

Une séance d'algorithmique en T^{le} S (durée : 2h)

J'ai mis longtemps à me décider à traiter un exercice sur les chemins aléatoires... en voici un qui mélange, algorithmique, simulation, probabilités et suites... tout ça avec beaucoup de **Larp** et un peu de **tableur**.

Antilles-Guyane Septembre 2013

EXERCICE 4

5 points

Candidats n'ayant pas suivi l'enseignement de spécialité

Les deux parties sont indépendantes

Le robot Tom doit emprunter un pont sans garde-corps de 10 pas de long et de 2 pas de large. Sa démarche est très particulière :

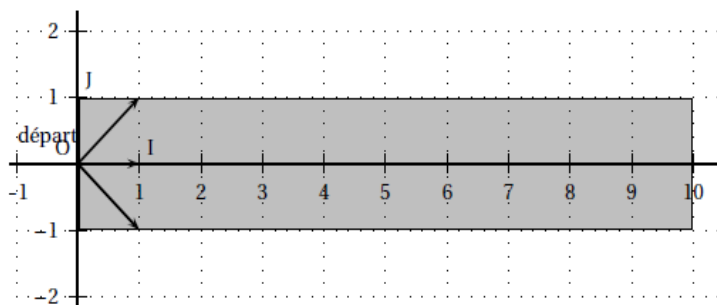
- Soit il avance d'un pas tout droit ;
- Soit il se déplace en diagonale vers la gauche (déplacement équivalent à un pas vers la gauche et un pas tout droit) ;
- Soit il se déplace en diagonale vers la droite (déplacement équivalent à un pas vers la droite et un pas tout droit).

On suppose que ces trois types de déplacement sont aléatoires et équiprobables.

L'objectif de cet exercice est (d'estimer) la probabilité p de l'évènement S « Tom traverse le pont » c'est-à-dire « Tom n'est pas tombé dans l'eau et se trouve encore sur le pont au bout de 10 déplacements ».

Partie A : modélisation et simulation

On schématise le pont par un rectangle dans le plan muni d'un repère orthonormé $(0, I, J)$ comme l'indique la figure ci-dessous. On suppose que Tom se trouve au point de coordonnées $(0; 0)$ au début de la traversée. On note $(x; y)$ les coordonnées de la position de Tom après x déplacements.



On a écrit l'algorithme suivant qui simule la position de Tom au bout de x déplacements :

```
x, y, n sont des entiers
Affecter à x la valeur 0
Affecter à y la valeur 0
Tant que y ≥ -1 et y ≤ 1 et x ≤ 9
    Affecter à n une valeur choisie au hasard entre -1, 0 et 1
    Affecter à y la valeur y + n
    Affecter à x la valeur x + 1
Fin tant que
Afficher « la position de Tom est » (x ; y)
```

1. On donne les couples suivants : $(-1; 1)$; $(10; 0)$; $(2; 4)$; $(10; 2)$.
Lesquels ont pu être obtenus avec cet algorithme ? Justifier la réponse.
2. Modifier cet algorithme pour qu'à la place de « la position de Tom est $(x; y)$ », il affiche finalement « Tom a réussi la traversée » ou « Tom est tombé ».

Un premier commentaire

Derrière « *estimer la proba* », moi j'entends « simuler un grand nombre de traversées et faire jouer la loi des grands nombres... » ! Or c'est la simulation d'une seule traversée qui est proposée...

Certes, le calcul sur tableur, dans la partie B, ne fournira qu'une valeur approchée de cette proba... (ce qui explique ici le mot *estimer* certainement)...

Algo1 : l'original... version Organigramme et version Pseudo-code

The screenshot shows the LARP software interface. On the left, there is a 'Navigateur' pane showing the project structure and a 'Modèles' pane with various flowchart symbols. The main workspace displays a flowchart for the 'PRINCIPAL' module. The flowchart starts with 'x=0', followed by 'y=0'. It then enters a loop with the condition 'y >= -1 et y <= 1 et x <= 9'. Inside the loop, the steps are 'n = alea(-1,1)', 'y = y+n', and 'x = x+1'. After the loop, there is an output box: 'ÉCRIRE "La position de Tom est : (" , x , " , " , y , ")"'. A 'Pseudo-code de PRINCIPAL' window is open on the right, showing the following code:

```
DÉBUT
x=0
y=0
TANTQUE y>=-1 et y<=1 et x<=9 FAIRE
  n=alea(-1,1)
  y=y+n
  x=x+1
FIN TANTQUE
ÉCRIRE "La position de Tom est : (" , x , " , " , y , " )"
```

The taskbar at the bottom shows the Windows Start button, 'Document1 - Microsof...', 'AlgoTice', and 'LARP'.

Une exécution (voir console)...

This screenshot shows the same LARP software interface as above, but with the flowchart in a greyed-out state, indicating it is being executed. A 'Console' window is open on the right, displaying the following output:

```
Terminé
La position de Tom est : ( 10 , 0 )
Appuyez sur une touche pour fermer la console...
```

The taskbar at the bottom shows the Windows Start button, 'Document1 - Microsof...', 'AlgoTice', and 'LARP'.

Une autre exécution...

The screenshot shows the LARP (Logiciel de Apprentissage de la Programmation) interface. The main workspace displays a flowchart for a program. The flowchart starts with an initialization block $x=0$, followed by $y=0$. It then enters a loop with the condition $y \geq -1$ et $y \leq 1$ et $x \leq 9$. If the condition is true (Vrai), it executes $n = \text{alea}(-1, 1)$, then $y = y + n$, and $x = x + 1$. If the condition is false (Faux), it exits the loop. The final output is $\text{ÉCRIRE "La position de Tom est : (" , x , " , " , y , ")"}$. A console window in the foreground shows the execution result: "Terminé", "La position de Tom est : (7 , 2)", and "Appuyez sur une touche pour fermer la console...".

☞ **Algo2** : avec la modif. demandée dans l'énoncé (question 2.), pour l'affichage...

On décide de garder l'affichage (d'origine) de la position, pour vérif.
Le rajout est en Boucle conditionnelle SI-SINON

The screenshot shows the LARP interface with a modified flowchart. It starts with the output $\text{ÉCRIRE "La position de Tom est : (" , x , " , " , y , ")"}$. This is followed by a decision diamond with the condition $y \geq -1$ et $y \leq 1$. If the condition is true (Vrai), the flowchart outputs $\text{ÉCRIRE "Tom a réussi la traversée"}$. If the condition is false (Faux), it outputs $\text{ÉCRIRE "Tom est tombé à l'eau"}$. Both paths lead to a final FIN block.

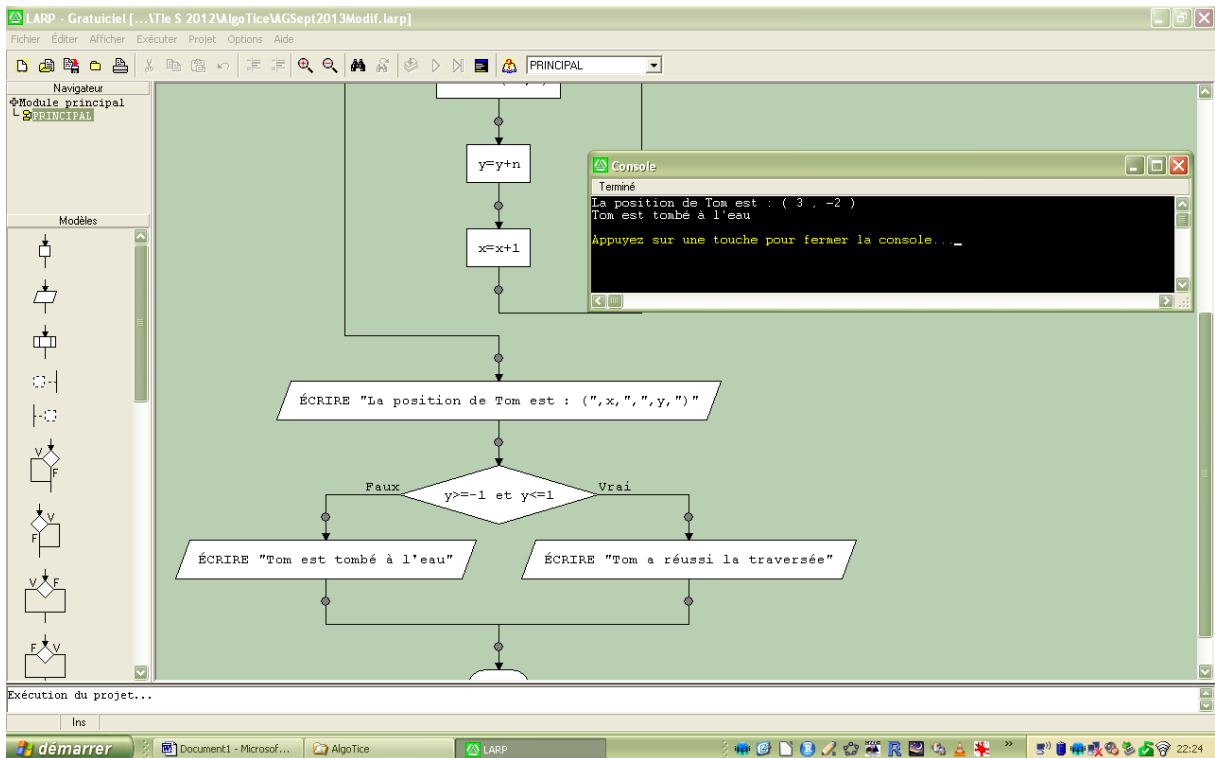
Ce qui donne en Pseudo-Code

```
Pseudo-code de PRINCIPAL

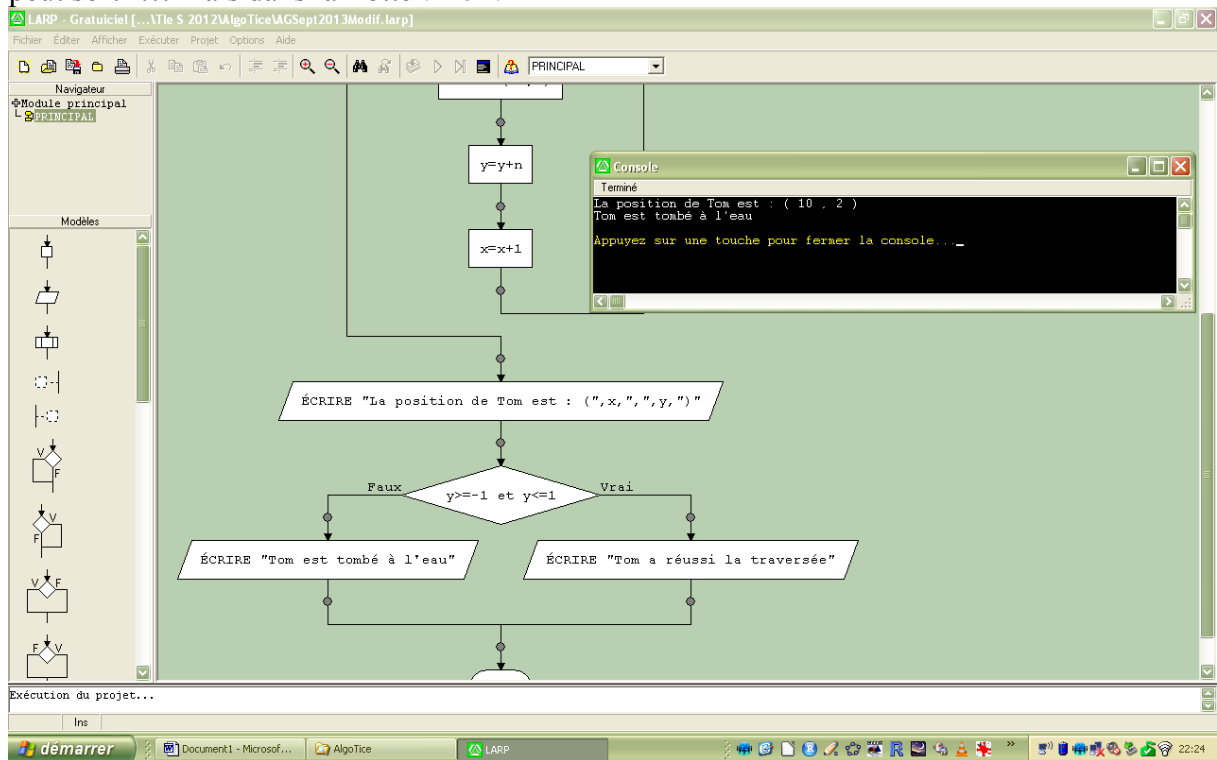
DÉBUT
  x=0
  y=0
  TANTQUE y>=-1 et y<=1 et x<=9 FAIRE
    n=alea(-1,1)
    y=y+n
    x=x+1
  FINTANTQUE
  ÉCRIRE "La position de Tom est : (",x,",",y,")"
  SI y>=-1 et y<=1 ALORS
    ÉCRIRE "Tom a réussi la traversée"
  SINON
    ÉCRIRE "Tom est tombé à l'eau"
  FINSI
FIN
```

Copier au presse-papiers Aide Fermer

Une exécution...



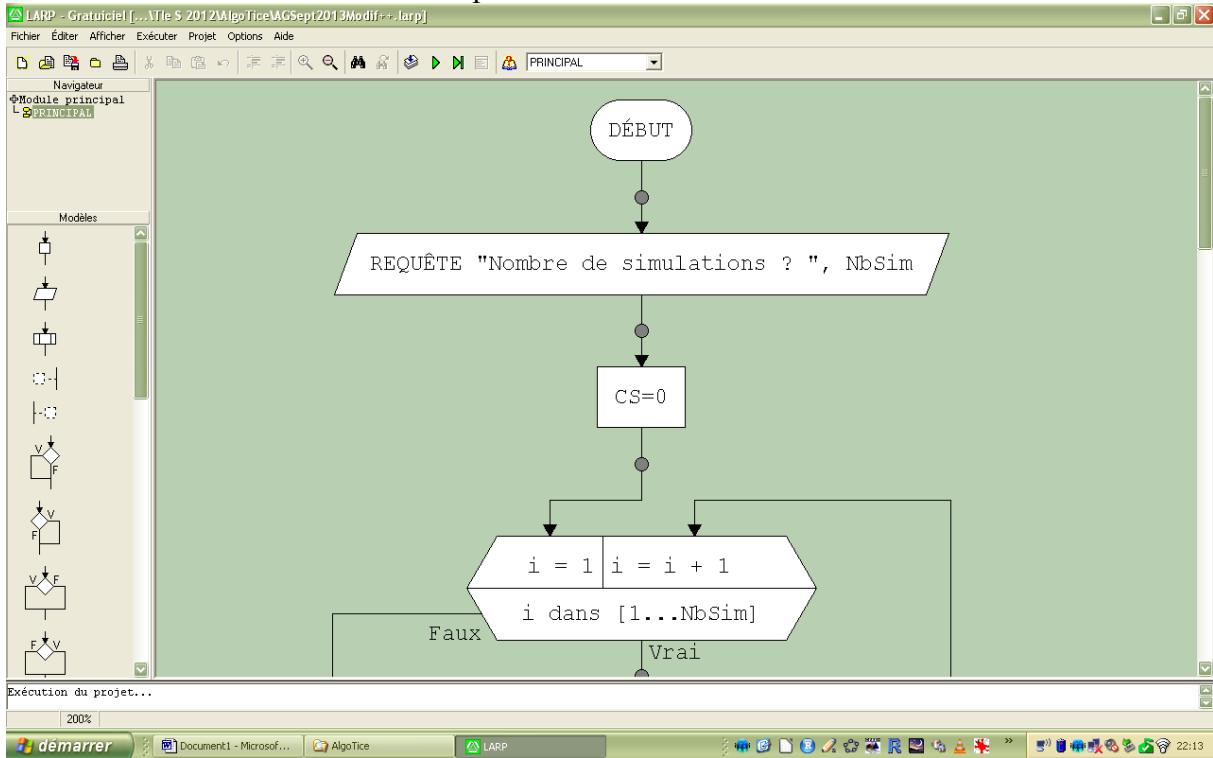
Une autre exécution... : on obtient l'un des 4 couples proposés par l'énoncé... qui peut sortir... mais dans la flotte ! Lol !



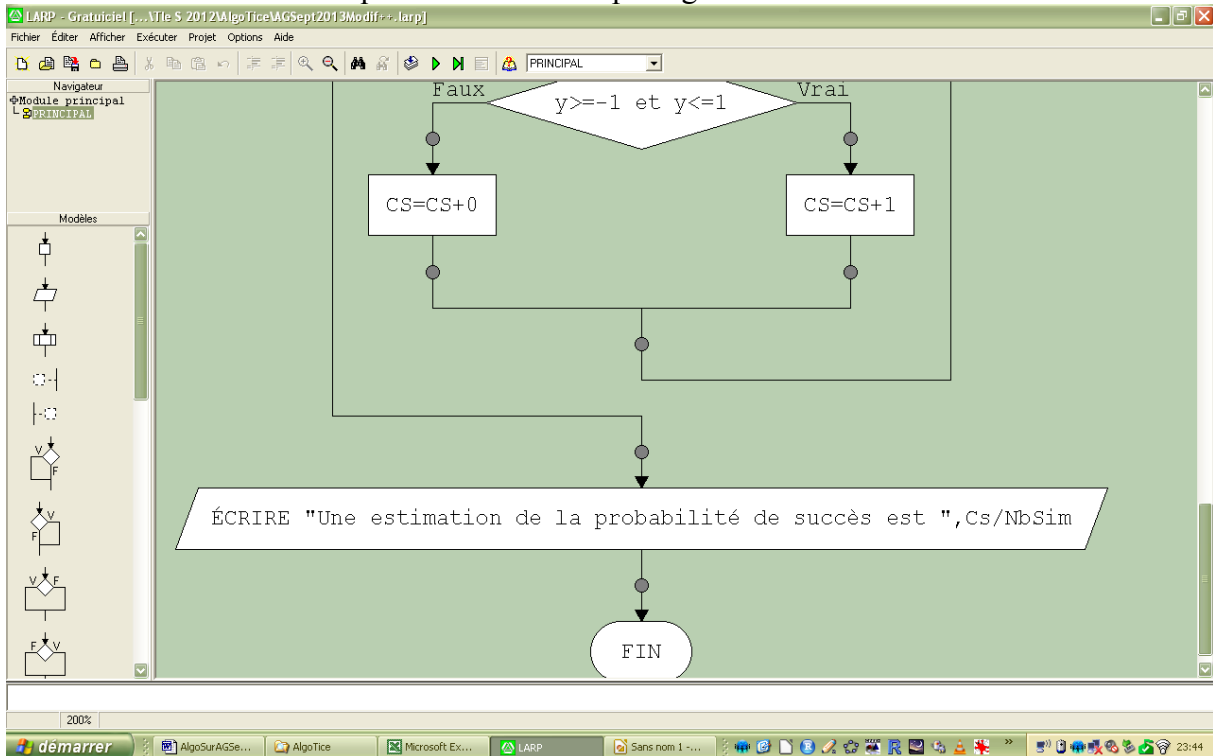
Je propose ci-dessous de simuler un grand nombre de traversées afin justement d'estimer la probabilité recherchée...

☞ **Algo3** : Les NbSim simulations dans une boucle POUR, pour pouvoir estimer la probabilité de succès (Tom réussit sa traversée !),

avec un compteur CS de succès initialisé à 0...



et qui s'incrémente au passage d'un succès...



La loi des grands nombres parachève le travail...

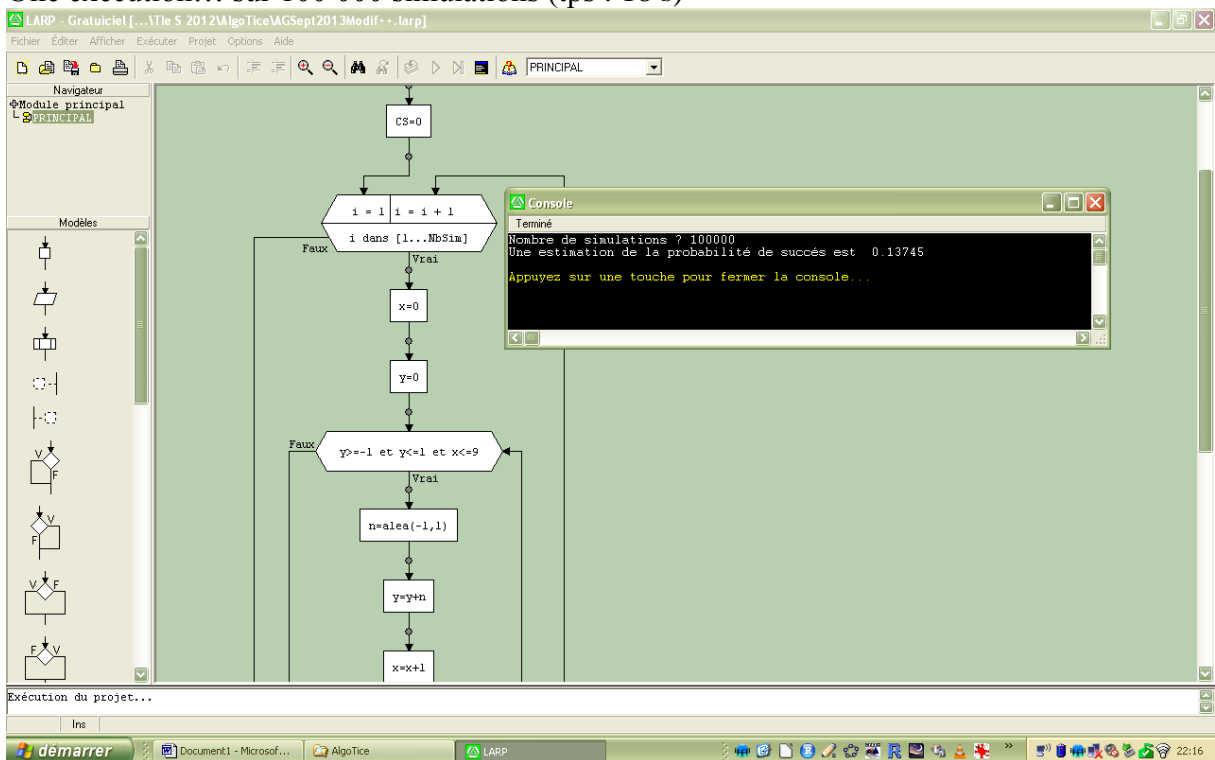
Ce qui donne en Pseudo-Code

```
Pseudo-code de PRINCIPAL

DÉBUT
  REQUÊTE "Nombre de simulations ? ", NbSim
  CS=0
  POUR i = 1 JUSQU'À NbSim INCRÉMENT 1 FAIRE
    x=0
    y=0
    TANTQUE y>=-1 et y<=1 et x<=9 FAIRE
      n=alea(-1,1)
      y=y+n
      x=x+1
    FINTANTQUE
    SI y>=-1 et y<=1 ALORS
      CS=CS+1
    SINON
      CS=CS+0
    FINSI
  FINPOUR
  ÉCRIRE "Une estimation de la probabilité de succès est ",Cs/NbSim
FIN
```

Copier au presse-papiers Aide Fermer

Une exécution... sur 100 000 simulations (tps : 18 s)



Une autre exécution... sur 1 000 000 de simulations (tps : 2 min 57 s)

The screenshot shows the LARP software interface. The main window displays a flowchart for a simulation. The flowchart starts with a box 'CS=0', followed by a loop 'i = 1 i = i + 1' and a decision 'i dans [1...NbSim]'. If 'Vrai', it goes to 'x=0', 'y=0', and a decision 'y >= -1 et y <= 1 et x <= 9'. If 'Vrai', it goes to 'n = alea(-1,1)', 'y = y + n', and 'x = x + 1'. If 'Faux', it loops back to the decision 'i dans [1...NbSim]'. A console window is open over the flowchart, displaying the following text:

```

Terminé
Nombre de simulations ? 1000000
Une estimation de la probabilité de succès est 0.137818
Appuyez sur une touche pour fermer la console...
    
```

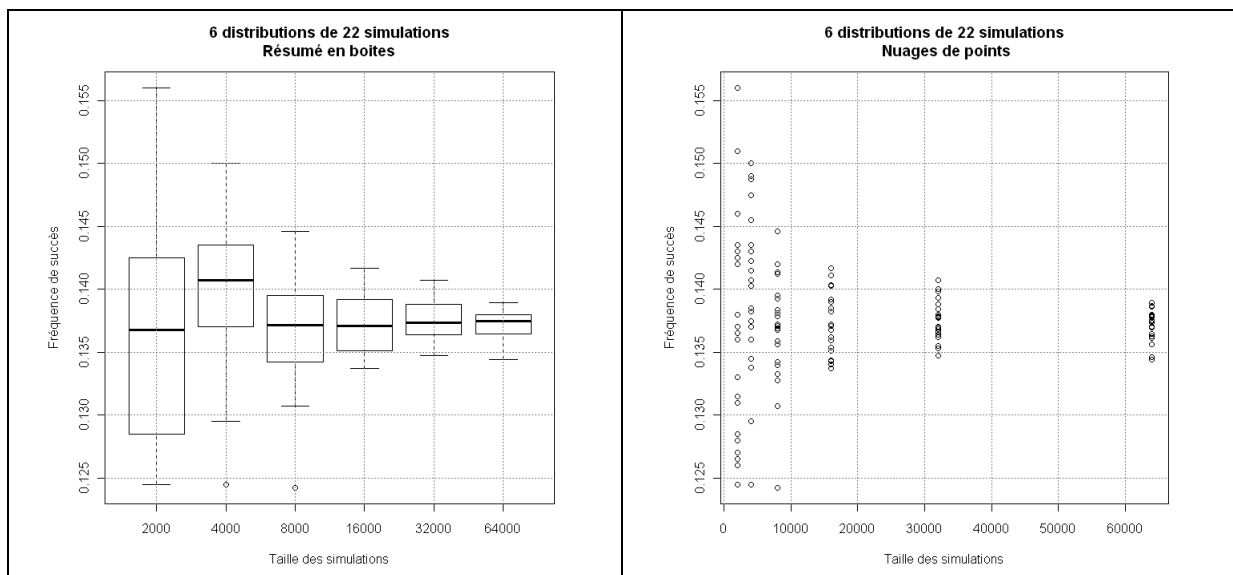
The taskbar at the bottom shows the Windows Start button, a document icon, 'AlgoTice', and the LARP application icon. The system tray shows the time as 22:20.

⋮

Et voilà ce que l'on peut faire observer...

Chacun des 22 élèves ayant effectué ses 6 simulations (d'ordre 2 000, 4 000, 8 000, 16 000, 32 000 et 64 000)...

Une belle illustration de **la loi des grands nombres...** : convergence "en faisceau".



☞ On peut maintenant passer au **modèle...**(Partie B)

On suit l'indication de l'énoncé, en faisant le tronçon d'arbre pondéré de l'étape n à l'étape $n+1$, et les relations de récurrences sortent facilement...

On sort un bon tableur, et voilà le tableau d'avancement des trois suites :

	A	B	C	D	E	F
1	n	an	bn	cn		
2	0	0	1	0		
3	1	0,33333333	0,33333333	0,33333333		
4	2	0,22222222	0,33333333	0,22222222		
5	3	0,18518519	0,25925926	0,18518519		
6	4	0,14814815	0,20987654	0,14814815		
7	5	0,11934156	0,16872428	0,11934156		
8	6	0,09602195	0,13580247	0,09602195		
9	7	0,07727481	0,10928212	0,07727481		
10	8	0,06218564	0,08794391	0,06218564		
11	9	0,05004318	0,07077173	0,05004318		
12	10	0,04027164	0,0569527	0,04027164	0,137495978	
13						
14						

La valeur de la probabilité cherchée est calculée en **E12** !