

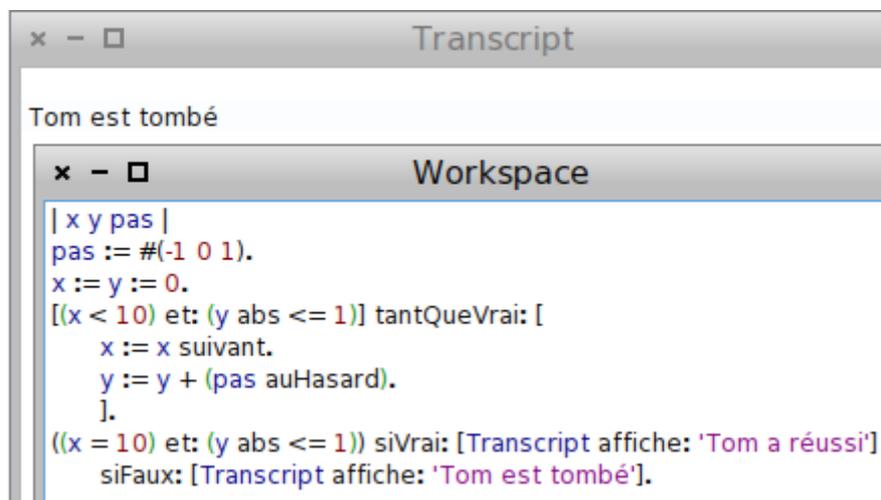
Traitement mathématique de l'algorithme Antilles-Guyane septembre 2013

Partie A : Simulation

L'algorithme de l'énoncé se traduit presque en français. Pour faire court, on a effectué deux changements :

- Plutôt que $-1 \leq y \leq 1$, en écrivant $|y| \leq 1$, on économise un test d'inégalité ;
- l'objet **pas** est un tableau contenant tous les pas possibles, et on puise un élément au hasard dans ce tableau.

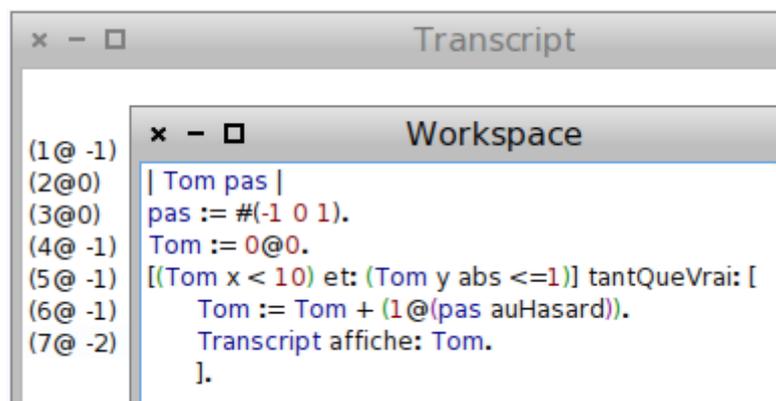
On obtient alors ceci :



```
Transcript
Tom est tombé

Workspace
| x y pas |
pas := #(-1 0 1).
x := y := 0.
[(x < 10) et: (y abs <= 1)] tantQueVrai: [
  x := x suivant.
  y := y + (pas auHasard).
].
((x = 10) et: (y abs <= 1)) siVrai: [Transcript affiche: 'Tom a réussi']
siFaux: [Transcript affiche: 'Tom est tombé'].
```

Mais on peut faire mieux, en modélisant la position de Tom par un point : Alors à chaque itération, on ajoute à Tom (qui est donc devenu un point), un vecteur d'abscisse 1 et d'ordonnée aléatoire. On y gagne en concision :



```
Transcript
(1@ -1)
(2@0)
(3@0)
(4@ -1)
(5@ -1)
(6@ -1)
(7@ -2)

Workspace
| Tom pas |
pas := #(-1 0 1).
Tom := 0@0.
[(Tom x < 10) et: (Tom y abs <=1)] tantQueVrai: [
  Tom := Tom + (1@(pas auHasard)).
  Transcript affiche: Tom.
].
```

Cette manière d'implémenter l'algorithme permet, en passant, de réviser les vecteurs (dont la syntaxe est similaire à celle des points, avec le symbole « @ » disposé entre l'abscisse et l'ordonnée).

L'algorithme de l'énoncé devient alors

```
Transcript
La position de Tom est (3@ -2)
Tom est tombé

Workspace
| Tom pas |
pas := #(-1 0 1).
Tom := 0@0.
[[Tom x < 10] et: (Tom y abs <= 1)] tantQueVrai: [
    Tom := Tom + (1@(pas auHasard)).
].
Transcript affiche: 'La position de Tom est ', (Tom commeTexte).
((Tom x = 10) et: (Tom y abs <= 1)) siVrai: [Transcript affiche: 'Tom a réussi']
siFaux: [Transcript affiche: 'Tom est tombé'].
```

Mais puisqu'on peut enregistrer une liste de positions successives¹ de Tom dans un tableau de points, on peut aussi dessiner un polygone² joignant ces points, et ceci, 10 fois pour se faire une idée statistique sur la question. Pour voir le dessin, on multiplie par 10 les positions successives, puis on leur ajoute (0 ; 10) pour centrer le polygone :

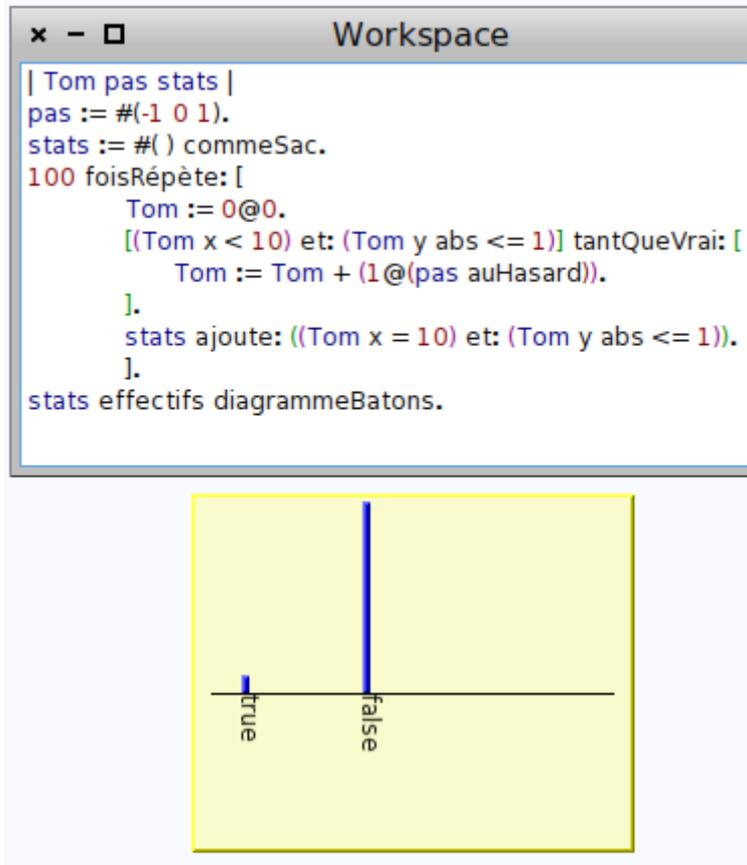
```
Workspace
| Tom pas dessin trajet tracé |
dessin := Morph new color: (Color cyan alpha: 0.02).
dessin extent: 100@20.
pas := #(-1 0 1).
100 foisRépète: [
    trajet := #() commeCollectionOrdonnée.
    Tom := 0@0.
    trajet ajoute: (10*Tom + (0@10)).
    [[Tom x < 10] et: (Tom y abs <=1)] tantQueVrai: [
        Tom := Tom + (1@(pas auHasard)).
        trajet ajoute: (10*Tom + (0@10) ).
    ].
    tracé := PolygonMorph new color: (Color yellow alpha: 0.2).
    tracé makeOpenOrClosed.
    tracé toggleSmoothing.
    tracé lineBorderColor: (Color blue alpha: 0.5).
    tracé borderWidth: 1.
    tracé setVertices: (trajet commeTableau).
    dessin addMorph: tracé.
].
dessin openInHand.
```



1 Cette liste s'appelle **trajet** dans le script ; en fait c'est une « collection ordonnée », convertie en tableau au moment du tracé
2 Appelé **tracé** dans le script

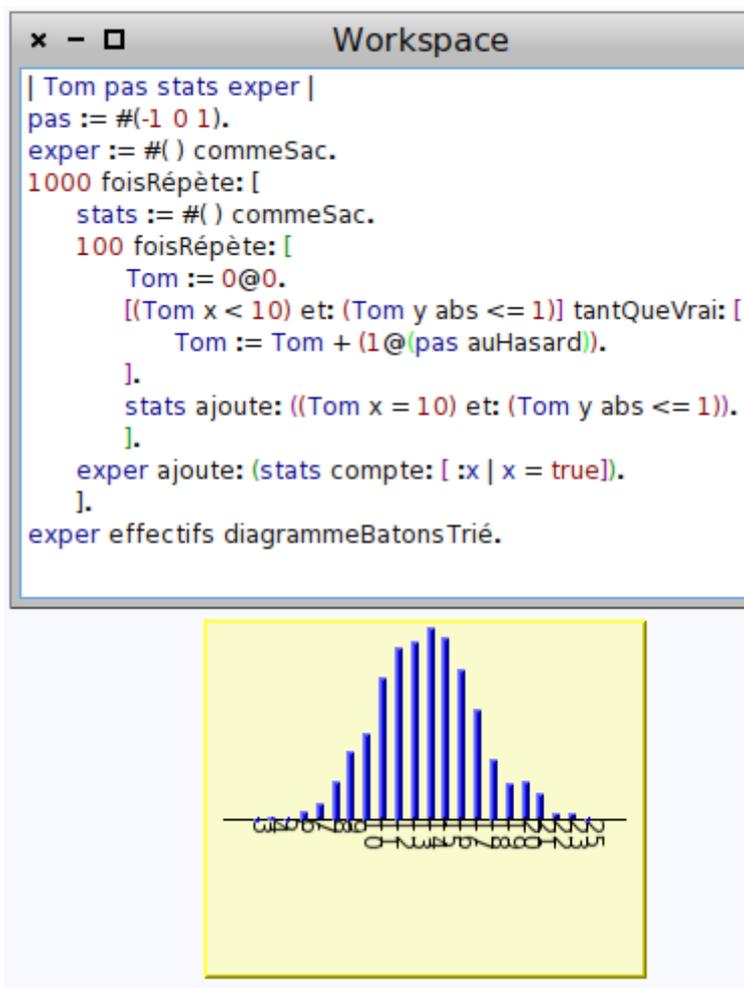
Statistiques

En fait, en enregistrant le fait que Tom a réussi ou non³ dans un « sac » appelé *stats*, on peut estimer la proportion de succès sur 100 tentatives de Tom, par exemple en dessinant le diagramme en bâtons :



³ *True* si Tom a réussi et *false* s'il a échoué ; on écrit juste *Tom x = 10* (comparaison de l'abscisse de Tom avec 10) pour cela.

En assimilant à une expérience les 100 tentatives de Tom, on peut rapidement simuler 1000 expériences⁴ et faire le diagramme en bâtons trié des nombres de succès :



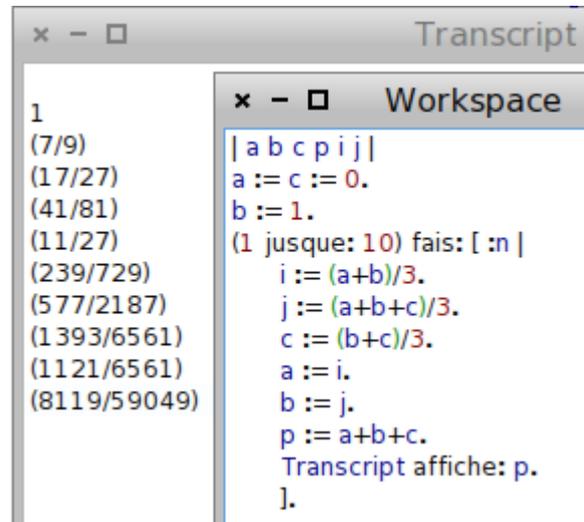
On constate alors que le nombre de succès sur 100 tentatives est binomial, de paramètres 100 et $\frac{8119}{59049}$ (probabilité que Tom réussisse sa première traversée ; voir partie B pour savoir comment a été calculée cette probabilité de succès).

4 Soit 100 000 tentatives...

Partie B : Calcul de la probabilité

MathsOntologie n'ayant pas de *tuple*, ni de matrices d'ordre 3, on va utiliser des variables auxiliaires i et j pour stocker provisoirement les nouvelles valeurs de a et b (soit $i \leftarrow a_{n+1}$ et $j \leftarrow b_{n+1}$).

Par contre on n'a pas besoin de variable supplémentaire pour $c_{n+1} = \frac{b_n + c_n}{3}$. Le script s'écrit alors⁵

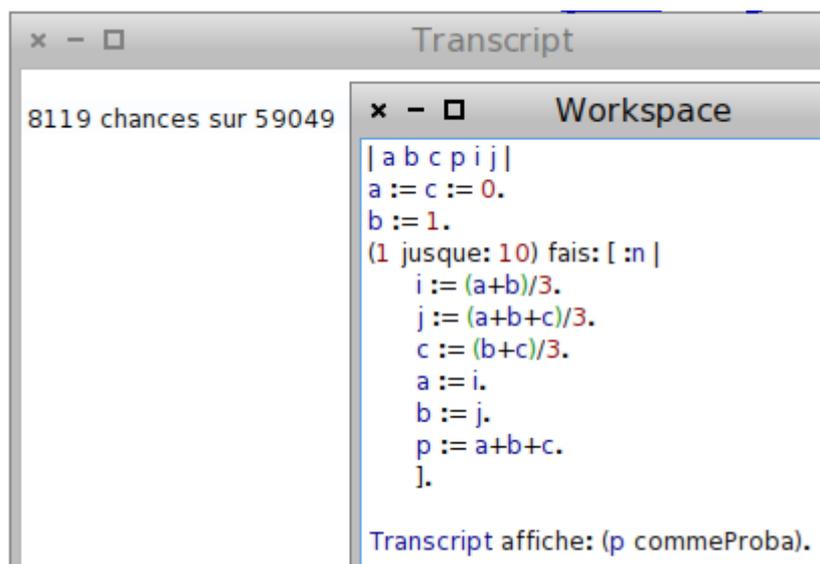


```
1
(7/9)
(17/27)
(41/81)
(11/27)
(239/729)
(577/2187)
(1393/6561)
(1121/6561)
(8119/59049)

| a b c p i j |
a := c := 0.
b := 1.
(1 jusque: 10) fais: [ :n |
  i := (a+b)/3.
  j := (a+b+c)/3.
  c := (b+c)/3.
  a := i.
  b := j.
  p := a+b+c.
  Transcript affiche: p.
].
```

On constate que, par défaut, les nombres de MathsOntologie sont des fractions, donc les probabilités sont données sous forme exacte, en fractions irréductibles. Si on avait ajouté *commeDécimal* après p , on aurait eu plus classiquement les valeurs approchées de l'énoncé.

Mais puisque MathsOntologie est fort en fractions, on peut avoir un affichage plus littéraire, en demandant l'affichage sous forme de probabilité :



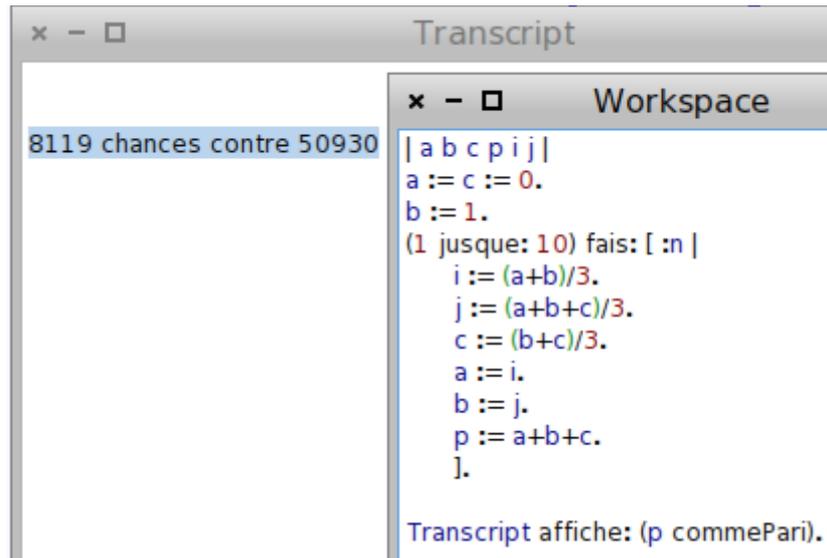
```
8119 chances sur 59049

| a b c p i j |
a := c := 0.
b := 1.
(1 jusque: 10) fais: [ :n |
  i := (a+b)/3.
  j := (a+b+c)/3.
  c := (b+c)/3.
  a := i.
  b := j.
  p := a+b+c.
  Transcript affiche: (p commeProba).
].
```

Tom a donc 8119 chances sur 59049 d'arriver au sec à la fin de l'expérience.

⁵ En tenant compte du fait que la probabilité $p_n = a_n + b_n + c_n$; ce sont ces probabilités qui sont affichées

Et si on lançait des paris sur la victoire de Tom ?



On parierait donc à 8119 contre 50930, ; autrement dit, si je parie 81,19 € sur l'arrivée de Tom au sec, et que la banque me donne 509,30 € en cas de chute dans l'eau, le pari est équitable...

Si on généralise à n pas, les suites a_n , b_n et c_n ont une forme explicite⁶ :

- $a_n = \sqrt{2} \frac{(1+\sqrt{2})^n - (1-\sqrt{2})^n}{4 \times 3^n}$ (oui, c'est une fraction !)
- $a_n = \frac{(1+\sqrt{2})^n - (1-\sqrt{2})^n}{2 \times 3^n}$
- $c_n = \sqrt{2} \frac{(1+\sqrt{2})^n - (1-\sqrt{2})^n}{4 \times 3^n}$

D'où la probabilité $p_n = \frac{(1+\sqrt{2})^{n+1} - (1-\sqrt{2})^{n-1}}{2 \times 3^n}$.

Cette probabilité, étant somme de deux suites géométriques de raison inférieure à 1, tend vers 0. On remarque que c'est à partir de 5 pas que la probabilité d'arriver au sec devient inférieure à 0,5 pour Tom (dans le tableau des valeurs successives de p ci-avant).

Alain Busser
LPO Roland-Garros
Le Tampon

⁶ Trouvées avec Xcas, en diagonalisant $\frac{1}{3} \begin{pmatrix} 110 \\ 111 \\ 011 \end{pmatrix}$ pour trouver l'expression explicite de ses puissances