	10	17	6
25	19	25	13	**1**
26	21	13	25	21

[5]

3.2.3 Lien avec l'écriture en base $\mathbf{k+1}$:

On a cherché un autre moyen d'exprimer les différentes suites obtenues. Pour ce faire nous avons donc décidé de les exprimer dans une base différente : la base $k+1$. Par exemple pour le cas $k=3$ (on tue 3 soldats par 3 soldats), on exprime les résultats en base 4 et on obtient le tableau suivant [6] :

n	$\overline{S(n,3)}^{(4)}$
1	1
2	1
3	1
4	1
5	11
6	11
7	11
8	1
9	21
10	21
11	11
12	1
13	31
14	21
15	11
16	1
17	31
18	21
19	11
20	101
21	31
22	21
23	111
24	101
25	31
26	121

Autre exemple : si on tue 9 personnes par 9 personnes (le cas où $k = 9$), on exprime donc les résultats en base 10. On obtient donc :

n	$\overline{S(n,9)}^{(10)}$
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	11
12	11
13	11
14	11
15	11
16	11
17	11
18	11
19	11
20	1
21	21
22	21
23	21
24	21
25	21
26	21

En créant ces tableaux, on a constaté une forte similitude entre les écritures en base $k + 1$ des numéros des survivants dans le cas où celui-ci était la n-ième personne (cas où $S(n, k) = n$).
En effet si on reprend le cas précédent, pour $k = 3$ on obtient :

$$\begin{aligned}
S(1,3) &= 1 = 1 \times 4^0 = \overline{1}^{(4)} \\
S(5,3) &= 5 = 1 \times 4^1 + 1 \times 4^0 = \overline{11}^{(4)} \\
S(9,3) &= 9 = 2 \times 4^1 + 1 \times 4^0 = \overline{21}^{(4)} \\
&\dots
\end{aligned}$$

On a ensuite généralisé cette observation. Si on tue k personnes par k personnes et qu'on exprime le numéro des soldats en base $k+1$ on obtient :
Valeurs de n, en base $k+1$, pour lesquelles $S(n,k)=n$:

$$\begin{gathered}
\overline{1}^{(k+1)} \\
\overline{11}^{(k+1)} \\
\overline{21}^{(k+1)} \\
\overline{31}^{(k+1)} \\
\vdots \\
\overline{k1}^{(k+1)} \\
\overline{1k1}^{(k+1)} \\
\overline{2k1}^{(k+1)} \\
\overline{3k1}^{(k+1)} \\
\vdots \\
\overline{kk1}^{(k+1)} \\
\overline{1kk1}^{(k+1)} \\
\overline{2kk1}^{(k+1)} \\
\overline{3kk1}^{(k+1)} \\
\vdots \\
\overline{kkk1}^{(k+1)} \\
\overline{1kkk1}^{(k+1)} \\
\overline{2kkk1}^{(k+1)} \\
\overline{3kkk1}^{(k+1)} \\
\vdots
\end{gathered}$$

De manière plus générale, les nombres pour lesquels on a $S(n,k)=n$ s'écrivent en base $k+1$ sous la forme $n=\overline{\alpha\underbrace{kk\dots k}_{\text{p fois}}1}^{(k+1)}$, où p est un entier et α un entier compris entre 0 et $k-1$.
On peut déduire de cette écriture une expression des valeurs de n pour lesquelles $S(n,k)=n$. En effet :

$$n=\overline{\alpha kk\dots k1}^{(k+1)}=\alpha\times(k+1)^{p+1}+k\times(k+1)^{p}+k\times(k+1)^{p-1}+\cdots+k\times(k+1)^{1}+1\times(k+1)^{0}$$

On factorise par $k\times(k+1)$:

$$n=\alpha\times(k+1)^{p+1}+k\times(k+1)\times[1+(k+1)+(k+1)^2+\cdots+(k+1)^{p-1}]+1$$

On utilise la formule pour la somme des termes d'une suite géométrique (vue en première) :

$$1+(k+1)+(k+1)^2+\cdots+(k+1)^{p-1}=\frac{1-(k+1)^p}{1-(k+1)}=\frac{(k+1)^p-1}{k}$$

D'où $n=\alpha\times(k+1)^{p+1}+k\times(k+1)\times\frac{(k+1)^p-1}{k}=\alpha\times(k+1)^{p+1}+(k+1)^{p+1}-k-1+1$, et $n=(\alpha+1)\times(k+1)^{p+1}-k$.
On a donc la conjecture :
Si $n=(\alpha+1)\times(k+1)^{p+1}-k$ avec α entier entre 0 et $k-1$ et p entier, alors $S(n,k)=n$.

3.2.4 Résultats

Les résultats pour $k = 1$ se généralisent.

Lemme 1.1 (généralisation du lemme 0.1) :
Soient n le nombre de soldats encore vivants et i le numéro du soldat dont c'est le tour.
Si n est divisible par $k+1$ et $m = n/(k+1)$, ce sera à nouveau le tour du soldat i après m tours et il restera m soldats vivants.

Preuve :
On re-numérote les soldats de façon à-ce-que le soldat i ait maintenant le numéro 1.
Alors 1 tue $[2, 3, \ldots, k+1]$, $k+2$ tue $[k+3, \ldots, 2k+2]$, ...,
$(m-1)(k+1)+1$ tue $[(m-1)(k+1)+2, \ldots, (m-1)(k+1)+k+1 = (k+1)m = n]$, c'est de nouveau le tour de 1.

Lemme 1.2 (généralisation du lemme 0.2) :
Soient n le nombre de soldats encore vivants et i le numéro du soldat dont c'est le tour.
Si $n = a \times (k+1)^p$ avec $1 \leq a < k+1$, alors le soldat i sera l'ultime survivant.

Preuve : On applique le lemme 1.1 p fois : il reste a soldats et c'est le tour du soldat i. Comme $a < k+1$, le soldat i tue tous les $k-1$ autres survivants.

Problème quand on veut appliquer la même méthode que pour $\mathbf{k = 1}$:
n peut s'écrire de façon unique sous la forme $a \times (k+1)^p + r$, avec $1 \leq a < k+1$ et $0 \leq r < (k+1)^p$.
Si k divise r, après $\frac{r}{k}$ tours il reste $a \times (k+1)^p$ soldats et ce sera le tour du soldat $\frac{r}{k}(k+1)+1$: on peut conclure par le lemme 1.2 que le survivant est le soldat $\frac{r}{k}(k+1)+1$.

Exemple :
Pour $n = 7567$ et $k = 9$ (il y a 7567 soldats et ils se tuent 9 par 9), on a $7567 = 7 \times (9+1)^3 + 567$ et $567 = 9 \times 63$. Si on effectue 63 séries de 9 meurtres, il restera 7×10^3 soldats, et ce sera le tour du soldat numéro $1 + 10 \times 63 = 631$: d'après le lemme 1.2 le survivant est le soldat 631.

Sinon, dans les $k-1$ autres cas possibles de reste dans la division de r par k, quel est le survivant ?
On a vu en 3.2.1 que les suites $(S(mk+q, k))$ avec q entre 0 et $k-1$ vérifient toutes la même relation de récurrence, et donc, ce sont des suites arithmétiques de raison k et de premier terme 1 qui "redémarrent" à 1 de temps en temps. Le cas $r = 0$ vient d'être traité dans le lemme 1.2.

Exemple :
Pour $n = 7571$ et toujours $k = 9$ (il y a 7571 soldats et ils se tuent 9 par 9), on a $7571 = 7 \times 10^3 + 571$ et $571 = 9 \times 63 + 4$
571 n'est pas divisible par 9. On ne peut donc pas "se débarrasser" des 571 soldats pour se ramener au lemme 1.2. Au mieux, si on effectue 63 séries de 9 meurtres, il restera $7 \times 10^3 + 4$ soldats. Ce qui ne permet pas de déterminer le survivant.

Cet obstacle nous a bloqués pendant un certain temps, mais nous en sommes finalement venus à bout : dans les cas où k ne divise pas r (et en particulier $r \neq 0$), il faut chercher à écrire n sous la forme $a(k+1)^p + qk$. Comme $k+1 \equiv 1[k]$ [7], $(k+1)^p \equiv 1[k]$ et comme de plus $qk \equiv 0[k]$, $a(k+1)^p + qk \equiv a[k]$.
Pour appliquer le lemme 1.2, il faudra de plus $1 \leq a < k+1$; le coefficient a doit donc être le reste dans la division euclidienne de n par k, ou si le reste est nul $a = k$. Une telle écriture de n existe : en effet la division euclidienne de n par k nous donne : $n = qk + a = qk + a(k+1)^0$.

Choisissons p le plus grand entier tel qu'une telle écriture existe avec $q \geq 0$, c'est-à-dire $a(k+1)^p \leq n < a(k+1)^{p+1}$. On effectue q tours, de façon à tuer qk soldats ; il reste alors $a(k+1)^p$ soldats, et c'est le tour du soldat $1 + q(k+1)$, pourvu que $1 + q(k+1) \leq n$ et donc qu'on fait moins d'un tour de cercle, ce qu'on va vérifier. D'après le lemme 1.2, le survivant sera donc le soldat numéro $q(k+1)+1$.

Vérifions maintenant que $1 + q(k+1) \leq n$:
L'entier p est le plus grand possible tel qu'il existe q positif avec $n = a(k+1)^p + qk$, donc

$$\begin{aligned} n - a(k+1)^{p+1} &< 0 \\ n - (k+1)(n-qk) &< 0 \\ n(1-k-1) + (k+1)qk &< 0 \\ -n + (k+1)q &< 0 \\ 1 + (k+1)q &\leq n. \end{aligned}$$

Calcul de p
On a choisi p le plus grand possible tel que $n = a(k+1)^p + q \times k$.
Par conséquent, on a $a(k+1)^p \leq n < a(k+1)^{p+1}$.
Comme $a \neq 0$ (le cas $a = 0$ ayant été traité séparément en premier), on a donc

$$(k+1)^p \leq \frac{n}{a} < (k+1)^{p+1}$$

On passe au logarithme et on divise par le logarithme de $k+1$: $p \leq \frac{\ln(n/a)}{\ln(k+1)} < p+1$
Ainsi p est la partie entière de $\frac{\ln(n/a)}{\ln(k+1)}$:

$$p = \left\lfloor \frac{\ln\left(\frac{n}{a}\right)}{\ln(k+1)} \right\rfloor$$

Calcul de q, puis de $S(n,k)$
Connaissant p, et puisque $n = a(k+1)^p + qk$, on a donc $q = (n - a(k+1)^p)/k$.
On a vu que $S(n,k) = (k+1)q + 1$, en substituant, on obtient $S(n,k) = \frac{(k+1)(n-a(k+1)^p)}{k} + 1$, ou encore, en explicitant p, $S(n,k) = \frac{(k+1)(n-a(k+1)^{\left\lfloor \frac{\ln(n/a)}{\ln(k+1)} \right\rfloor})}{k} + 1$.
On obtient ainsi le théorème 1.3 :
Pour tout n,

$$S(n,k) = \frac{(k+1)(n-a(k+1)^{\left\lfloor \frac{\ln(n/a)}{\ln(k+1)} \right\rfloor})}{k} + 1,$$

où a est le reste de n dans la division euclidienne par k s'il est non nul, et sinon $a = k$ [8].

Exemple : Reprenons le cas précédent qui posait problème où il y avait 7571 soldats.
Calcul de a : a est le reste de la division euclidienne de 7571 par 9 : $7571 = 841 \times 9 + 2$ donc $a = 2$.
On peut maintenant déterminer p de 2 façons :

1. "à la main" : p est le plus grand entier tel que $2 \times 10^p \leq n$. On a $2 \times 10^3 = 2000$ et $2 \times 10^4 = 20000$; $20000 > 7571$ donc $p = 3$;
2. à la calculatrice : $\frac{\ln(n/a)}{\ln(10)} = \frac{\ln(7571/2)}{\ln(10)} \approx 3,5781$ donc $p = 3$.

Déterminons à présent q :
On a $n = a(k+1)^p + q \times k = 200 + q \times 9$ donc $q = \frac{7571-2000}{9} = 619$.
Ainsi $S(7571, 9) = 10q + 1 = 6191$.

4 Conclusion

Une autre question proposée dans l'énoncé était de "sauter" un nombre fixe de soldats entre deux tours : par exemple, 1 tue 2, puis 4 tue 5, puis 7 tue 8 etc. Nous avons donc également pensé à une autre généralisation du problème. Au lieu de tuer k soldats, on en aurait laissé k vivants entre 2 assassinats : 1 tue 2, passe le couteau au soldat $k+2$, qui tue le soldat $k+3$ et passe le couteau au soldat $2k+3$, etc. Cependant, nous n'avons pas eu le temps de nous pencher sur la question, nous n'avons donc pas pu établir de résultats.

Notes d'édition

[1] Il est clair que $S(1) = 1$, et donc la relation de récurrence permet bien de déterminer tous les termes de la suite.

[2] Chaque fois que le lemme 0.1 est appliqué, le nombre de soldats est divisé par 2.

[3] 1111011 en base 2 correspond à $1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 64 + 32 + 16 + 8 + 2 + 1 = 123$ en base 10.

[4] La condition d'arrêt de cet algorithme est $L \leq k + 1$ car s'il y a jusqu'à $k + 1$ soldats le premier soldat tue tous les autres et est donc le survivant. On a en particulier $S(1) = S(2) = \ldots = S(k + 1) = 1$.

[5] Il n'est pas dit à quoi correspondent les couleurs dans ce tableau. Elles visualisent les sous-suites $(S(mk + r, k))$ (pour $0 \leq r < k$).

[6] La notation $\overline{n}^{(4)}$ dans le tableau représente l'écriture en base 4 du nombre n, par exemple $\overline{21}^{(4)}$ représente l'écriture en base 4 du nombre 9, où 1 est le chiffre des unités et 2 celui des "quatraines" $(9 := 2 \times 4^1 + 1 \times 4^0)$.

[7] L'écriture $m \equiv n[k]$ ("m est congru à n modulo k") signifie que m et n ont même reste dans la division euclidienne par k, autrement dit que $m - n$ est divisible par k. On peut additionner et multiplier terme à terme les congruences modulo un entier k donné.

[8] En pratique on procèdera comme dans l'exemple en-dessous : déterminer successivement les entiers a, p et q.
Cela donne la preuve de la conjecture sur les entiers n tels que $S(n, k) = n$ énoncée en fin de § 3.2.3 : si $n = a(k + 1)^{(p+1)} - k$, n s'écrit aussi $a(k + 1)^p + k[a(k + 1)^p - 1]$ et $S(n, k) = 1 + (k + 1)[a(k + 1)^p - 1] = a(k + 1)^{(p+1)} + 1 - k - 1 = n$.