

# La géolocalisation

Élèves :

FARGEAU Lucille

JERRETIE Marine

DOUENEAU Gaëtan

MENAGER Hugo

TROUSSEAU Antoine

Enseignant :

SANS Nicolas

Chercheur :

GAMBS Sébastien

Lycée Benjamin Franklin AURAY

Année 2011/2012

## Introduction

Notre projet est né d'une volonté d'approfondir nos connaissances en mathématiques, plus particulièrement nous voulions aller plus loin que ce qui a été appris en cours. C'est dans cet état d'esprit que notre groupe du lycée Benjamin Franklin d'Auray s'est engagé dans l'expérience MATH.en.JEANS (MeJ). Il s'agit du premier groupe MeJ créé dans le Morbihan et ceci grâce à Mr. Sans, qui a pris ce projet très à cœur. Nos recherches furent effectuées en collaboration avec l'équipe du lycée Victor Hélène Bash de Rennes supervisée par leur professeure, Mme Forgeoux. C'est aussi grâce à Sébastien Gambis, chercheur à l'INRIA et maître de conférences à l'université Rennes 1, que nous avons avancé dans notre travail, ce dernier nous ayant proposé le sujet de recherche suivant :

### How to define and quantify the concepts of "anonymity" and "privacy"?

(1)

Information Technologies have invaded many aspects of people's daily lives, creating new possibilities but also raising privacy concerns to the point that some individuals feel that they no longer have suitable guarantees or control of their privacy. Indeed, protecting the privacy of individuals is one of the main challenges of the «Information Society» but is difficult to achieve as individuals constantly leave digital traces of their actions and whereabouts, often without even knowing it. If an unauthorized entity gathers these digital traces, he (or she) can use them for malicious purposes ranging from targeted spam to profiling, and even identity theft.

In general, simply removing the identifiers of individuals or replacing them by a pseudonym is usually not sufficient to protect their privacy. Therefore privacy researchers have developed more sophisticated techniques for protecting privacy, such as privacy-enhancing technologies (PETs) or sanitization methods which add uncertainty to the data and remove some sensible information. However, in order to be able to assess the effectiveness of such methods and to compare them, it is necessary to:

- (A) have a precise and meaningful definition of the concept of "anonymity" and "privacy" adapted to the specific context considered and
- (B) practical methods to quantify the level of privacy achieved by a particular service or architecture.

While it is very difficult (and maybe even impossible) to come with a "universal" definition of privacy, it is possible to derive precise definitions when considering a particular context. In this project, the students will explore how to address the two fundamental questions (A) and (B) in one of the following contexts:

1. anonymous communication networks.
2. social networking sites.
3. geolocated applications.
4. electronic voting.
5. public release of sanitized data (e.g. medical or census data).

A partir de notre problématique, nous avons réfléchi à plusieurs thèmes potentiels. Nous avons ainsi travaillé sur le vote électronique, la protection des données médicales, les réseaux de communication, les réseaux sociaux et la géolocalisation. C'est finalement ce dernier thème qui a retenu notre attention, en faisant ainsi notre sujet principal.

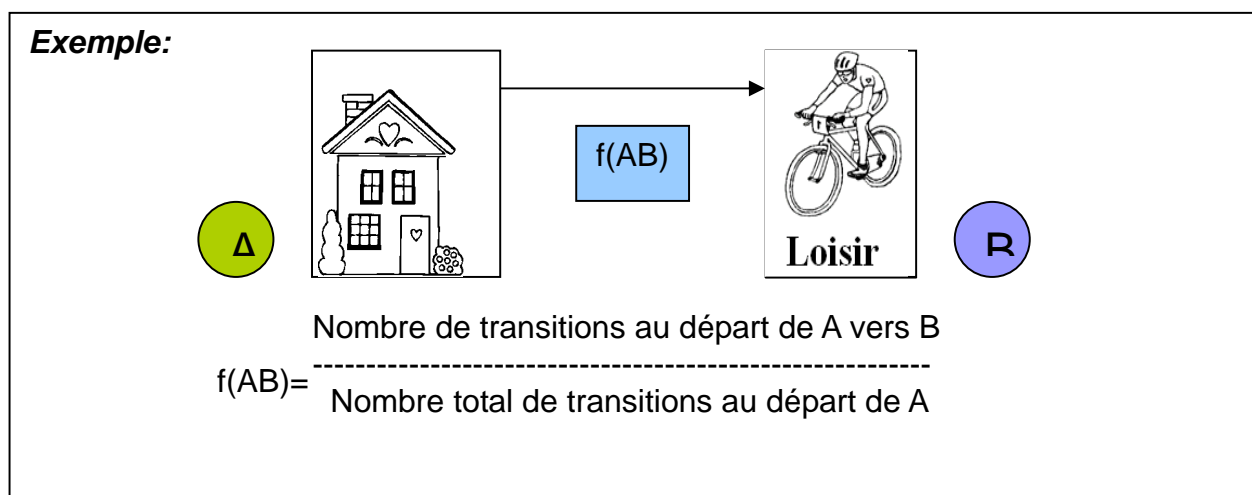
## I Qu'est-ce que la géolocalisation?

La géolocalisation est un procédé permettant de situer un individu ou un objet sur une carte grâce à ses coordonnées géographiques (par exemple la localisation d'un salarié par son entreprise). Il existe plusieurs techniques de géolocalisation :

- À l'aide de satellites. Le GPS (Global Positioning System) installé dans un équipement mobile effectue une triangulation afin de mesurer la distance entre l'utilisateur et un certain nombre de satellites de coordonnées connues.
- Avec un téléphone portable. On utilise le réseau GSM (téléphonique) en utilisant les coordonnées des antennes relais.
- Par les ondes Wifi. L'individu connecté par un dispositif numérique (par exemple ordinateur, tablette, téléphone) peut être repéré par son adresse MAC et l'identifiant de cette borne wifi.
- Par l'adresse IP. Chaque ordinateur ou dispositif connecté à internet possède une adresse IP propre. Celles-ci sont répertoriées par l'IANA (Internet Assigned Numbers Authority) et donc repérables.

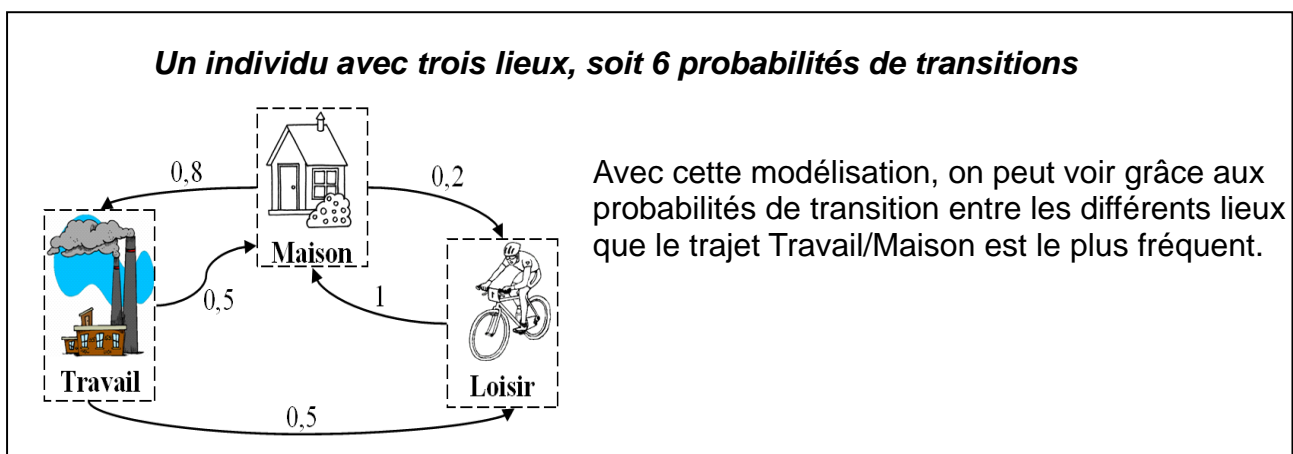
### La chaîne de mobilité de Markov pour localiser: un danger pour la vie privée

La chaîne de mobilité de Markov est un modèle mathématique permettant à partir de la localisation actuelle ou passée d'un individu de prédire sa localisation future. Appliqué à la géolocalisation, c'est un véritable danger pour la vie privée. Ainsi, en observant les déplacements d'un individu, on peut établir des fréquences qui détermineront des probabilités de transitions entre les lieux qu'il visite fréquemment. (2)



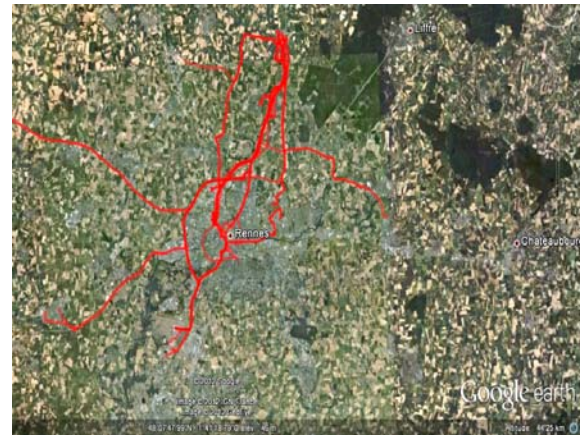
Afin de mesurer les atteintes à la vie privée possible, nous avons effectué deux expériences avec des relevés de trajets d'un élève.

#### **Première modélisation :**



## Deuxième modélisation avec un individu ayant une multitude de déplacements

Cette seconde expérience consistait à enregistrer les trajets par un logiciel gratuit sur smartphone disponible sous Android. Elle a permis d'augmenter le nombre de lieux (car l'élève était plus actif) et donc le nombre de possibilités transitionnelles.



*Modélisation des trajets pendant une semaine*

	A	B	C	D	E	F	H	I	J	K	L	M	N	O	P	Q
A	0	0,531	0	0,178	0,059	0	0,029	0,029	0,029	0	0,029	0,029	0,029	0,029	0	0,029
B	0,6	0	0,3	0	0	0,05	0	0	0	0,05	0	0	0	0	0	0
C	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
D	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
E	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
H	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
I	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
J	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
K	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
L	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
M	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
N	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
O	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
P	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Q	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Grâce à cette modélisation, on peut voir que les lieux A et B sont à l'origine des transitions les plus courantes. On peut donc penser que ces deux endroits sont le lieu d'habitation et le lieu de travail de l'individu.

(3)

## II/ Le jeu des transitions

Afin de sensibiliser au fait que la prévision des déplacements d'une personne est une tâche aisée, nous avons eu l'idée de créer un jeu avec Albox (un logiciel d'aide à l'élaboration et à l'exécution

d'algorithmes). Dans ce jeu, après observation des transitions entre les lieux, le joueur est invité de manière ludique à prévoir les déplacements d'un individu.

### 1/ Synopsis

« Vous êtes l'inspecteur Eve. Votre nouvelle mission : stopper Bob, dangereux criminel, qui prépare un complot terroriste aboutissant dans 3 semaines. Vous disposez pour cela de plusieurs informateurs qui vous indiquent le nombre de passages entre les différents lieux (A,B...) que fréquente Bob ainsi que de la magie des maths! Chaque semaine, ils rassemblent suffisamment d'informations pour réussir à l'arrêter, au bout de cette semaine, vous pouvez choisir le lieu où il va probablement se trouver et y envoyer vos policiers. Si vous ne réussissez pas au bout de 3 semaines (= 3 essais), vous êtes renvoyé et la partie se termine. »

Ainsi, il existe dans le jeu quatre niveaux (2 lieux, 3 lieux, 4 lieux, 5 lieux) et pour chacun d'entre eux le joueur dispose de 3 essais. S'il réussit avant la fin des 3 tentatives, il passe au niveau suivant.

### 2/ Fonctionnement

Le principe général est simple : l'algorithme crée d'abord une chaîne avec des probabilités de transition prises de manière aléatoire (c'est-à-dire sans regarder les traces de mobilité). Il choisit un lieu de départ puis va d'un lieu à un autre en suivant les probabilités. Pour renseigner le joueur, le programme affiche le nombre de transitions différentes effectuées. Il choisit alors un lieu au hasard, qui sera le départ de Bob et l'affiche au joueur. Par simple observation du nombre de transitions entre les lieux, le joueur choisit alors le lieu qu'il juge être le suivant. L'algorithme va alors s'orienter du lieu qu'il a affiché vers un autre en respectant les probabilités. Si le lieu choisi par l'algorithme (par Bob) est le même que celui choisi par le joueur, alors on considère que le joueur a pu coincer Bob.

(4)

*Voir le code en langage AlgoBox en annexe 2*

Affichage à 4 lieux :

NIVEAU III :

Cette semaine, Bob a fréquenté les lieux A, B, C et D dont voici la liste des transitions avec le nombre de transitions observées pour chacun d'entre elles :

AB : 5

AC : 38

AD : 14

BA : 1

BC : 8

BD : 3

CA : 31

CB : 6

CD : 9

DA : 25

DB : 1

DC : 0

Bob est maintenant parti de A. Inspecteur, vous devez décider où envoyer vos effectifs. Par observation des transitions, le joueur va sûrement choisir le lieu C, qui est celui vers lequel Bob (et donc l'algorithme) ira le plus probablement.

### 3/ Quel intérêt ?

Quelque soit le niveau du joueur (collégien ou lycéen), il est en mesure de déterminer simplement le lieu où Bob ira probablement, même sans avoir connaissance de la théorie de probabilités.

Il peut donc comprendre :

- que certaines transitions sont plus fréquentes que d'autres.

(5)

- qu'il est assez facile prévoir les déplacements de quelqu'un et donc de les devancer.

De plus, même en misant sur le lieu le plus probable, il n'est pas toujours sûr de choisir le bon : ce modèle de transitions (donc la chaîne de Markov) manque parfois de précision, car elle garde une notion de l'aléatoire, que ne conserverait peut-être pas une chaîne horaire (en tenant compte du temps passé dans chaque lieu).

(6)

### III/ Le poids transitionnel

Par l'observation simple des déplacements qu'effectue une personne, nous souhaitons créer un indicateur de l'importance relative des lieux. Cette importance est dite **poids transitionnel**, et ne dépend que des transitions effectuées entre les lieux et non du temps passé à chaque lieu, ce qui sera une limite de notre modèle.

Considérons dans cette partie une chaîne de trois lieux A, B et C.  $p(AB)$  est la probabilité qu'en étant en A Bob se déplace ensuite vers B, etc. Cette chaîne donne, sous forme de matrice (7) :

$$\begin{pmatrix} 0 & p(AB) & p(AC) \\ p(BA) & 0 & p(BC) \\ p(CA) & p(CB) & 0 \end{pmatrix}$$

Nous avons bien sûr des 0 pour  $p(AA)$ ,  $p(BB)$ ... car nous ne considérons que les transitions entre des lieux différents (c'est-à-dire que les transitions d'un lieu vers lui-même sont impossibles).

#### 1/ Pourquoi privilégier les probabilités de transitions ?

Une matrice de transitions indique des probabilités que la personne étudiée, étant à un lieu, parte vers un autre. De fait, on peut avoir  $p(AB) = 0.5$  et  $p(CA) = 0.5$  ; mais cela ne garantit pas qu'elle effectue autant de transitions de A vers B que de C vers A, bien au contraire. Nous nous sommes alors demandé quel était l'intérêt d'une matrice de probabilités de transitions, qui semblait contenir moins d'informations (notamment pour calculer le poids des lieux) qu'une matrice nous donnant le nombre exact de transitions effectuées sur une période donnée.

Une réponse simple : la matrice de probabilités de transitions a la somme des valeurs de ses lignes qui vaut 1, elle est dite stochastique et possède donc des propriétés particulières (voir section 4). (8)

Nous avons alors cherché à démontrer que cette matrice contenait autant d'informations qu'une matrice donnant le nombre exact de transitions.

A partir de la matrice de probabilités exemple :

$$\begin{cases} p(AB) + p(AC) = 1 \\ p(BA) + p(BC) = 1 \\ p(CA) + p(CB) = 1 \end{cases}$$

On souhaite trouver une matrice de nombre de transitions correspondante. Les nombres de transitions sont donc proportionnels aux probabilités de transitions, au départ d'un même lieu : par ligne. Soit  $n1$  le facteur multipliant la première ligne,  $n2$  la seconde et  $n3$  la troisième, les matrices donnant le nombre de transitions sont donc de la forme :

$$\begin{pmatrix} 0 & n1 \times p(AB) & n1 \times p(AC) \\ n2 \times p(BA) & 0 & n2 \times p(BC) \\ n3 \times p(CA) & n3 \times p(CB) & 0 \end{pmatrix}$$

Si les facteurs  $n1$ ,  $n2$  et  $n3$  sont indépendants les uns des autres, alors cette seconde matrice contient plus d'informations que celle des probabilités. Démontrons le contraire.

$$\begin{cases} n1 \times (p(AB) + p(AC)) = n1 \\ n2 \times (p(BA) + p(BC)) = n2 \\ n3 \times (p(CA) + p(CB)) = n3 \end{cases} \Leftrightarrow \begin{cases} n1 \times p(AB) + n1 \times p(AC) = n1 \\ n2 \times p(BA) + n2 \times p(BC) = n2 \\ n3 \times p(CA) + n3 \times p(CB) = n3 \end{cases}$$

Mais il faut prendre en compte le fait qu'il y ait autant de transitions (ou presque) au départ de chaque point d'intérêt qu'à l'arrivée sur ce point :

$$\begin{cases} n1 \times p(AB) + n1 \times p(AC) = n2 \times p(BA) + n3 \times p(CA) \\ n2 \times p(BA) + n2 \times p(BC) = n1 \times p(AB) + n3 \times p(CB) \\ n3 \times p(CA) + n3 \times p(CB) = n1 \times p(AC) + n2 \times p(BC) \end{cases} \Leftrightarrow \begin{cases} n1 = n2 \times p(BA) + n3 \times p(CA) \\ n2 = n1 \times p(AB) + n3 \times p(CB) \\ n3 = n1 \times p(AC) + n2 \times p(BC) \end{cases}$$

En connaissant les  $p(AB)$ ,  $p(AC)$ ,  $p(BA)$ ... notre travail revient à résoudre un système de 3 équations à 3 inconnues chacune. Nous pouvons donc exprimer  $n2$  en fonction de  $n1$  ; et  $n3$  en fonction de  $n1$  (et des  $p(AB)$ ... bien sûr) :

$$n2 = \frac{n1(p(AB) + p(AC) \times p(CB))}{1 - p(BC) \times p(CB)}$$

$$n3 = n1 \times p(AC) + \frac{n1 \times p(BC)(p(AB) + p(AC) \times p(CB))}{1 - p(BC) \times p(CB)}$$

Donc les 3 facteurs ne sont pas indépendants les uns des autres. Il existe une infinité de matrices donnant le nombre de transitions correspondantes à une matrice de probabilités, mais dès lors que l'on fixe  $n1$ , alors  $n2$  et  $n3$  sont aussi fixés : cette matrice n'apporte rien de plus que celle des probabilités (que nous nommerons maintenant simplement matrice de transitions).

## 2/ Méthode algorithmique

La chaîne de mobilité de Markov peut-être considérée comme une loi de probabilité des déplacements. Notre idée a alors été de créer un algorithme qui respecterait cette loi de probabilité pour passer d'un lieu à un autre. En expérimentant un très grand nombre de fois, il suit la loi des grands nombres et nous donne la fréquence d'apparition de chaque lieu : son **poids transitionnel**.

### Résumé :

Lecture les probabilités de transition

Choix d'un lieu de départ au hasard

Pour  $i$  allant de 1 à 100000 :

- Passage d'un lieu vers un autre en respectant les probabilités de transition
- Comptage du nombre de passages à chaque lieu avec 3 variables

Fin pour

Affichage des fréquences d'apparition de chaque lieu

*Voir le code en langage AlgoBox en annexe 1*

Cette technique semble précise mais l'algorithme est long à répéter la boucle 100000 fois. De plus cette méthode n'est pas purement calculatoire car elle passe par de l'**échantillonnage**.

Cet algorithme est beaucoup plus long à programmer quand on augmente le nombre de lieux : cette technique ne peut plus être utilisée pour nous.

## 3/ Système d'équations et vecteur stable

### a) Système

Pour évaluer l'importance de chaque lieu ( $a$ ,  $b$  et  $c$  respectivement pour  $A$ ,  $B$  et  $C$ ) nous avons pensé à faire la somme des probabilités qui se dirigent vers ce lieu. Ainsi :  $a = p(BA) + p(CA)$ . Mais cette technique semble fort étrange car  $B$  et  $C$  n'ont peut-être pas la même importance,



on ne peut pas mettre en simple somme  $p(BA)$  et  $p(CA)$ . Il faudrait les pondérer avec  $b$  et  $c$ , ce qui équivaut à multiplier la probabilité de transition par le poids du lieu de départ.

Nous avons donc déduit ce système qui pouvait nous donner en solutions  $a$ ,  $b$  et  $c$  :

$$\begin{cases} a = p(BA)b + p(CA)c \\ b = p(AB)a + p(CB)c \\ c = p(AC)a + p(BC)b \end{cases} \quad \text{Pour avoir des solutions uniques, on fixe } a + b + c = 1$$

Après quelques calculs, notre système peut s'écrire sous la forme :

$$\begin{cases} c = \frac{p(AB)p(BA) - 1}{p(AB)(p(BA) - p(CA)) - p(CB)(p(BA) + 1) - p(CA) - 1} \\ b = \frac{1 - c(p(CA) + 1)}{p(BA) + 1} \\ a = \frac{c - p(BC)b}{p(AC)} \end{cases}$$

Nous l'avons seulement résolu dans le cas d'une chaîne de 3 lieux, car au-delà la situation devient beaucoup plus complexe. Ainsi, nous ne pourrions pas résoudre le système pour  $n$  lieux avec nos connaissances actuelles. Il faudrait utiliser une autre technique pour cela.

### b) Vecteur stable

Soit  $A$  une matrice carrée stochastique (rappelons le, une matrice est dite stochastique lorsque la somme des valeurs d'une ligne est égale à 1, elle est dite carrée lorsqu'elle a autant de lignes que de colonnes) de rang  $n$  (elle aura  $n$  lignes et  $n$  colonnes).

Soit  $h$  un vecteur dans un espace à  $n$  dimensions,  
 $h$  est dit stable de  $A$  si et seulement si :  $hA = h$ .

Considérons notre matrice de rang 3,  $h$  a donc 3 coordonnées.  $h = (a \quad b \quad c)$  On note

Mais comme  $h \cdot A = h$ , on peut trouver le système :

$$\begin{cases} a = p(BA)b + p(CA)c \\ b = p(AB)a + p(CB)c \\ c = p(AC)a + p(BC)b \end{cases}$$

Ce qui revient à notre calcul précédent, que nous avons trouvé intuitivement. Les **poids transitionnels** sont donc les coordonnées du vecteur stable de la matrice de transitions !

### 4/ Puissances d'une matrice carrée stochastique

Une matrice carrée  $A$  peut-être mise à la puissance  $n$ , en effet :  $A^n = (((A \times A) \times A) \times \dots) \times A$  et on peut multiplier une matrice carrée par une matrice carrée de mêmes dimensions. Ces matrices sont les seules matrices dont on peut calculer les puissances.

Le produit matriciel étant un calcul complexe, le calcul des puissances d'une matrice l'est encore plus. C'est pourquoi, après quelques tentatives à la main, nous avons décidé de créer un algorithme permettant de mettre rapidement une matrice carrée d'ordre 3 à la puissance  $n$ .



Les matrices de transition étant stochastiques, nous nous sommes intéressés aux puissances de ces matrices carrées stochastiques. Nous avons d'abord pu démontrer qu'une de ces matrices mise à la puissance  $n$  demeurerait stochastique. (9)

Par de nombreuses observations mais sans pouvoir le démontrer, nous avons remarqué que n'importe quelle matrice carrée stochastique (ici la chaîne exemple) suivait le théorème suivant :

$a, b$  et  $c$  sont les coordonnées du vecteur stable, les **poids transitionnels** !

Cette technique a l'avantage de pouvoir être employée pour des chaînes comprenant un grand nombre de lieux, car notre calculatrice peut assez rapidement calculer une matrice carrée d'ordre 15 à la puissance 100...

### 5/ Conclusion par un exemple :

Prenons un exemple pour rapprocher les trois méthodes :  
Matrice de transitions entre 3 lieux :

$$\begin{pmatrix} 0 & 0,5 & 0,5 \\ 0,2 & 0 & 0,8 \\ 1 & 0 & 0 \end{pmatrix}$$

$$a \approx 0,42 ; b \approx 0,21 \text{ et } c \approx 0,37$$

1/ Algorithme

$$\left\{ \begin{array}{l} a = \frac{5}{12} \approx 0,42 \\ b = \frac{5}{24} \approx 0,21 \\ c = \frac{3}{8} \approx 0,37 \end{array} \right.$$

2/ Coordonnées du vecteur stable :

3/ Matrice à la puissance  $\begin{pmatrix} 0,42 & 0,21 & 0,37 \\ 0,42 & 0,21 & 0,37 \\ 0,42 & 0,21 & 0,37 \end{pmatrix} 500 :$

Ces trois méthodes sont donc sensiblement les mêmes, mais la précision est plus grande avec la résolution du système car on obtient des valeurs exactes. Nous avons créé une animation Geogebra dans laquelle on entre les valeurs des probabilités de transition entre 3 lieux. (10)  
Ensuite, à partir du système d'équations nous obtenons le poids transitionnel de chaque lieu.

### 6/ Et dans la réalité ?

Nous avons pu réaliser une chaîne de Markov réelle basée sur les déplacements d'un membre de notre groupe qui contient plus de 10 lieux différents. Nous avons également calculé, sous forme de fréquence, le temps passé à chaque lieu, que nous avons appelé le **poids horaire**.

Nous avons évalué le **poids transitionnel**. Il s'est avéré classer les lieux dans un ordre proche de celui du **poids horaire** (les valeurs ne sont cependant pas exactement les mêmes). Le **poids transitionnel** est donc un indicateur fiable pour quantifier l'importance de chaque lieu pour une personne, et même éventuellement en déduire ce qu'ils représentent pour lui (domicile, travail...).

**NB :** À partir des observations des déplacements d'une personne, nous avons déduit des probabilités de transition et calculé le **poids transitionnel**. Nos techniques partent des **probabilités de transition** pour le calculer. Mais si l'on se base sur une simple observation des déplacements, nous pouvons regarder le nombre de fois où l'individu passe à un lieu, comme l'algorithme, et calculer trivialement le **poids transitionnel**...

#### **IV/ Le cercle de présence**

Au cours de nos recherches, nous avons aussi investigué le concept de **géocodage**. Cette technique consiste à attribuer à chacun des lieux de vie d'une personne des coordonnées dans un certain repère.

Par exemple, considérons 4 lieux de vie nommés : A ( $x_A, y_A$ ), B ( $x_B, y_B$ ), C ( $x_C, y_C$ ) et D ( $x_D, y_D$ ). Par observation des transitions effectuées, nous évaluons les **poids transitionnel** de chaque lieu : respectivement a, b, c et d. En connaissant les coordonnées des 4 points ainsi qu'un « poids » pour chacun d'entre eux, nous avons eu l'idée de calculer les coordonnées du barycentre G de ce système de points pondérés. Nous avons donc calculé la moyenne (pondérée par les poids de chaque lieu) des abscisses des points pour obtenir l'abscisse de G avec la formule suivante :

$$x_G = \frac{x_A a + x_B b + x_C c + x_D d}{a + b + c + d}$$

De même pour les ordonnées.

Nous avons donc obtenu en quelque sorte le centre des lieux de vie visités par une personne. Par la suite, nous calculons la moyenne (pondérée par le poids de A, de B...) des distances GA, GB... Elle devient le rayon d'un cercle de centre G. Ce cercle constitue un espace de présence moyen le plus probable. Il englobe les lieux les plus importants.

Ce « cercle de présence » peut avoir une utilité en statistiques par exemple, car il est plus représentatif des activités de l'individu que ne l'est le domicile seul. Mais il peut également être un danger car pour quelqu'un de mal intentionné, il peut en déduire que l'individu ne sort presque jamais de son cercle.

Nous avons réalisé une animation avec Géogébra pour obtenir ce cercle des lieux, en faisant varier ces lieux ainsi que leurs poids :

*Voir la capture d'écran en annexe photo 1*

**(11)**

#### **V/ La marge d'erreur algorithmique**

Pour une chaîne de 3 lieux, nous avons mis en place 3 techniques pour évaluer les poids transitionnels. La technique du vecteur stable nous fournissait (voir les formules) une valeur exacte. L'algorithme s'en rapprochait après un très grand nombre d'essais, représentant en soi la matrice de transitions élevée à une très grande puissance. Notre recherche s'est alors poursuivie afin d'évaluer la marge d'erreur du poids transitionnel algorithmique en fonction du nombre de transitions

employé par l'algorithme pour évaluer ce poids transitionnel. Le poids transitionnel résultant bien sûr des transitions, sa marge d'erreur correspond également à la moyenne de la marge de ces transitions (nous aurions pu l'évaluer plus formellement mais cela aurait été beaucoup plus complexe). Notre ancien algorithme (100000 transitions) semblait relativement fiable face aux valeurs exactes.

### 1/ Comment calculer la marge d'erreur ?

Le principe d'obtention de la marge d'erreur pour un lieu est plutôt simple et peut être réalisée avec la formule suivante : il s'agit du ratio de la différence entre la valeur théorique et la valeur obtenue expérimentalement sur la valeur théorique elle-même. On considère la valeur absolue de ce ratio car on n'a pas besoin de savoir lequel est le plus grand (marge positive ou négative), mais juste leur rapport (marge positive).

$$\text{Marge d'erreur} = \text{abs}\left(\frac{\text{Valeur expérimentale} - \text{Valeur théorique}}{\text{Valeur théorique}}\right)$$

(12)

Pour mesurer l'erreur globale, il faut faire la moyenne des marges d'erreurs des poids des trois lieux de notre chaîne.

### 2/ Algorithmique

Le principe général est simple : l'algorithme crée d'abord une chaîne de 3 lieux (transitions prises au hasard) et évalue les valeurs du poids transitionnel de chaque lieu avec les formules. Puis il fait suivre  $n$  transitions respectant cette loi de probabilité en évaluant à la fin les valeurs expérimentales des poids transitionnels.

#### Résumé :

Lire r

Lire n

Pour i allant de 1 à r :

- Création d'une chaîne aléatoire
- Calcul par les formules des poids
- Calcul des poids avec un algorithme au bout de  $n$  transitions
- Calcul de la marge d'erreur

Fin pour

L'algorithme calcule la marge d'erreur entre les poids transitionnels algorithmiques et les valeurs théoriques du système pour chaque lieu (A, B et C). On a donc 3 pourcentages dont il fait une moyenne qui est la valeur produite en sortie par l'algorithme.

Cet algorithme a été rendu ergonomique : on peut créer autant de chaînes que nécessaire (il les évaluera séparément) et faire subir à chacune autant de tests que l'on veut.

*Voir le code en langage AlgoBox en annexe 3*

#### Exemple :

On a demandé 3 chaînes et pour chacune on évalue les marges d'erreur moyennes sur 20, 40, 100 et 500 transitions.

#### Résultats :

##### Chaîne n°1

20 transitions : 23.129502% de différence  
 40 transitions : 10.848396% de différence  
 100 transitions : 1.8791675% de différence  
 500 transitions : 1.686865% de différence

##### Chaîne n°2

20 transitions : 18.978489% de différence  
 40 transitions : 11.086811% de différence  
 100 transitions : 9.5488923% de différence  
 500 transitions : 3.1311144% de différence

##### Chaîne n°3

20 transitions : 15.371826% de différence  
 40 transitions : 20.896946% de différence  
 100 transitions : 8.6918286% de différence  
 500 transitions : 1.000761% de différence

On note une diminution de la marge d'erreur quand le nombre de transitions devient de plus en plus grand, mais seulement en moyenne : ce n'est parfois pas le cas (chaîne 3, entre 20 et 40 transitions). Mais surtout, comme nos valeurs diffèrent d'une chaîne à une autre, la question se pose de savoir si on peut obtenir des moyennes sur un très grand nombre de chaînes.

### **3/ L'art de l'approximation**

Après ce premier algorithme, nous avons décidé d'en créer un plus précis. Cet algorithme ne fait qu'un seul type de test (on choisit le nombre de transitions) et le réalise sur un grand nombre (à entrer) de chaînes créées au hasard. Au lieu de les afficher séparément, il en fait une moyenne des marges d'erreur obtenues sur les différentes chaînes.

#### Résumé :

Lire r

Lire n

M prend la valeur 0

Pour i allant de 1 à r :

- Création d'une chaîne aléatoire
- Calcul par les formules des poids
- Calcul des poids avec un algorithme au bout de  $n$  transitions
- Calcul de la marge d'erreur
- M prend la valeur  $M + \text{marge d'erreur}$

Fin pour

Marge d'erreur moyenne :  $M/r$

*Voir le code en langage AlgoBox en annexe 4*

Plus l'on crée de chaînes (en conservant le même nombre de transitions), plus le pourcentage d'erreur semble tendre vers une valeur qui se révèle caractéristique de ce nombre de transitions. Nous avons alors eu l'idée de faire calculer ces fiabilités caractéristiques et de les associer au nombre de transitions effectuées (10, 20...) dans un tableau. Notre but était d'établir des correspondances entre le nombre de transitions et la marge d'erreur moyenne

Nombre $n$ de transitions	Valeur de la marge d'erreur
10	24.8
20	16.8
30	13.3
40	11.2
50	10.1
60	9.3
70	8.6
80	7.9
90	7.5
100	7.1
110	6.8
120	6.4
130	6.2
140	6
150	5.8
160	5.6
170	5.4
180	5.2

Et ainsi de suite... Il est logique (même trivial) que :

$$\lim_{n \rightarrow \infty} \text{marge d'erreur} = 0$$

(13)

Mais ces valeurs ont une faible précision car elles sont issues d'un échantillonnage. A l'aide du tableur de Géogébra nous avons tâtonné afin d'obtenir une fonction résumant la relation entre nombre de transitions et marge d'erreur moyenne. Nous en avons déduit un lien avec la fonction inverse.

(14)

Voir la capture d'écran en annexe photo 2

Ainsi, nous avons pu obtenir cette expression qui satisfait toutes nos expérimentations (avec  $f(n)$  la marge d'erreur en % au bout de  $n$  transitions),  $a$  étant une valeur à déterminer (vers 75).

$$f(n) = \frac{a}{\sqrt{n}} \quad (15)$$

Ceci pour une chaîne de 3 lieux. Nous ne pouvons rien démontrer mais il semble qu'il y ait bien une loi cachée là-dessous. La même question se pose maintenant pour une chaîne de 4 lieux.

#### **4/ Et dans la réalité ?**

Une personne ne se déplace pas comme le fait l'ordinateur (celui-ci est trop peu humain dans sa logique). Cependant ces travaux nous permettent d'avoir tout de même un aperçu de la mobilité possible d'un individu.

Dans le cadre d'une autre expérimentation que nous avons réalisée, un élève a équipé son téléphone d'une application permettant de le géolocaliser. Nous avons enregistré chaque semaine les déplacements qu'il réalisait, et pu en déduire les probabilités de transition entre les lieux qu'il fréquentait.

Ainsi, nous avons calculé les poids transitionnels dans la semaine [1], puis dans les semaines [1 et 2] regroupées, puis [1, 2 et 3] regroupées, etc. Nous avons déterminé un pourcentage moyen de variation des poids transitionnels entre les périodes [semaine 1] et [semaines 1 et 2], puis entre les périodes [semaines 1 et 2] et [semaines 1, 2 et 3], etc.

Il s'est avéré que le pourcentage de variation moyen des poids devenait inférieur à 1% au bout de 6 semaines (soit entre les périodes [semaines 1, 2, 3, 4 et 5] et [semaines 1, 2, 3, 4, 5 et 6]) : ils se stabilisaient autour de valeurs précises que nous allions pouvoir exploiter pour déterminer ce à quoi chaque lieu correspondait.

Les poids transitionnels sont fiables assez rapidement. On peut donc facilement les trouver, sans avoir d'observation trop longue : la menace pour notre vie privée en est d'autant plus grande.

#### **5/ Répartition des valeurs de marge d'erreur**

##### **A/ Répartition simple**

Revenons au premier algorithme. Il nous donnait des valeurs de marge d'erreur différentes pour un même nombre de transitions. Si pour le second algorithme la moyenne semblait caractéristique, elle ne nous donnait aucunement la répartition des marges obtenues. Ainsi la question se pose de déterminer si la probabilité d'obtenir une marge d'erreur de 1% est la même que d'obtenir une marge de 50% pour un nombre de transitions fixées.

Nous avons donc créé un algorithme permettant de réaliser un certain nombre de chaînes, et pour chacune d'entre elles, de calculer une marge d'erreur à un nombre  $n$  de transitions, qui est fixé. Comme les valeurs de marge d'erreur sont un phénomène continu, nous avons décidé de les regrouper en classes de 1. Ainsi, l'algorithme compte le nombre de fois où la marge obtenue était entre 0% et 1%, entre 1% et 2%, etc.

L'algorithme a simplement calculé des fréquences d'apparition de chaque classe pour tracer un graphique.

*Voir la capture d'écran en annexe photo 3*

**(16)**

Ainsi, en abscisse de chaque point nous avons la valeur de départ de la classe et en ordonnée la fréquence d'apparition des valeurs de la classe par rapport à toutes les valeurs. Le point n'apparaît que si son ordonnée est non-nulle.

Il se forme une courbe en cloche (avec un grand nombre de chaînes) qui semble avoir pour maximum la valeur moyenne que nous avons calculée en 3/, caractéristique de ce nombre  $n$  de transitions (ici 10, la moyenne vaut environ 25%).

### B/ Répartition de plusieurs transitions

Cependant, nous souhaitons aller encore plus loin. Ainsi, nous avons modifié notre algorithme pour comparer plusieurs entiers  $n$  (jusqu'à 3). Les 3 valeurs sont passées en entrée de manière successive. Dans le cas où nous voulions moins de valeurs, nous pouvions indiquer 0 et il ne ferait pas le calcul.

Sur un même graphe, nous avons fait figuré les 3 listes de points en utilisant des couleurs différentes.

*Voir le code en langage AlgoBox en annexe 5*

### Exemple :

Nombre de transitions au 1er test : 2  
 Nombre de transitions au 2nd test : 20  
 Nombre de transitions au 3ème test : 70  
 Nombre de chaînes testées : 4000

### Observations :

*Voir la capture d'écran en annexe photo 4*

Les fréquences maximales de chaque courbe correspondent environ aux moyennes : les sommets sont alignés sur la courbe figurant dans la section 3.

Plus l'on augmente le nombre de chaînes testées (elle ne varie pas sur le graphique ici), plus la courbe en cloche se précise donc plus la marge d'erreur devient précise. Ceci est logique et nous pourrions même faire des marges d'erreurs de marges d'erreur... Plus le nombre de transitions augmente (et c'est ce qui varie ici), plus le sommet de la courbe se décale vers 0, autrement dit plus la marge d'erreur moyenne tend vers 0. Par exemple, pour  $n=70$ , le maximum est à environ 6% et pour  $n=2$ , il est d'environ 65%. Cependant, plus ce nombre de transitions augmente, plus les valeurs sont resserrées autour de la moyenne (voir le graphique) : la marge d'erreur devient alors de plus en plus précise !

### 6/ Chaîne par chaîne



Tout notre travail se focalisait sur le point suivant : on testait un très grand nombre de chaînes, une fois chacune (voir résumés précédents des algorithmes). Et si on testait une seule et unique chaîne, plusieurs fois sur un même nombre de transitions.

### A/ Répartition des valeurs

#### Résumé :

```

Lire r
Création d'une chaîne aléatoire
Calcul par les formules des poids
Pour i allant de 1 à r :
- Calcul des poids avec un algorithme au bout de  $n$  transitions
- Calcul de la marge d'erreur
Fin pour

```

Avec le test de chaînes différentes, nous avons une répartition en cloche. Mais qu'en est-il si l'on ne considère qu'une seule chaîne : y a-t-il des valeurs remarquables ou encore une cloche, autrement dit la cloche est-elle due au fait que les chaînes soient différentes les unes des autres ?

**(17)**

Dans ce but, nous retravaillons l'algorithme de telle sorte que, pour une même chaîne testée, il nous renvoie graphiquement les fréquences d'apparition des marges d'erreurs.

#### Résumé :

```

Lire r
Lire n1
Lire n2
Lire n3
Création d'une chaîne aléatoire
Calcul par les formules des poids
Pour i allant de 1 à r :
- Calcul des poids avec un algorithme au bout de  $n1$  transitions
- Calcul de la marge d'erreur
Fin pour
Affichage des fréquences pour  $n1$  (rouge)
Pour i allant de 1 à r :
- Calcul des poids avec un algorithme au bout de  $n2$  transitions
- Calcul de la marge d'erreur
Fin pour
Affichage des fréquences pour  $n2$  (vert)
Pour i allant de 1 à r :
- Calcul des poids avec un algorithme au bout de  $n3$  transitions
- Calcul de la marge d'erreur
Fin pour
Affichage des fréquences pour  $n3$  (bleu)

```

### Exemple 1:

L'on commence par fixer :  $n1 = n2 = n3 = 3$ . Ainsi, l'on observe les fréquences d'apparition (nous aurons les mêmes points rouges, bleus et verts) pour les trois essais. Les valeurs se

répartissent sur 4 marges d'erreurs précises : ici, environ 12%, 62%, 68% et 83%. Pour cette chaîne, à 10 transitions, l'on n'obtient pas d'autres valeurs, ainsi elles semblent caractéristiques de cette chaîne à 10 transitions.

#### Exemple 2:

Avec une autre chaîne tirée au hasard, nous obtenons de manière logique d'autres marges précises, ici au nombre de 5 : 52%, 55%, 65%, 88% et 97%. Ces valeurs sont caractéristiques de la chaîne.

#### Exemple 3:

Faisons maintenant varier  $n_1$ ,  $n_2$  et  $n_3$ . Ici, sur une autre chaîne,  $n_1 = 3$ ,  $n_2 = 8$  et  $n_3 = 12$  transitions par répétition.

#### OBSERVATIONS :

- les valeurs précises sont caractéristiques d'une chaîne à un nombre de transitions données : elles varient quand on change les entiers  $n$ .
- plus  $n$  est grand, plus la marge d'erreur semble diminuer, même sur une même chaîne.
- plus  $n$  est grand, plus le nombre de valeurs caractéristiques semble augmenter et ressembler à une courbe en cloche.

#### Exemple 4 :

Afin de vérifier cette dernière hypothèse, prenons un  $n_1 = n_2 = 3$  et  $n_3 = 50$ .

*Voir la capture d'écran en annexe photo 5*

On note en effet un nombre de points beaucoup plus importants sur  $n_3$ . Ces points semblent s'aligner en une courbe en cloche.

#### B/ D'une manière moyenne

Notre algorithme nous affichera maintenant, en plus du graphe, la moyenne d'une chaîne pour un certain nombre de transitions.

Avec  $n_1 = n_2 = n_3$ , nous obtenons une valeur moyenne qui est la même, caractéristique de la chaîne à un certain  $n_1$  nombre de transitions.

Avec  $n_1 < n_2 < n_3$ , la moyenne diminue, ce qui est logique.

Mais dès lors que l'on prend une autre chaîne, les valeurs de moyennes sont modifiées, même en gardant le même nombre de transitions.

### **7/ Conclusion**

Nos études cachent encore des lois de probabilités complexes que nous n'avons pas abordées, mais elles semblent d'un grand intérêt..

Cependant, pour expliquer la répartition en cloche de la section 5A) à 10 transitions sur de nombreuses chaînes, il faudrait :

- déterminer le lien entre les valeurs précises et leurs fréquences d'apparition (que l'on obtient pour une chaîne précise) en fonction des probabilités de transitions de cette chaîne (encore à creuser algorithmiquement ?).

- prendre en compte les choix aléatoires que réalise l'ordinateur sur les probabilités de transition.

## VI/ Comment crypter une chaîne de Markov ?

Les chaînes de mobilité de Markov possèdent une particularité : la somme des probabilités au départ d'un lieu vaut 1. Nous nous sommes alors posé la question : pouvait-on crypter de manière simple, une chaîne de mobilité de Markov afin de réaliser un transfert sécurisé de ces données personnelles.

Dans ce but, nous avons cherché à modifier la chaîne en rajoutant une **unique** information de type nombre : la **clef de cryptage**  $k$ , commune à l'expéditeur et au destinataire. Par divers procédés nous avons tenté de trouver des manières originales de crypter une chaîne, en employant notamment la propriété de somme des probabilités.

Pour l'exemple, nous considérons une chaîne de 3 lieux avec une matrice de transitions de ce type :

$$\begin{pmatrix} 0 & p(AB) & p(AC) \\ p(BA) & 0 & p(BC) \\ p(CA) & p(CB) & 0 \end{pmatrix}$$

### 1/ Produit de la matrice de transitions par un réel

Le plus simple semblait, bien sûr, de multiplier l'ensemble de la matrice de transitions par un réel  $k$  pour obtenir la matrice cryptée. D'après les règles de calcul :

$$\begin{pmatrix} 0 & p(AB) & p(AC) \\ p(BA) & 0 & p(BC) \\ p(CA) & p(CB) & 0 \end{pmatrix} \times k = \begin{pmatrix} 0 & k \times p(AB) & k \times p(AC) \\ k \times p(BA) & 0 & k \times p(BC) \\ k \times p(CA) & k \times p(CB) & 0 \end{pmatrix}$$

Mais cela pose un gros problème :  $k$  est très facile à trouver pour quelqu'un interceptant le message. En effet, la somme de la première ligne vaut :

$$k \times p(AB) + k \times p(AC) = k$$

En général, on pourra donc facilement casser le code dès que les lignes sont multipliées par des valeurs, du fait de l'aspect stochastique de la matrice.

## 2/ Ajout d'une valeur à chaque probabilité

Si l'on ajoute à chaque probabilité une valeur  $k$ , la somme de la première ligne vaut :

$$p(AB) + p(AC) + 2k = 1 + 2k$$

L'aspect stochastique de la matrice pose encore le même problème : on peut retrouver très facilement la valeur de  $k$ .

## 3/ Multiplication des transitions à l'arrivée

Multiplier les transitions au départ d'un lieu n'était pas très efficace, du fait de la stochasticité de la matrice. Mais celle-ci n'est pas bistochastique : la somme des probabilités arrivant sur un lieu de vaut pas 1 : nous n'aurons pas de problème avec ce qui va suivre. **(18)**

Nous allons donc multiplier les valeurs d'une même colonne par un nombre, les valeurs d'une autre par un autre... Mais il ne nous faut qu'une seule clef de cryptage. Nous multiplierons donc la première ligne par  $k$ , la seconde par  $k+1$ , la troisième par  $k+2$  et ainsi de suite.

Ainsi, à partir de la matrice de départ nous avons :

$$\begin{pmatrix} 0 & (k+1) \times p(AB) & (k+2) \times p(AC) \\ k \times p(BA) & 0 & (k+2) \times p(BC) \\ k \times p(CA) & (k+1) \times p(CB) & 0 \end{pmatrix}$$

C'est assez dérangeant pour notre espion. Quand bien même il saurait ce que la seconde ligne aurait été multipliée par  $1 + \text{le facteur multiplicateur de la première}$ , il ne serait tout de même pas bien avancé car il lui manquerait toujours  $k$ . Le destinataire connaissant  $k$  n'aura qu'à raisonner à l'envers pour retrouver la matrice de départ.

Cependant, il pourrait toujours trouver le rapport entre  $p(BA)$  et  $p(CA)$  : car ces deux grandeurs sont multipliées par la même valeur  $k$ . Puis il pourrait progresser en sachant  $p(AB) + p(AC) = 1$ , en employant un système d'équations (cela devient par contre plus difficile quand le nombre de lieux augmente).

Pour cela, il nous fallait quelque chose de plus radical, de plus dérangeant encore pour l'espion.

## 4/ Une chaîne de Markov donne une autre chaîne de Markov

Dans tous les cas, la matrice obtenue n'était plus stochastique : elle ne pouvait plus correspondre à une chaîne de Markov. Et si, dans notre cryptage, la chaîne cryptée n'était autre... qu'une chaîne de Markov elle-même, différente de l'originale. Et l'espion ne saurait plus si les chaînes sont véritables ou faussées (remarque : dans ce cas-là on fait de la stéganographie qui consiste à cacher la présence d'un message plutôt que de la cryptographie...).

Pour cela, nous avons établi le protocole suivant :

- on choisit une valeur de  $k$  (un réel strictement supérieur à 0)
- la première probabilité de chaque ligne est multipliée par  $k$  puis divisée par  $k+1$  (avec  $k > 0$ , elle reste bien entre 0 et 1)

- la seconde probabilité (sur une chaîne de trois lieux) prend la valeur  $1 - la$  première.

Ainsi, l'on a :

$$\begin{pmatrix} 0 & \frac{p(AB) \times k}{k+1} & 1 - \frac{p(AB) \times k}{k+1} \\ \frac{p(BA) \times k}{k+1} & 0 & 1 - \frac{p(BA) \times k}{k+1} \\ \frac{p(CA) \times k}{k+1} & 1 - \frac{p(CA) \times k}{k+1} & 0 \end{pmatrix}$$

Cette chaîne de Markov est une chaîne de Markov valide. Ce protocole étant complexe, un algorithme peut permettre de crypter la chaîne et un autre de la décrypter aisément (en inversant le calcul), ce que nous avons réalisé.

*Voir le code en langage AlgoBox en annexe 6*

Ce type de cryptage peut-être couplé avec un autre, par exemple celui de la multiplication des données de l'arrivée à un lieu.

L'espion peut récolter des données personnelles, mais celles-ci seront toutes erronées (transitions, donc poids transitionnels et rôle de chaque lieu, cercle des déplacements). La vie privée est donc respectée.

## VII Conclusion

Au cours de nos recherches, nous avons découvert qu'il est très facile de retrouver l'importance relative des différents lieux visités par un individu et d'en déduire ce qu'ils constituent pour l'individu suivi. Nous en avons donc conclu que l'on doit absolument protéger son anonymat afin d'éviter des atteintes à la vie privée tel que le marketing ciblé, le vol mais également l'usurpation d'identité qui a des conséquences bien plus désastreuses. Notre anonymat peut être atteint par le biais d'objets du quotidien tels qu'une carte navigo, un GPS, un téléphone portable... tous ces objets à l'apparence anodine permettent de tout savoir sur l'adresse, les déplacements, les horaires d'un individu, ils ont donc le pouvoir de détruire l'anonymat.

Plusieurs questions sont encore ouvertes :

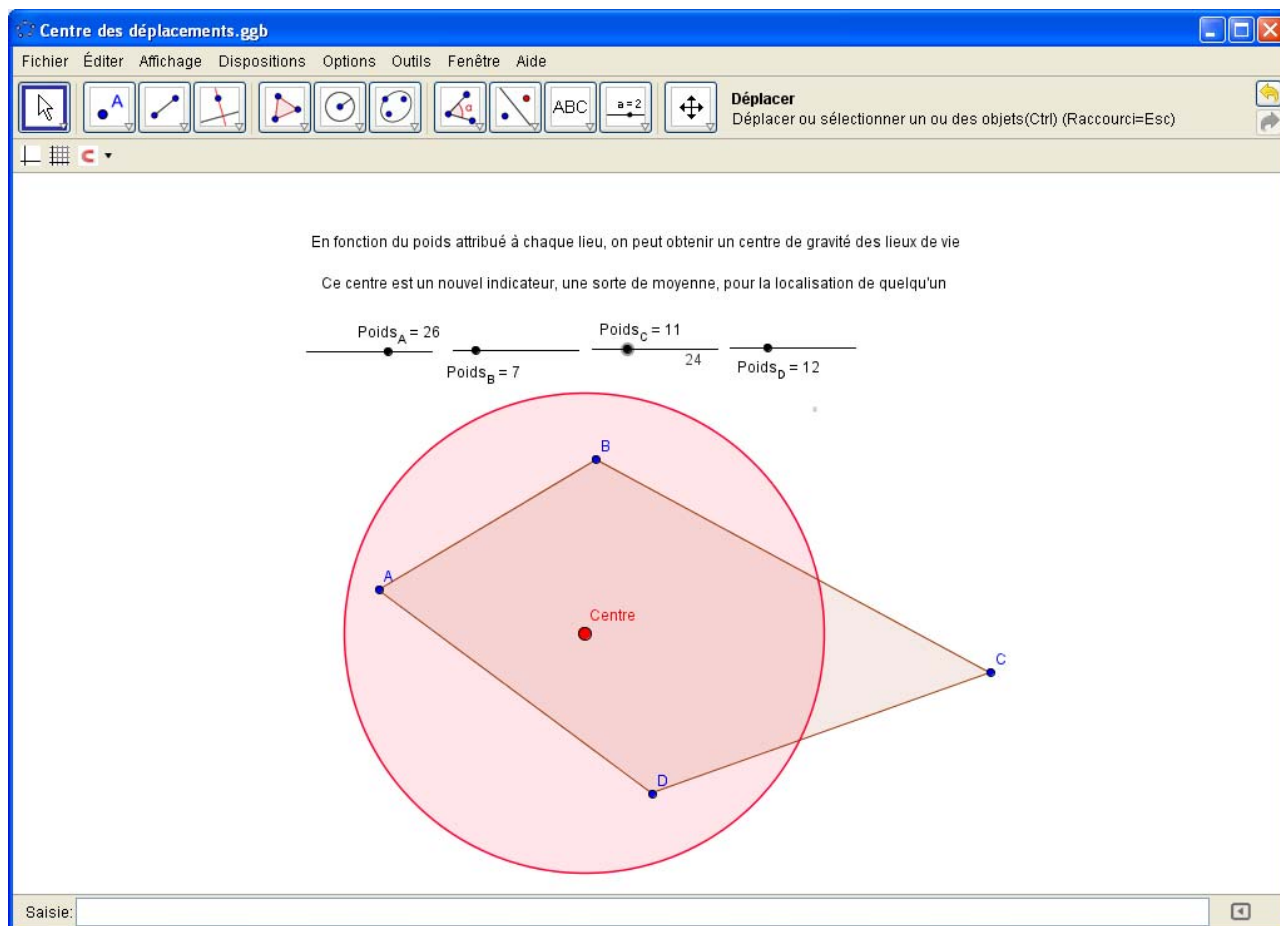
- la chaîne de Markov est dans un espace markovien, c'est-à-dire que la probabilité d'une transition ne dépend pas de la transition précédente. Il s'agit d'une limite de précision par rapport à la réalité. Peut-on créer une chaîne non-markovienne ? (remarque : il est possible d'étendre le modèle de la chaîne de Markov en gardant en mémoire les 2 ou + derniers lieux visités).
- une chaîne de Markov complète possède des probabilités AA, BB... (l'individu reste à A, à B...) Comment les évaluer ?
- peut-on démontrer la valeur de la limite des puissances d'une matrice stochastique, vue en III.4) (page 9) ?

## Notes d'édition

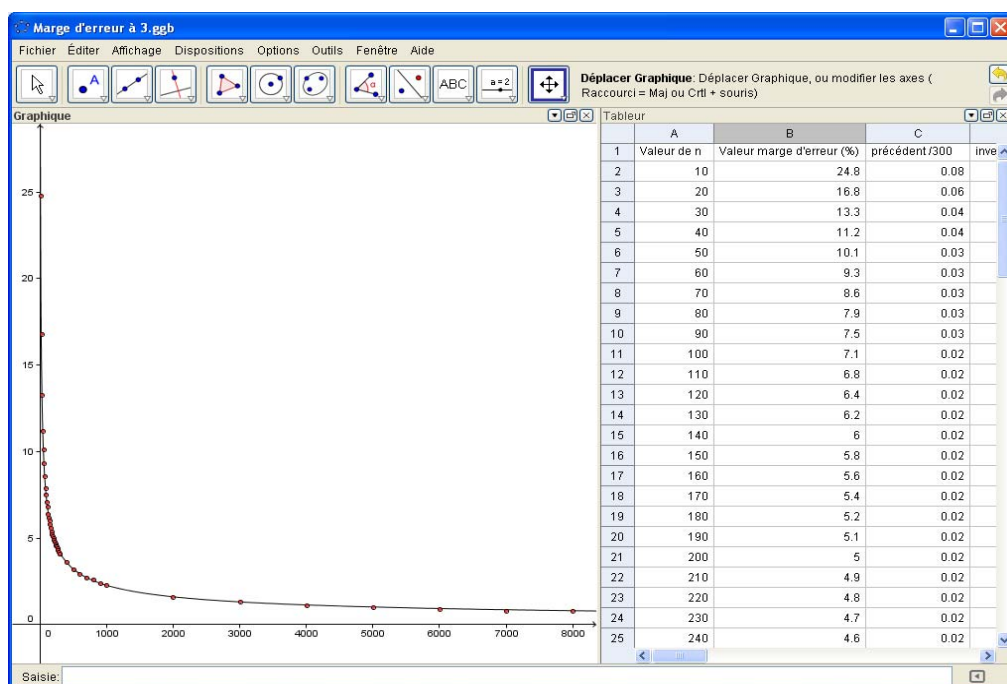
- (1) Le sujet original posé par le chercheur a été donné en langue anglaise, et considère une thématique vaste. Le lecteur peu familier avec cette langue peut sauter ce paragraphe, le thème retenu par les élèves étant précisé et explicité ci-après.
- (2) Les élèves emploient ici et pour la suite de l'article un vocabulaire expert, en parlant de « transition » cela correspond à un « déplacement » entre deux lieux, dans le contexte de la géolocalisation. De plus, la connaissance des fréquences de déplacement entre divers lieux, observées expérimentalement, permet de définir des probabilités de déplacement entre ces lieux, appelées pour la suite « probabilités de transition ».
- (3) Le tableau ci-dessus indique les fréquences des déplacements effectués par l'élève sur une semaine. 16 lieux différents ont été identifiés, ils sont numérotés de A à Q.
- (4) Pour chaque niveau de jeu, l'ordinateur crée des probabilités de transitions (déplacements) de façon aléatoire entre les lieux fréquentés par Bob (de 2 à 5 suivant le niveau). Puis, il simule un ensemble fini de déplacements de Bob, cohérent avec ces probabilités, et qu'il indique au joueur. Ce dernier doit alors prévoir les déplacements à venir de Bob à partir d'un lieu de départ, afin de le capturer.
- (5) c'est-à-dire, que certains déplacements sont plus fréquents que d'autres.
- (6) Il est difficile de comprendre ici ce que les élèves veulent exprimer. En effet, même si l'on s'intéresse au temps passé dans chaque lieu, l'aléatoire restera présent.
- (7) Une matrice est une forme de représentation d'un tableau à double entrée.
- (8) Une matrice stochastique est en effet une matrice à  $n$  lignes et  $n$  colonnes, dont chaque élément est un nombre compris entre 0 et 1 et dont la somme des éléments de chaque ligne vaut 1.
- (9) La démonstration n'a pas été fournie par les élèves. Le résultat annoncé est toutefois exact.
- (10) Les élèves n'ont pas fourni d'information supplémentaire concernant cette animation Geogebra.
- (11) Attention : l'exemple qui est donné n'est pas compatible avec les définitions précédentes. En effet, les poids devraient être des nombres compris entre 0 et 1 de somme totale égale à 1, ce qui n'est pas le cas de l'exemple de la photo.
- (12)  $\text{abs}$  désigne la fonction valeur absolue d'un nombre réel.
- (13) Le résultat est conjecturé par les élèves, mais n'est pas démontré.
- (14) Et avec la fonction racine carrée.
- (15) La formule est obtenue de façon expérimentale, et n'est pas exacte.
- (16) On regrettera que sur ce graphique (ainsi que sur les deux suivants) les échelles en abscisse et en ordonnée ne soient pas indiquées, de même que les légendes pour les différentes couleurs de points.
- (17) Les élèves parlent de « courbe en forme de cloche », la célèbre courbe de Gauss très présente dans le domaine des probabilités. Pour en savoir plus, on pourra consulter : <http://images.math.cnrs.fr/La-courbe-verte-en-cloche.html>.
- (18) Une matrice est dite bistochastique si la somme des éléments de chaque ligne ET de chaque colonne vaut 1.

# Annexes photos :

## 1/ Géogebra : le cercle de présence

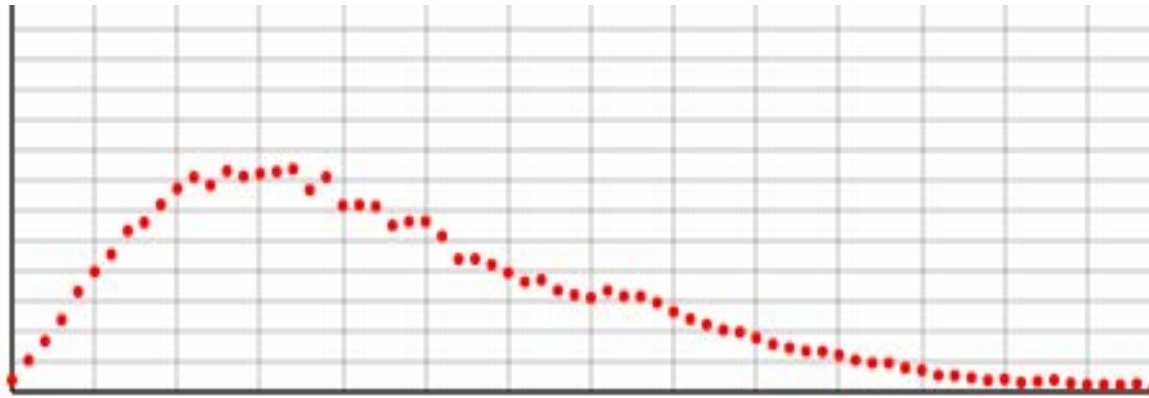


## 2/ Géogebra : la marge d'erreur moyenne est fonction du nombre de transitions

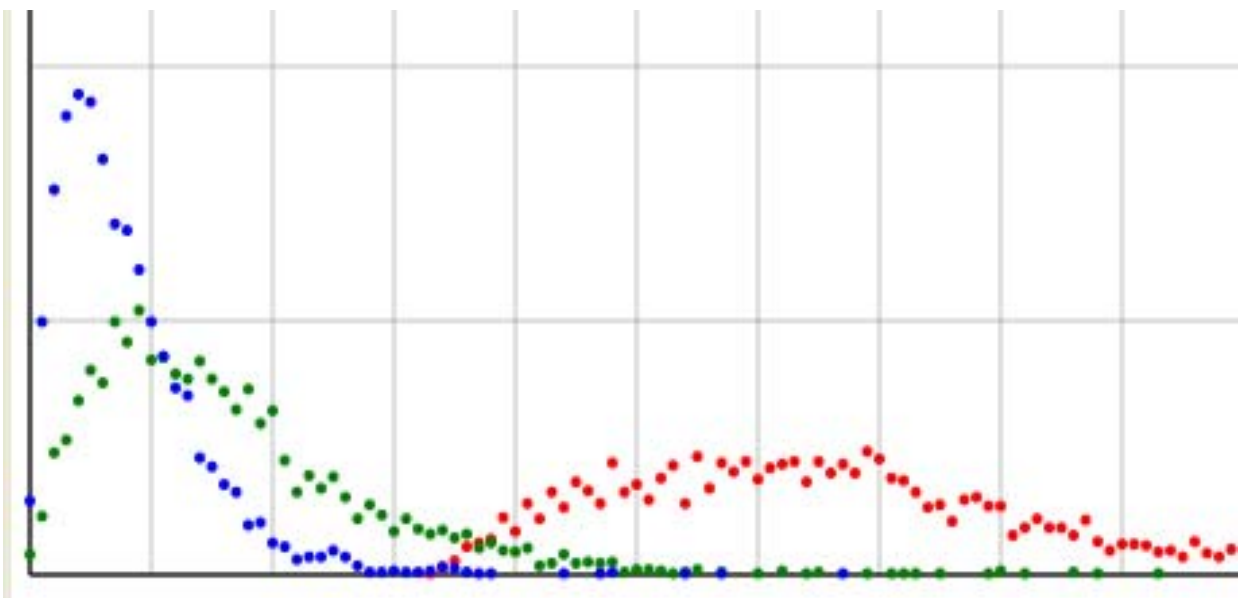




### 3 / Algobox : répartition des erreurs effectuées à 10 transitions



### 4/ Algobox : répartition des erreurs effectuées à 2, 20 et 70 transitions



### 5/ Algobox : répartition des erreurs effectuées à 3, 3 et 50 transitions (1 chaîne)



## **Annexes :**

### **1/ Un chemin de transitions**

```
1  VARIABLES
2  AB EST_DU_TYPE NOMBRE
3  AC EST_DU_TYPE NOMBRE
4  BA EST_DU_TYPE NOMBRE
5  BC EST_DU_TYPE NOMBRE
6  CA EST_DU_TYPE NOMBRE
7  CB EST_DU_TYPE NOMBRE
8  r EST_DU_TYPE NOMBRE
9  n EST_DU_TYPE NOMBRE
10 A EST_DU_TYPE NOMBRE
11 B EST_DU_TYPE NOMBRE
12 C EST_DU_TYPE NOMBRE
13 fA EST_DU_TYPE NOMBRE
14 fB EST_DU_TYPE NOMBRE
15 fC EST_DU_TYPE NOMBRE
16 i EST_DU_TYPE NOMBRE
17 DEBUT_ALGORITHME
18 AFFICHER "Enter les probabilités de transition avec 2 décimales"
19 LIRE AB
20 LIRE AC
21 LIRE BA
22 LIRE BC
23 LIRE CA
24 LIRE CB
25 SI (AB+AC!=1 OU BA+BC!=1 OU CA+CB!=1) ALORS
26   DEBUT_SI
27   AFFICHER "Les transitions sont incorrectes"
28   FIN_SI
29   SINON
30     DEBUT_SINON
31     r PREND_LA_VALEUR floor(2*random())
32     POUR i ALLANT_DE 1 A 50000
33       DEBUT_POUR
34       SI (r==0) ALORS
35         DEBUT_SI
36         n PREND_LA_VALEUR random()
37         A PREND_LA_VALEUR A+1
38         SI (n<AB) ALORS
39           DEBUT_SI
40           r PREND_LA_VALEUR 1
41           FIN_SI
42           SINON
43             DEBUT_SINON
44             r PREND_LA_VALEUR 2
45             FIN_SINON
46           FIN_SI
47         SI (r==1) ALORS
48           DEBUT_SI
49           n PREND_LA_VALEUR random()
50           B PREND_LA_VALEUR B+1
51           SI (n<BA) ALORS
52             DEBUT_SI
53             r PREND_LA_VALEUR 0
54             FIN_SI
55             SINON
56               DEBUT_SINON
57               r PREND_LA_VALEUR 2
58               FIN_SINON
59             FIN_SI
60           SI (r==2) ALORS
61             DEBUT_SI
62             n PREND_LA_VALEUR random()
63             C PREND_LA_VALEUR C+1
64             SI (n<CA) ALORS
65               DEBUT_SI
66               r PREND_LA_VALEUR 0
67               FIN_SI
68               SINON
69                 DEBUT_SINON
70                 r PREND_LA_VALEUR 1
71                 FIN_SINON
72               FIN_SI
```

```

73     FIN_POUR
74     fA PREND_LA_VALEUR A/(A+B+C)
75     fB PREND_LA_VALEUR B/(A+B+C)
76     fC PREND_LA_VALEUR C/(A+B+C)
77     AFFICHER "f(A) = "
78     AFFICHER fA
79     AFFICHER "f(B) = "
80     AFFICHER fB
81     AFFICHER "f(C) = "
82     AFFICHER fC
83     FIN_SINON
84 FIN_ALGORITHME

```

## 2/ Le jeu

```

1  VARIABLES
2  i EST_DU_TYPE NOMBRE
3  AB EST_DU_TYPE NOMBRE
4  AC EST_DU_TYPE NOMBRE
5  BA EST_DU_TYPE NOMBRE
6  BC EST_DU_TYPE NOMBRE
7  CA EST_DU_TYPE NOMBRE
8  CB EST_DU_TYPE NOMBRE
9  D EST_DU_TYPE NOMBRE
10 Intervention EST_DU_TYPE CHAINE
11 i2 EST_DU_TYPE NOMBRE
12 H EST_DU_TYPE NOMBRE
13 Bob1 EST_DU_TYPE NOMBRE
14 Bob EST_DU_TYPE NOMBRE
15 T EST_DU_TYPE NOMBRE
16 pAB EST_DU_TYPE NOMBRE
17 pAC EST_DU_TYPE NOMBRE
18 pAD EST_DU_TYPE NOMBRE
19 pBA EST_DU_TYPE NOMBRE
20 pBC EST_DU_TYPE NOMBRE
21 pBD EST_DU_TYPE NOMBRE
22 pCA EST_DU_TYPE NOMBRE
23 pCB EST_DU_TYPE NOMBRE
24 pCD EST_DU_TYPE NOMBRE
25 pDA EST_DU_TYPE NOMBRE
26 pDB EST_DU_TYPE NOMBRE
27 pDC EST_DU_TYPE NOMBRE
28 AD EST_DU_TYPE NOMBRE
29 BD EST_DU_TYPE NOMBRE
30 CD EST_DU_TYPE NOMBRE
31 DA EST_DU_TYPE NOMBRE
32 DB EST_DU_TYPE NOMBRE
33 DC EST_DU_TYPE NOMBRE
34 U EST_DU_TYPE LISTE
35 r EST_DU_TYPE NOMBRE
36 A EST_DU_TYPE NOMBRE
37 Continuer EST_DU_TYPE CHAINE
38 pAE EST_DU_TYPE NOMBRE
39 AE EST_DU_TYPE NOMBRE
40 pBE EST_DU_TYPE NOMBRE
41 BE EST_DU_TYPE NOMBRE
42 pCE EST_DU_TYPE NOMBRE
43 CE EST_DU_TYPE NOMBRE
44 pDE EST_DU_TYPE NOMBRE
45 DE EST_DU_TYPE NOMBRE
46 pEA EST_DU_TYPE NOMBRE
47 EA EST_DU_TYPE NOMBRE
48 pEB EST_DU_TYPE NOMBRE
49 EB EST_DU_TYPE NOMBRE
50 pEC EST_DU_TYPE NOMBRE
51 EC EST_DU_TYPE NOMBRE
52 pED EST_DU_TYPE NOMBRE
53 ED EST_DU_TYPE NOMBRE
54 DEBUT_ALGORITHME
55 H PREND_LA_VALEUR 3
56 D PREND_LA_VALEUR 2
57 POUR i ALLANT_DE 1 A 1
58     DEBUT_POUR
59     AB PREND_LA_VALEUR 0
60     AC PREND_LA_VALEUR 0
61     AD PREND_LA_VALEUR 0

```

```

62 AE PREND_LA_VALEUR 0
63 BA PREND_LA_VALEUR 0
64 BC PREND_LA_VALEUR 0
65 BD PREND_LA_VALEUR 0
66 BE PREND_LA_VALEUR 0
67 CA PREND_LA_VALEUR 0
68 CB PREND_LA_VALEUR 0
69 CD PREND_LA_VALEUR 0
70 CE PREND_LA_VALEUR 0
71 DA PREND_LA_VALEUR 0
72 DB PREND_LA_VALEUR 0
73 DC PREND_LA_VALEUR 0
74 DE PREND_LA_VALEUR 0
75 EA PREND_LA_VALEUR 0
76 EB PREND_LA_VALEUR 0
77 EC PREND_LA_VALEUR 0
78 ED PREND_LA_VALEUR 0
79 FIN_POUR
80 TANT_QUE (H>0) FAIRE
81   DEBUT_TANT_QUE
82   SI (D==2) ALORS
83     DEBUT_SI
84     T PREND_LA_VALEUR pow(D,3)
85     AFFICHER "NIVEAU I : "
86     AFFICHER "Cette semaine, Bob a fréquenté les lieux A et B dont voici la liste des transitions"
87     PAUSE
88     POUR i ALLANT_DE 1 A 1
89       DEBUT_POUR
90       pAB PREND_LA_VALEUR 1
91       pBA PREND_LA_VALEUR 1
92       AB PREND_LA_VALEUR floor((T/D)*pAB)+1
93       BA PREND_LA_VALEUR floor((T/D)*pBA)+1
94       FIN_POUR
95     POUR i ALLANT_DE 1 A 1
96       DEBUT_POUR
97       AFFICHER "AB : "
98       AFFICHER AB
99       AFFICHER "BA : "
100      AFFICHER BA
101      FIN_POUR
102    POUR i ALLANT_DE 1 A 1
103      DEBUT_POUR
104      Bob1 PREND_LA_VALEUR floor(D*random())+1
105      SI (Bob1==1) ALORS
106        DEBUT_SI
107        AFFICHER "Bob est maintenant parti de A."
108        Bob PREND_LA_VALEUR 2
109        FIN_SI
110      SI (Bob1==2) ALORS
111        DEBUT_SI
112        AFFICHER "Bob est maintenant parti de B."
113        Bob PREND_LA_VALEUR 1
114        FIN_SI
115      AFFICHER "Inspecteur, vous devez décider où envoyer vos effectifs : "
116      PAUSE
117      LIRE Intervention
118      SI (Intervention=="A" OU Intervention=="a") ALORS
119        DEBUT_SI
120        i2 PREND_LA_VALEUR 1
121        FIN_SI
122      SI (Intervention=="B" OU Intervention=="b") ALORS
123        DEBUT_SI
124        i2 PREND_LA_VALEUR 2
125        FIN_SI
126      SI (i2==Bob) ALORS
127        DEBUT_SI
128        D PREND_LA_VALEUR D+1
129        AFFICHER "Bravo ! Vos agents ont capturé Bob. "
130        AFFICHER "Mais réussirez-vous à le faire avec "
131        AFFICHER D
132        AFFICHER " lieux ?"
133        H PREND_LA_VALEUR 3
134        FIN_SI
135      SINON
136        DEBUT_SINON
137        H PREND_LA_VALEUR H-1

```

```

138     AFFICHER "Non, Bob n'était pas là et il risque de mettre son plan à exécution."
139     AFFICHER "Il ne vous reste plus que "
140     AFFICHER H
141     SI (H>1 OU H==0) ALORS
142         DEBUT_SI
143         AFFICHER " semaines."
144         FIN_SI
145     SINON
146         DEBUT_SINON
147         AFFICHER " semaine."
148         FIN_SINON
149     SI (H==0) ALORS
150         DEBUT_SI
151         AFFICHER "Vous avez perdu le jeu (et votre emploi...)"
152         FIN_SI
153     FIN_SINON
154     PAUSE
155     FIN_POUR
156     POUR i ALLANT_DE 1 A 1
157         DEBUT_POUR
158         AB PREND_LA_VALEUR 0
159         AC PREND_LA_VALEUR 0
160         AD PREND_LA_VALEUR 0
161         AE PREND_LA_VALEUR 0
162         BA PREND_LA_VALEUR 0
163         BC PREND_LA_VALEUR 0
164         BD PREND_LA_VALEUR 0
165         BE PREND_LA_VALEUR 0
166         CA PREND_LA_VALEUR 0
167         CB PREND_LA_VALEUR 0
168         CD PREND_LA_VALEUR 0
169         CE PREND_LA_VALEUR 0
170         DA PREND_LA_VALEUR 0
171         DB PREND_LA_VALEUR 0
172         DC PREND_LA_VALEUR 0
173         DE PREND_LA_VALEUR 0
174         EA PREND_LA_VALEUR 0
175         EB PREND_LA_VALEUR 0
176         EC PREND_LA_VALEUR 0
177         ED PREND_LA_VALEUR 0
178         FIN_POUR
179         FIN_SI
180     SI (D==3) ALORS
181         DEBUT_SI
182         n PREND_LA_VALEUR D+1
183         T PREND_LA_VALEUR pow(D,3)
184         AFFICHER "NIVEAU II : "
185         AFFICHER "Cette semaine, Bob a fréquenté les lieux A,B et C dont voici la liste des transitions"
186         PAUSE
187         POUR i ALLANT_DE 1 A 1
188             DEBUT_POUR
189             pAB PREND_LA_VALEUR random()
190             pAC PREND_LA_VALEUR 1-pAB
191             pBA PREND_LA_VALEUR random()
192             pBC PREND_LA_VALEUR 1-pBA
193             pCA PREND_LA_VALEUR random()
194             pCB PREND_LA_VALEUR 1-pCA
195             r PREND_LA_VALEUR floor(3*random())+1
196             FIN_POUR
197         POUR i ALLANT_DE 1 A T
198             DEBUT_POUR
199             SI (r==1) ALORS
200                 DEBUT_SI
201                 r PREND_LA_VALEUR random()
202                 SI (r<pAB) ALORS
203                     DEBUT_SI
204                     AB PREND_LA_VALEUR AB+1
205                     r PREND_LA_VALEUR 2
206                     FIN_SI
207                 SINON
208                     DEBUT_SINON
209                     AC PREND_LA_VALEUR AC+1
210                     r PREND_LA_VALEUR 3
211                     FIN_SINON
212                 FIN_SI
213             SI (r==2) ALORS

```

```

214     DEBUT_SI
215     r PREND_LA_VALEUR random()
216     SI (r<pBA) ALORS
217         DEBUT_SI
218         BA PREND_LA_VALEUR BA+1
219         r PREND_LA_VALEUR 1
220         FIN_SI
221     SINON
222         DEBUT_SINON
223         BC PREND_LA_VALEUR BC+1
224         r PREND_LA_VALEUR 3
225         FIN_SINON
226     FIN_SI
227 SI (r==3) ALORS
228     DEBUT_SI
229     r PREND_LA_VALEUR random()
230     SI (r<pCA) ALORS
231         DEBUT_SI
232         CA PREND_LA_VALEUR CA+1
233         r PREND_LA_VALEUR 1
234         FIN_SI
235     SINON
236         DEBUT_SINON
237         CB PREND_LA_VALEUR CB+1
238         r PREND_LA_VALEUR 2
239         FIN_SINON
240     FIN_SI
241     FIN_POUR
242 POUR i ALLANT_DE 1 A 1
243     DEBUT_POUR
244     AFFICHER "AB : "
245     AFFICHER AB
246     AFFICHER "AC : "
247     AFFICHER AC
248     AFFICHER "BA : "
249     AFFICHER BA
250     AFFICHER "BC : "
251     AFFICHER BC
252     AFFICHER "CA : "
253     AFFICHER CA
254     AFFICHER "CB : "
255     AFFICHER CB
256     FIN_POUR
257 POUR i ALLANT_DE 1 A 1
258     DEBUT_POUR
259     Bob1 PREND_LA_VALEUR floor(D*random())+1
260     SI (Bob1==1) ALORS
261         DEBUT_SI
262         AFFICHER "Bob est maintenant parti de A."
263         r PREND_LA_VALEUR random()
264         SI (r<pAB) ALORS
265             DEBUT_SI
266             Bob PREND_LA_VALEUR 2
267             FIN_SI
268         SINON
269             DEBUT_SINON
270             Bob PREND_LA_VALEUR 3
271             FIN_SINON
272         FIN_SI
273     SI (Bob1==2) ALORS
274         DEBUT_SI
275         AFFICHER "Bob est maintenant parti de B."
276         r PREND_LA_VALEUR random()
277         SI (r<pBA) ALORS
278             DEBUT_SI
279             Bob PREND_LA_VALEUR 1
280             FIN_SI
281         SINON
282             DEBUT_SINON
283             Bob PREND_LA_VALEUR 3
284             FIN_SINON
285         FIN_SI
286     SI (Bob1==3) ALORS
287         DEBUT_SI
288         AFFICHER "Bob est maintenant parti de C."
289         r PREND_LA_VALEUR random()

```

```

290 SI (r<pCA) ALORS
291   DEBUT_SI
292   Bob PREND_LA_VALEUR 1
293   FIN_SI
294   SINON
295     DEBUT_SINON
296     Bob PREND_LA_VALEUR 2
297     FIN_SINON
298   FIN_SI
299 AFFICHER "Inspecteur, vous devez décider où envoyer vos effectifs : "
300 PAUSE
301 LIRE Intervention
302 SI (Intervention=="A" OU Intervention=="a") ALORS
303   DEBUT_SI
304   i2 PREND_LA_VALEUR 1
305   FIN_SI
306 SI (Intervention=="B" OU Intervention=="b") ALORS
307   DEBUT_SI
308   i2 PREND_LA_VALEUR 2
309   FIN_SI
310 SI (Intervention=="C" OU Intervention=="c") ALORS
311   DEBUT_SI
312   i2 PREND_LA_VALEUR 3
313   FIN_SI
314 SI (i2==Bob) ALORS
315   DEBUT_SI
316   D PREND_LA_VALEUR D+1
317   AFFICHER "Bravo ! Vos agents ont capturé Bob."
318   AFFICHER "Mais réussirez-vous à le faire avec "
319   AFFICHER D
320   AFFICHER " lieux ?"
321   H PREND_LA_VALEUR 3
322   FIN_SI
323   SINON
324     DEBUT_SINON
325     H PREND_LA_VALEUR H-1
326     AFFICHER "Non, Bob n'était pas là et il risque de mettre son plan à exécution."
327     AFFICHER "Il ne vous reste plus que "
328     AFFICHER H
329     SI (H>1 OU H==0) ALORS
330       DEBUT_SI
331       AFFICHER " semaines."
332       FIN_SI
333       SINON
334         DEBUT_SINON
335         AFFICHER " semaine."
336         FIN_SINON
337     SI (H==0) ALORS
338       DEBUT_SI
339       AFFICHER "Vous avez perdu le jeu (et votre emploi...)"
340       FIN_SI
341     FIN_SINON
342   PAUSE
343   FIN_POUR
344 POUR i ALLANT_DE 1 A 1
345   DEBUT_POUR
346   AB PREND_LA_VALEUR 0
347   AC PREND_LA_VALEUR 0
348   AD PREND_LA_VALEUR 0
349   AE PREND_LA_VALEUR 0
350   BA PREND_LA_VALEUR 0
351   BC PREND_LA_VALEUR 0
352   BD PREND_LA_VALEUR 0
353   BE PREND_LA_VALEUR 0
354   CA PREND_LA_VALEUR 0
355   CB PREND_LA_VALEUR 0
356   CD PREND_LA_VALEUR 0
357   CE PREND_LA_VALEUR 0
358   DA PREND_LA_VALEUR 0
359   DB PREND_LA_VALEUR 0
360   DC PREND_LA_VALEUR 0
361   DE PREND_LA_VALEUR 0
362   EA PREND_LA_VALEUR 0
363   EB PREND_LA_VALEUR 0
364   EC PREND_LA_VALEUR 0
365   ED PREND_LA_VALEUR 0

```



```

366     FIN_POUR
367     FIN_SI
368     SI (D==4) ALORS
369         DEBUT_SI
370         T PREND_LA_VALEUR pow(D,3)
371         AFFICHER "NIVEAU III : "
372         AFFICHER "Cette semaine, Bob a fréquenté les lieux A, B, C et D dont voici la liste des transitions"
373         PAUSE
374         POUR i ALLANT_DE 1 A 1
375             DEBUT_POUR
376             pAB PREND_LA_VALEUR random()
377             pAC PREND_LA_VALEUR (1-pAB)*random()
378             pAD PREND_LA_VALEUR 1-pAB-pAC
379             pBA PREND_LA_VALEUR random()
380             pBC PREND_LA_VALEUR (1-pBA)*random()
381             pBD PREND_LA_VALEUR 1-pBA-pBC
382             pCA PREND_LA_VALEUR random()
383             pCB PREND_LA_VALEUR (1-pCA)*random()
384             pCD PREND_LA_VALEUR 1-pCA-pCB
385             pDA PREND_LA_VALEUR random()
386             pDB PREND_LA_VALEUR (1-pDA)*random()
387             pDC PREND_LA_VALEUR 1-pDA-pDB
388             r PREND_LA_VALEUR floor(4*random())+1
389             FIN_POUR
390         POUR i ALLANT_DE 1 A T
391             DEBUT_POUR
392             SI (r==1) ALORS
393                 DEBUT_SI
394                 r PREND_LA_VALEUR random()
395                 SI (r<pAB) ALORS
396                     DEBUT_SI
397                     r PREND_LA_VALEUR 2
398                     AB PREND_LA_VALEUR AB+1
399                     FIN_SI
400                 SINON
401                     DEBUT_SINON
402                     SI (r<pAB+pAC) ALORS
403                         DEBUT_SI
404                         r PREND_LA_VALEUR 3
405                         AC PREND_LA_VALEUR AC+1
406                         FIN_SI
407                     SINON
408                         DEBUT_SINON
409                         r PREND_LA_VALEUR 4
410                         AD PREND_LA_VALEUR AD+1
411                         FIN_SINON
412                     FIN_SINON
413                 FIN_SI
414             SI (r==2) ALORS
415                 DEBUT_SI
416                 r PREND_LA_VALEUR random()
417                 SI (r<pBA) ALORS
418                     DEBUT_SI
419                     BA PREND_LA_VALEUR BA+1
420                     r PREND_LA_VALEUR 1
421                     FIN_SI
422                 SINON
423                     DEBUT_SINON
424                     SI (r<pBA+pBC) ALORS
425                         DEBUT_SI
426                         BC PREND_LA_VALEUR BC+1
427                         r PREND_LA_VALEUR 3
428                         FIN_SI
429                     SINON
430                         DEBUT_SINON
431                         BD PREND_LA_VALEUR BD+1
432                         r PREND_LA_VALEUR 4
433                         FIN_SINON
434                     FIN_SINON
435                 FIN_SI
436             SI (r==3) ALORS
437                 DEBUT_SI
438                 r PREND_LA_VALEUR random()
439                 SI (r<pCA) ALORS
440                     DEBUT_SI
441                     CA PREND_LA_VALEUR CA+1

```

```

442     r PREND_LA_VALEUR 1
443     FIN_SI
444     SINON
445         DEBUT_SINON
446         SI (r<pCA+pCB) ALORS
447             DEBUT_SI
448             CB PREND_LA_VALEUR CB+1
449             r PREND_LA_VALEUR 2
450             FIN_SI
451             SINON
452                 DEBUT_SINON
453                 CD PREND_LA_VALEUR CD+1
454                 r PREND_LA_VALEUR 4
455                 FIN_SINON
456             FIN_SINON
457         FIN_SI
458     SI (r==4) ALORS
459         DEBUT_SI
460         r PREND_LA_VALEUR random()
461         SI (r<pDA) ALORS
462             DEBUT_SI
463             DA PREND_LA_VALEUR DA+1
464             r PREND_LA_VALEUR 1
465             FIN_SI
466             SINON
467                 DEBUT_SINON
468                 SI (r<pDA+pDB) ALORS
469                     DEBUT_SI
470                     DB PREND_LA_VALEUR DB+1
471                     r PREND_LA_VALEUR 2
472                     FIN_SI
473                     SINON
474                         DEBUT_SINON
475                         DC PREND_LA_VALEUR DC+1
476                         r PREND_LA_VALEUR 3
477                         FIN_SINON
478                     FIN_SINON
479                 FIN_SI
480             FIN_POUR
481     POUR i ALLANT_DE 1 A 1
482         DEBUT_POUR
483             AFFICHER "AB : "
484             AFFICHER AB
485             AFFICHER "AC : "
486             AFFICHER AC
487             AFFICHER "AD : "
488             AFFICHER AD
489             AFFICHER "BA : "
490             AFFICHER BA
491             AFFICHER "BC : "
492             AFFICHER BC
493             AFFICHER "BD : "
494             AFFICHER BD
495             AFFICHER "CA : "
496             AFFICHER CA
497             AFFICHER "CB : "
498             AFFICHER CB
499             AFFICHER "CD : "
500             AFFICHER CD
501             AFFICHER "DA : "
502             AFFICHER DA
503             AFFICHER "DB : "
504             AFFICHER DB
505             AFFICHER "DC : "
506             AFFICHER DC
507         FIN_POUR
508     POUR i ALLANT_DE 1 A 1
509         DEBUT_POUR
510             Bob1 PREND_LA_VALEUR floor(D*random()+1
511             SI (Bob1==1) ALORS
512                 DEBUT_SI
513                 AFFICHER "Bob est maintenant parti de A."
514                 r PREND_LA_VALEUR random()
515                 SI (r<pAB) ALORS
516                     DEBUT_SI
517                     Bob PREND_LA_VALEUR 2

```

```

518     FIN_SI
519     SINON
520         DEBUT_SINON
521         SI (r<pAB+pAC) ALORS
522             DEBUT_SI
523             Bob PREND_LA_VALEUR 3
524             FIN_SI
525             SINON
526                 DEBUT_SINON
527                 Bob PREND_LA_VALEUR 4
528                 FIN_SINON
529             FIN_SINON
530     FIN_SI
531     SI (Bob1==2) ALORS
532         DEBUT_SI
533         AFFICHER "Bob est maintenant parti de B."
534         r PREND_LA_VALEUR random()
535         SI (r<pBA) ALORS
536             DEBUT_SI
537             Bob PREND_LA_VALEUR 1
538             FIN_SI
539             SINON
540                 DEBUT_SINON
541                 SI (r<pBA+pBC) ALORS
542                     DEBUT_SI
543                     Bob PREND_LA_VALEUR 3
544                     FIN_SI
545                     SINON
546                         DEBUT_SINON
547                         Bob PREND_LA_VALEUR 4
548                         FIN_SINON
549                     FIN_SINON
550                 FIN_SI
551             SI (Bob1==3) ALORS
552                 DEBUT_SI
553                 AFFICHER "Bob est maintenant parti de C."
554                 r PREND_LA_VALEUR random()
555                 SI (r<pCA) ALORS
556                     DEBUT_SI
557                     Bob PREND_LA_VALEUR 1
558                     FIN_SI
559                     SINON
560                         DEBUT_SINON
561                         SI (r<pCA+pCB) ALORS
562                             DEBUT_SI
563                             Bob PREND_LA_VALEUR 2
564                             FIN_SI
565                             SINON
566                                 DEBUT_SINON
567                                 Bob PREND_LA_VALEUR 4
568                                 FIN_SINON
569                             FIN_SINON
570                         FIN_SI
571                 SI (Bob1==4) ALORS
572                     DEBUT_SI
573                     AFFICHER "Bob est maintenant parti de D."
574                     r PREND_LA_VALEUR random()
575                     SI (r<pDA) ALORS
576                         DEBUT_SI
577                         Bob PREND_LA_VALEUR 1
578                         FIN_SI
579                         SINON
580                             DEBUT_SINON
581                             SI (r<pDA+pDB) ALORS
582                                 DEBUT_SI
583                                 Bob PREND_LA_VALEUR 2
584                                 FIN_SI
585                                 SINON
586                                    DEBUT_SINON
587                                    Bob PREND_LA_VALEUR 3
588                                    FIN_SINON
589                                FIN_SINON
590                            FIN_SI
591                     AFFICHER "Inspecteur, vous devez décider où envoyer vos effectifs : "
592                     PAUSE
593                     LIRE Intervention

```

```

594 SI (Intervention=="A" OU Intervention=="a") ALORS
595   DEBUT_SI
596   i2 PREND_LA_VALEUR 1
597   FIN_SI
598 SI (Intervention=="B" OU Intervention=="b") ALORS
599   DEBUT_SI
600   i2 PREND_LA_VALEUR 2
601   FIN_SI
602 SI (Intervention=="C" OU Intervention=="c") ALORS
603   DEBUT_SI
604   i2 PREND_LA_VALEUR 3
605   FIN_SI
606 SI (Intervention=="D" OU Intervention=="d") ALORS
607   DEBUT_SI
608   i2 PREND_LA_VALEUR 4
609   FIN_SI
610 SI (Intervention=="E" OU Intervention=="e") ALORS
611   DEBUT_SI
612   i2 PREND_LA_VALEUR 5
613   FIN_SI
614 SI (i2==Bob) ALORS
615   DEBUT_SI
616   D PREND_LA_VALEUR D+1
617   AFFICHER "Bravo ! Vos agents ont capturé Bob. "
618   AFFICHER "Mais réussirez-vous à le faire avec "
619   AFFICHER D
620   AFFICHER " lieux ?"
621   H PREND_LA_VALEUR 3
622   FIN_SI
623   SINON
624     DEBUT_SINON
625     H PREND_LA_VALEUR H-1
626     AFFICHER "Non, Bob n'était pas là et il risque de mettre son plan à exécution."
627     AFFICHER "Il ne vous reste plus que "
628     AFFICHER H
629     SI (H>1 OU H==0) ALORS
630       DEBUT_SI
631       AFFICHER " semaines."
632       FIN_SI
633       SINON
634         DEBUT_SINON
635         AFFICHER " semaine."
636         FIN_SINON
637     SI (H==0) ALORS
638       DEBUT_SI
639       AFFICHER "Vous avez perdu le jeu (et votre emploi...)"
640       FIN_SI
641     FIN_SINON
642   PAUSE
643   FIN_POUR
644 POUR i ALLANT_DE 1 A 1
645   DEBUT_POUR
646   AB PREND_LA_VALEUR 0
647   AC PREND_LA_VALEUR 0
648   AD PREND_LA_VALEUR 0
649   AE PREND_LA_VALEUR 0
650   BA PREND_LA_VALEUR 0
651   BC PREND_LA_VALEUR 0
652   BD PREND_LA_VALEUR 0
653   BE PREND_LA_VALEUR 0
654   CA PREND_LA_VALEUR 0
655   CB PREND_LA_VALEUR 0
656   CD PREND_LA_VALEUR 0
657   CE PREND_LA_VALEUR 0
658   DA PREND_LA_VALEUR 0
659   DB PREND_LA_VALEUR 0
660   DC PREND_LA_VALEUR 0
661   DE PREND_LA_VALEUR 0
662   EA PREND_LA_VALEUR 0
663   EB PREND_LA_VALEUR 0
664   EC PREND_LA_VALEUR 0
665   ED PREND_LA_VALEUR 0
666   FIN_POUR
667   FIN_SI
668 SI (D==5) ALORS
669   DEBUT_SI

```

```

670 n PREND_LA_VALEUR D+1
671 T PREND_LA_VALEUR pow(D,3)
672 AFFICHER "NIVEAU IV : Final"
673 AFFICHER "Cette semaine, Bob a fréquenté les lieux A, B, C, D et E dont voici la liste des transitions"
674 PAUSE
675 POUR i ALLANT_DE 1 A 1
676   DEBUT_POUR
677     pAB PREND_LA_VALEUR random()
678     pAC PREND_LA_VALEUR (1-pAB)*random()
679     pAD PREND_LA_VALEUR (1-pAB-pAC)*random()
680     pAE PREND_LA_VALEUR 1-pAB-pAC-pAD
681     pBA PREND_LA_VALEUR random()
682     pBC PREND_LA_VALEUR (1-pBA)*random()
683     pBD PREND_LA_VALEUR (1-pBA-pBC)*random()
684     pBE PREND_LA_VALEUR 1-pBA-pBC-pBD
685     pCA PREND_LA_VALEUR random()
686     pCB PREND_LA_VALEUR (1-pCA)*random()
687     pCD PREND_LA_VALEUR (1-pCA-pCB)*random()
688     pCD PREND_LA_VALEUR 1-pCA-pCB-pCD
689     pDA PREND_LA_VALEUR random()
690     pDB PREND_LA_VALEUR (1-pDA)*random()
691     pDC PREND_LA_VALEUR (1-pDA-pDB)*random()
692     pDE PREND_LA_VALEUR 1-pDA-pDB-pDC
693     pEA PREND_LA_VALEUR random()
694     pEB PREND_LA_VALEUR (1-pEA)*random()
695     pEC PREND_LA_VALEUR (1-pEA-pEB)*random()
696     pED PREND_LA_VALEUR 1-pEA-pEB-pEC
697     r PREND_LA_VALEUR floor(D*random())+1
698     FIN_POUR
699   POUR i ALLANT_DE 1 A T
700     DEBUT_POUR
701       SI (r==1) ALORS
702         DEBUT_SI
703         r PREND_LA_VALEUR random()
704         SI (r<pAB) ALORS
705           DEBUT_SI
706           r PREND_LA_VALEUR 2
707           AB PREND_LA_VALEUR AB+1
708           FIN_SI
709         SINON
710           DEBUT_SINON
711           SI (r<pAB+pAC) ALORS
712             DEBUT_SI
713             r PREND_LA_VALEUR 3
714             AC PREND_LA_VALEUR AC+1
715             FIN_SI
716           SINON
717             DEBUT_SINON
718             SI (r<pAB+pAC+pAD) ALORS
719               DEBUT_SI
720               r PREND_LA_VALEUR 4
721               AD PREND_LA_VALEUR AD+1
722               FIN_SI
723             SINON
724               DEBUT_SINON
725               r PREND_LA_VALEUR 5
726               AE PREND_LA_VALEUR AE+1
727               FIN_SINON
728             FIN_SINON
729           FIN_SINON
730         FIN_SI
731       SI (r==2) ALORS
732         DEBUT_SI
733         r PREND_LA_VALEUR random()
734         SI (r<pBA) ALORS
735           DEBUT_SI
736           BA PREND_LA_VALEUR BA+1
737           r PREND_LA_VALEUR 1
738           FIN_SI
739         SINON
740           DEBUT_SINON
741           SI (r<pBA+pBC) ALORS
742             DEBUT_SI
743             BC PREND_LA_VALEUR BC+1
744             r PREND_LA_VALEUR 3
745             FIN_SI

```

```

746     SINON
747     DEBUT_SINON
748     SI (r<pBA+pBC+pBD) ALORS
749     DEBUT_SI
750     BD PREND_LA_VALEUR BD+1
751     r PREND_LA_VALEUR 4
752     FIN_SI
753     SINON
754     DEBUT_SINON
755     BE PREND_LA_VALEUR BE+1
756     r PREND_LA_VALEUR 5
757     FIN_SINON
758     FIN_SINON
759     FIN_SINON
760     FIN_SI
761     SI (r==3) ALORS
762     DEBUT_SI
763     r PREND_LA_VALEUR random()
764     SI (r<pCA) ALORS
765     DEBUT_SI
766     CA PREND_LA_VALEUR CA+1
767     r PREND_LA_VALEUR 1
768     FIN_SI
769     SINON
770     DEBUT_SINON
771     SI (r<pCA+pCB) ALORS
772     DEBUT_SI
773     CB PREND_LA_VALEUR CB+1
774     r PREND_LA_VALEUR 2
775     FIN_SI
776     SINON
777     DEBUT_SINON
778     SI (r<pCA+pCB+pCD) ALORS
779     DEBUT_SI
780     CD PREND_LA_VALEUR CD+1
781     r PREND_LA_VALEUR 4
782     FIN_SI
783     SINON
784     DEBUT_SINON
785     CE PREND_LA_VALEUR CE+1
786     r PREND_LA_VALEUR 5
787     FIN_SINON
788     FIN_SINON
789     FIN_SINON
790     FIN_SI
791     SI (r==4) ALORS
792     DEBUT_SI
793     r PREND_LA_VALEUR random()
794     SI (r<pDA) ALORS
795     DEBUT_SI
796     DA PREND_LA_VALEUR DA+1
797     r PREND_LA_VALEUR 1
798     FIN_SI
799     SINON
800     DEBUT_SINON
801     SI (r<pDA+pDB) ALORS
802     DEBUT_SI
803     DB PREND_LA_VALEUR DB+1
804     r PREND_LA_VALEUR 2
805     FIN_SI
806     SINON
807     DEBUT_SINON
808     SI (r<pDA+pDB+pDC) ALORS
809     DEBUT_SI
810     DC PREND_LA_VALEUR DC+1
811     r PREND_LA_VALEUR 3
812     FIN_SI
813     SINON
814     DEBUT_SINON
815     DE PREND_LA_VALEUR DE+1
816     r PREND_LA_VALEUR 5
817     FIN_SINON
818     FIN_SINON
819     FIN_SINON
820     FIN_SI
821     SI (r==5) ALORS

```

```

822     DEBUT_SI
823     r PREND_LA_VALEUR random()
824     SI (r<pEA) ALORS
825         DEBUT_SI
826         EA PREND_LA_VALEUR EA+1
827         r PREND_LA_VALEUR 1
828         FIN_SI
829     SINON
830         DEBUT_SINON
831         SI (r<pEA+pEB) ALORS
832             DEBUT_SI
833             EB PREND_LA_VALEUR EB+1
834             r PREND_LA_VALEUR 2
835             FIN_SI
836         SINON
837             DEBUT_SINON
838             SI (r<pEA+pEB+pEC) ALORS
839                 DEBUT_SI
840                 EC PREND_LA_VALEUR EC+1
841                 r PREND_LA_VALEUR 3
842                 FIN_SI
843             SINON
844                 DEBUT_SINON
845                 ED PREND_LA_VALEUR ED+1
846                 r PREND_LA_VALEUR 4
847                 FIN_SINON
848             FIN_SINON
849         FIN_SINON
850     FIN_SI
851     FIN_POUR
852     POUR i ALLANT_DE 1 A 1
853     DEBUT_POUR
854     AFFICHER "AB : "
855     AFFICHER AB
856     AFFICHER "AC : "
857     AFFICHER AC
858     AFFICHER "AD : "
859     AFFICHER AD
860     AFFICHER "AE : "
861     AFFICHER AE
862     AFFICHER "BA : "
863     AFFICHER BA
864     AFFICHER "BC : "
865     AFFICHER BC
866     AFFICHER "BD : "
867     AFFICHER BD
868     AFFICHER "BE : "
869     AFFICHER BE
870     AFFICHER "CA : "
871     AFFICHER CA
872     AFFICHER "CB : "
873     AFFICHER CB
874     AFFICHER "CD : "
875     AFFICHER CD
876     AFFICHER "CE : "
877     AFFICHER CE
878     AFFICHER "DA : "
879     AFFICHER DA
880     AFFICHER "DB : "
881     AFFICHER DB
882     AFFICHER "DC : "
883     AFFICHER DC
884     AFFICHER "DE : "
885     AFFICHER DE
886     AFFICHER "EA : "
887     AFFICHER EA
888     AFFICHER "EB : "
889     AFFICHER EB
890     AFFICHER "EC : "
891     AFFICHER EC
892     AFFICHER "ED : "
893     AFFICHER ED
894     FIN_POUR
895     POUR i ALLANT_DE 1 A 1
896     DEBUT_POUR
897     Bob1 PREND_LA_VALEUR floor(D*random()+1

```



```

898 SI (Bob1==1) ALORS
899   DEBUT_SI
900   AFFICHER "Bob est maintenant parti de A."
901   r PREND_LA_VALEUR random()
902   SI (r<pAB) ALORS
903     DEBUT_SI
904     Bob PREND_LA_VALEUR 2
905     FIN_SI
906   SINON
907     DEBUT_SINON
908     SI (r<pAB+pAC) ALORS
909       DEBUT_SI
910       Bob PREND_LA_VALEUR 3
911       FIN_SI
912     SINON
913       DEBUT_SINON
914       SI (r<pAB+pAC+pAD) ALORS
915         DEBUT_SI
916         Bob PREND_LA_VALEUR 4
917         FIN_SI
918       SINON
919         DEBUT_SINON
920         Bob PREND_LA_VALEUR 5
921         FIN_SINON
922       FIN_SINON
923     FIN_SINON
924   FIN_SI
925 SI (Bob1==2) ALORS
926   DEBUT_SI
927   AFFICHER "Bob est maintenant parti de B."
928   r PREND_LA_VALEUR random()
929   SI (r<pBA) ALORS
930     DEBUT_SI
931     Bob PREND_LA_VALEUR 1
932     FIN_SI
933   SINON
934     DEBUT_SINON
935     SI (r<pBA+pBC) ALORS
936       DEBUT_SI
937       Bob PREND_LA_VALEUR 3
938       FIN_SI
939     SINON
940       DEBUT_SINON
941       SI (r<pBA+pBC+pBD) ALORS
942         DEBUT_SI
943         Bob PREND_LA_VALEUR 4
944         FIN_SI
945       SINON
946         DEBUT_SINON
947         Bob PREND_LA_VALEUR 5
948         FIN_SINON
949       FIN_SINON
950     FIN_SINON
951   FIN_SI
952 SI (Bob1==3) ALORS
953   DEBUT_SI
954   AFFICHER "Bob est maintenant parti de C."
955   r PREND_LA_VALEUR random()
956   SI (r<pCA) ALORS
957     DEBUT_SI
958     Bob PREND_LA_VALEUR 1
959     FIN_SI
960   SINON
961     DEBUT_SINON
962     SI (r<pCA+pCB) ALORS
963       DEBUT_SI
964       Bob PREND_LA_VALEUR 2
965       FIN_SI
966     SINON
967       DEBUT_SINON
968       SI (r<pCA+pCB+pCD) ALORS
969         DEBUT_SI
970         Bob PREND_LA_VALEUR 4
971         FIN_SI
972       SINON
973         DEBUT_SINON

```

```

974         Bob PREND_LA_VALEUR 5
975     FIN_SINON
976     FIN_SINON
977     FIN_SINON
978     FIN_SI
979 SI (Bob1==4) ALORS
980     DEBUT_SI
981     AFFICHER "Bob est maintenant parti de D."
982     r PREND_LA_VALEUR random()
983     SI (r<pDA) ALORS
984         DEBUT_SI
985         Bob PREND_LA_VALEUR 1
986         FIN_SI
987     SINON
988         DEBUT_SINON
989         SI (r<pDA+pDB) ALORS
990             DEBUT_SI
991             Bob PREND_LA_VALEUR 2
992             FIN_SI
993         SINON
994             DEBUT_SINON
995             SI (r<pDA+pDB+pDC) ALORS
996                 DEBUT_SI
997                 Bob PREND_LA_VALEUR 3
998                 FIN_SI
999             SINON
1000                 DEBUT_SINON
1001                 Bob PREND_LA_VALEUR 5
1002             FIN_SINON
1003         FIN_SINON
1004     FIN_SINON
1005     FIN_SI
1006 SI (Bob1==5) ALORS
1007     DEBUT_SI
1008     AFFICHER "Bob est maintenant parti de E."
1009     r PREND_LA_VALEUR random()
1010     SI (r<pEA) ALORS
1011         DEBUT_SI
1012         Bob PREND_LA_VALEUR 1
1013         FIN_SI
1014     SINON
1015         DEBUT_SINON
1016         SI (r<pEA+pEB) ALORS
1017             DEBUT_SI
1018             Bob PREND_LA_VALEUR 2
1019             FIN_SI
1020         SINON
1021             DEBUT_SINON
1022             SI (r<pEA+pEB+pEC) ALORS
1023                 DEBUT_SI
1024                 Bob PREND_LA_VALEUR 3
1025                 FIN_SI
1026             SINON
1027                 DEBUT_SINON
1028                 Bob PREND_LA_VALEUR 4
1029                 FIN_SINON
1030             FIN_SINON
1031         FIN_SINON
1032     FIN_SI
1033 AFFICHER "Inspecteur, vous devez décider où envoyer vos effectifs : "
1034 PAUSE
1035 LIRE Intervention
1036 SI (Intervention=="A" OU Intervention=="a") ALORS
1037     DEBUT_SI
1038     i2 PREND_LA_VALEUR 1
1039     FIN_SI
1040 SI (Intervention=="B" OU Intervention=="b") ALORS
1041     DEBUT_SI
1042     i2 PREND_LA_VALEUR 2
1043     FIN_SI
1044 SI (Intervention=="C" OU Intervention=="c") ALORS
1045     DEBUT_SI
1046     i2 PREND_LA_VALEUR 3
1047     FIN_SI
1048 SI (Intervention=="D" OU Intervention=="d") ALORS
1049     DEBUT_SI

```

```

1050     i2 PREND_LA_VALEUR 4
1051     FIN_SI
1052     SI (Intervention=="E" OU Intervention=="e") ALORS
1053         DEBUT_SI
1054         i2 PREND_LA_VALEUR 5
1055         FIN_SI
1056     SI (i2==Bob) ALORS
1057         DEBUT_SI
1058         AFFICHER "Bravo ! Vos agents ont capturé Bob. "
1059         AFFICHER "Vous êtes parvenus à stopper Bob 4 fois et devenez commissaire."
1060         PAUSE
1061         AFFICHER "Conclusion du jeu : "
1062         AFFICHER "Il est très facile de prévoir les déplacements d'une personne."
1063         AFFICHER "Mais ceux-ci peuvent parfois être considérés comme aléatoires car même en allant"
1064         AFFICHER "sur le lieu le plus probable, vous n'avez peut-être pas toujours attrapé Bob."
1065         AFFICHER "Pour recommencer à jouer, taper OUI, sinon autre chose."
1066     LIRE Continuer
1067     SI (Continuer=="OUI" OU Continuer=="Oui" OU Continuer=="oui") ALORS
1068         DEBUT_SI
1069         H PREND_LA_VALEUR 3
1070         D PREND_LA_VALEUR 2
1071         AFFICHER " "
1072         AFFICHER " "
1073         FIN_SI
1074         SINON
1075             DEBUT_SINON
1076             H PREND_LA_VALEUR 0
1077             FIN_SINON
1078         FIN_SI
1079         SINON
1080             DEBUT_SINON
1081             H PREND_LA_VALEUR H-1
1082             AFFICHER "Non, Bob n'était pas là et il risque de mettre son plan à exécution."
1083             AFFICHER "Il ne vous reste plus que "
1084             AFFICHER H
1085             SI (H>1 OU H==0) ALORS
1086                 DEBUT_SI
1087                 AFFICHER " semaines."
1088                 FIN_SI
1089                 SINON
1090                     DEBUT_SINON
1091                     AFFICHER " semaine."
1092                     FIN_SINON
1093             SI (H==0) ALORS
1094                 DEBUT_SI
1095                 AFFICHER "Vous avez perdu le jeu (et votre emploi...)"
1096                 FIN_SI
1097                 PAUSE
1098                 FIN_SINON
1099         FIN_POUR
1100     POUR i ALLANT_DE 1 A 1
1101         DEBUT_POUR
1102         AB PREND_LA_VALEUR 0
1103         AC PREND_LA_VALEUR 0
1104         AD PREND_LA_VALEUR 0
1105         AE PREND_LA_VALEUR 0
1106         BA PREND_LA_VALEUR 0
1107         BC PREND_LA_VALEUR 0
1108         BD PREND_LA_VALEUR 0
1109         BE PREND_LA_VALEUR 0
1110         CA PREND_LA_VALEUR 0
1111         CB PREND_LA_VALEUR 0
1112         CD PREND_LA_VALEUR 0
1113         CE PREND_LA_VALEUR 0
1114         DA PREND_LA_VALEUR 0
1115         DB PREND_LA_VALEUR 0
1116         DC PREND_LA_VALEUR 0
1117         DE PREND_LA_VALEUR 0
1118         EA PREND_LA_VALEUR 0
1119         EB PREND_LA_VALEUR 0
1120         EC PREND_LA_VALEUR 0
1121         ED PREND_LA_VALEUR 0
1122         FIN_POUR
1123     FIN_SI
1124     FIN_TANT_QUE
1125

```

**3/ Marge d'erreur ergonomisée**

```

1  VARIABLES
2  AB EST_DU_TYPE NOMBRE
3  AC EST_DU_TYPE NOMBRE
4  BA EST_DU_TYPE NOMBRE
5  BC EST_DU_TYPE NOMBRE
6  CA EST_DU_TYPE NOMBRE
7  CB EST_DU_TYPE NOMBRE
8  r EST_DU_TYPE NOMBRE
9  n EST_DU_TYPE NOMBRE
10 A EST_DU_TYPE NOMBRE
11 B EST_DU_TYPE NOMBRE
12 C EST_DU_TYPE NOMBRE
13 i EST_DU_TYPE NOMBRE
14 pA EST_DU_TYPE NOMBRE
15 pB EST_DU_TYPE NOMBRE
16 pC EST_DU_TYPE NOMBRE
17 t EST_DU_TYPE NOMBRE
18 q EST_DU_TYPE NOMBRE
19 fA EST_DU_TYPE NOMBRE
20 fB EST_DU_TYPE NOMBRE
21 fC EST_DU_TYPE NOMBRE
22 rr EST_DU_TYPE NOMBRE
23 TT EST_DU_TYPE NOMBRE
24 mA EST_DU_TYPE NOMBRE
25 mB EST_DU_TYPE NOMBRE
26 mC EST_DU_TYPE NOMBRE
27 M EST_DU_TYPE NOMBRE
28 o EST_DU_TYPE NOMBRE
29 u EST_DU_TYPE LISTE
30 d EST_DU_TYPE NOMBRE
31 DEBUT_ALGORITHME
32 AFFICHER "Nombre q de chaînes à créer (<20)"
33 LIRE q
34 AFFICHER "Nombre o de tests par chaîne (pas trop)"
35 LIRE o
36 AFFICHER "Suite des tests"
37 AFFICHER " "
38 POUR t ALLANT_DE 1 A o
39   DEBUT_POUR
40     LIRE u[t]
41   FIN_POUR
42 POUR t ALLANT_DE 1 A q
43   DEBUT_POUR
44     AFFICHER "      Chaîne n°"
45     AFFICHER t
46     AB PREND_LA_VALEUR random()
47     AC PREND_LA_VALEUR 1-AB
48     BA PREND_LA_VALEUR random()
49     BC PREND_LA_VALEUR 1-BA
50     CA PREND_LA_VALEUR random()
51     CB PREND_LA_VALEUR 1-CA
52     pC PREND_LA_VALEUR ((AB*BA)-1)/(AB*(BA-CA)-CB*(BA+1)-CA-1)
53     pB PREND_LA_VALEUR (1-pC*(CA+1))/(BA+1)
54     pA PREND_LA_VALEUR 1-pB-pC
55     POUR d ALLANT_DE 1 A o
56       DEBUT_POUR
57         r PREND_LA_VALEUR floor(3*random())
58         A PREND_LA_VALEUR 0
59         B PREND_LA_VALEUR 0
60         C PREND_LA_VALEUR 0
61         POUR i ALLANT_DE 1 A u[d]
62           DEBUT_POUR
63             SI (r==0) ALORS
64               DEBUT_SI
65                 n PREND_LA_VALEUR random()
66                 A PREND_LA_VALEUR A+1
67                 SI (n<AB) ALORS
68                   DEBUT_SI
69                     rr PREND_LA_VALEUR 1
70                   FIN_SI
71                 SINON
72                   DEBUT_SINON

```

```

73     rr PREND_LA_VALEUR 2
74     FIN_SINON
75     FIN_SI
76     SI (r==1) ALORS
77     DEBUT_SI
78     n PREND_LA_VALEUR random()
79     B PREND_LA_VALEUR B+1
80     SI (n<BA) ALORS
81     DEBUT_SI
82     rr PREND_LA_VALEUR 0
83     FIN_SI
84     SINON
85     DEBUT_SINON
86     rr PREND_LA_VALEUR 2
87     FIN_SINON
88     FIN_SI
89     SI (r==2) ALORS
90     DEBUT_SI
91     n PREND_LA_VALEUR random()
92     C PREND_LA_VALEUR C+1
93     SI (n<CA) ALORS
94     DEBUT_SI
95     rr PREND_LA_VALEUR 0
96     FIN_SI
97     SINON
98     DEBUT_SINON
99     rr PREND_LA_VALEUR 1
100    FIN_SINON
101    FIN_SI
102    SI (rr==0) ALORS
103    DEBUT_SI
104    r PREND_LA_VALEUR 0
105    FIN_SI
106    SI (rr==1) ALORS
107    DEBUT_SI
108    r PREND_LA_VALEUR 1
109    FIN_SI
110    SI (rr==2) ALORS
111    DEBUT_SI
112    r PREND_LA_VALEUR 2
113    FIN_SI
114    FIN_POUR
115    TT PREND_LA_VALEUR A+B+C
116    fA PREND_LA_VALEUR A/TT
117    fB PREND_LA_VALEUR B/TT
118    fC PREND_LA_VALEUR C/TT
119    mA PREND_LA_VALEUR abs((pA-fA)/pA)
120    mB PREND_LA_VALEUR abs((pB-fB)/pB)
121    mC PREND_LA_VALEUR abs((pC-fC)/pC)
122    M PREND_LA_VALEUR (mA+mB+mC)*100/3
123    AFFICHER u[d]
124    AFFICHER " transitions : "
125    AFFICHER M
126    AFFICHER "% de différence"
127    FIN_POUR
128    AFFICHER " "
129    FIN_POUR
130    FIN_ALGORITHME

```

#### **4/ Marge d'erreur moyenne**

```

1  VARIABLES
2  AB EST_DU_TYPE NOMBRE
3  AC EST_DU_TYPE NOMBRE
4  BA EST_DU_TYPE NOMBRE
5  BC EST_DU_TYPE NOMBRE
6  CA EST_DU_TYPE NOMBRE
7  CB EST_DU_TYPE NOMBRE
8  r EST_DU_TYPE NOMBRE
9  n EST_DU_TYPE NOMBRE
10 A EST_DU_TYPE NOMBRE
11 B EST_DU_TYPE NOMBRE

```

```

12 C EST_DU_TYPE NOMBRE
13 pA EST_DU_TYPE NOMBRE
14 pB EST_DU_TYPE NOMBRE
15 pC EST_DU_TYPE NOMBRE
16 t EST_DU_TYPE NOMBRE
17 q EST_DU_TYPE NOMBRE
18 fA EST_DU_TYPE NOMBRE
19 fB EST_DU_TYPE NOMBRE
20 fC EST_DU_TYPE NOMBRE
21 rr EST_DU_TYPE NOMBRE
22 TT EST_DU_TYPE NOMBRE
23 mA EST_DU_TYPE NOMBRE
24 mB EST_DU_TYPE NOMBRE
25 mC EST_DU_TYPE NOMBRE
26 M EST_DU_TYPE NOMBRE
27 o EST_DU_TYPE NOMBRE
28 g EST_DU_TYPE NOMBRE
29 S EST_DU_TYPE NOMBRE
30 i EST_DU_TYPE NOMBRE
31 u EST_DU_TYPE LISTE
32 DEBUT_ALGORITHME
33 LIRE q
34 LIRE g
35 S20 PREND_LA_VALEUR 0
36 POUR t ALLANT_DE 1 A q
37   DEBUT_POUR
38   AB PREND_LA_VALEUR random()
39   AC PREND_LA_VALEUR 1-AB
40   BA PREND_LA_VALEUR random()
41   BC PREND_LA_VALEUR 1-BA
42   CA PREND_LA_VALEUR random()
43   CB PREND_LA_VALEUR 1-CA
44   pC PREND_LA_VALEUR ((AB*BA)-1)/((AB*(BA-CA)-CB*(BA+1)-CA-1)
45   pB PREND_LA_VALEUR (1-pC*(CA+1))/(BA+1)
46   pA PREND_LA_VALEUR 1-pB-pC
47   POUR o ALLANT_DE 1 A 1
48   DEBUT_POUR
49   r PREND_LA_VALEUR floor(3*random())
50   A PREND_LA_VALEUR 0
51   B PREND_LA_VALEUR 0
52   C PREND_LA_VALEUR 0
53   POUR i ALLANT_DE 1 A g
54   DEBUT_POUR
55   SI (r==0) ALORS
56   DEBUT_SI
57   n PREND_LA_VALEUR random()
58   A PREND_LA_VALEUR A+1
59   SI (n<AB) ALORS
60   DEBUT_SI
61   rr PREND_LA_VALEUR 1
62   FIN_SI
63   SINON
64   DEBUT_SINON
65   rr PREND_LA_VALEUR 2
66   FIN_SINON
67   FIN_SI
68   SI (r==1) ALORS
69   DEBUT_SI
70   n PREND_LA_VALEUR random()
71   B PREND_LA_VALEUR B+1
72   SI (n<BA) ALORS
73   DEBUT_SI
74   rr PREND_LA_VALEUR 0
75   FIN_SI
76   SINON
77   DEBUT_SINON
78   rr PREND_LA_VALEUR 2
79   FIN_SINON
80   FIN_SI
81   SI (r==2) ALORS
82   DEBUT_SI
83   n PREND_LA_VALEUR random()
84   C PREND_LA_VALEUR C+1
85   SI (n<CA) ALORS
86   DEBUT_SI
87   rr PREND_LA_VALEUR 0

```

```

88     FIN_SI
89     SINON
90     DEBUT_SINON
91     rr PREND_LA_VALEUR 1
92     FIN_SINON
93     FIN_SI
94     SI (rr==0) ALORS
95     DEBUT_SI
96     r PREND_LA_VALEUR 0
97     FIN_SI
98     SI (rr==1) ALORS
99     DEBUT_SI
100    r PREND_LA_VALEUR 1
101    FIN_SI
102    SI (rr==2) ALORS
103    DEBUT_SI
104    r PREND_LA_VALEUR 2
105    FIN_SI
106    FIN_POUR
107    TT PREND_LA_VALEUR A+B+C
108    fA PREND_LA_VALEUR A/TT
109    fB PREND_LA_VALEUR B/TT
110    fC PREND_LA_VALEUR C/TT
111    mA PREND_LA_VALEUR abs((pA-fA)/pA)
112    mB PREND_LA_VALEUR abs((pB-fB)/pB)
113    mC PREND_LA_VALEUR abs((pC-fC)/pC)
114    M PREND_LA_VALEUR (mA+mB+mC)*100/3
115    S PREND_LA_VALEUR S+M
116    FIN_POUR
117    FIN_POUR
118    S PREND_LA_VALEUR S/q
119    AFFICHER "Marge d'erreur moyenne à "
120    AFFICHER g
121    AFFICHER " transitions"
122    AFFICHER S
123    AFFICHER " %"
124    FIN_ALGORITHME

```

## **5/ Graphe 1**

```

1  VARIABLES
2  AB EST_DU_TYPE NOMBRE
3  AC EST_DU_TYPE NOMBRE
4  BA EST_DU_TYPE NOMBRE
5  BC EST_DU_TYPE NOMBRE
6  CA EST_DU_TYPE NOMBRE
7  CB EST_DU_TYPE NOMBRE
8  r EST_DU_TYPE NOMBRE
9  n EST_DU_TYPE NOMBRE
10 A EST_DU_TYPE NOMBRE
11 B EST_DU_TYPE NOMBRE
12 C EST_DU_TYPE NOMBRE
13 pA EST_DU_TYPE NOMBRE
14 pB EST_DU_TYPE NOMBRE
15 pC EST_DU_TYPE NOMBRE
16 t EST_DU_TYPE NOMBRE
17 fA EST_DU_TYPE NOMBRE
18 fB EST_DU_TYPE NOMBRE
19 fC EST_DU_TYPE NOMBRE
20 rr EST_DU_TYPE NOMBRE
21 TT EST_DU_TYPE NOMBRE
22 mA EST_DU_TYPE NOMBRE
23 mB EST_DU_TYPE NOMBRE
24 mC EST_DU_TYPE NOMBRE
25 M EST_DU_TYPE NOMBRE
26 S EST_DU_TYPE NOMBRE
27 i EST_DU_TYPE NOMBRE
28 u EST_DU_TYPE LISTE

```

```

29  n1 EST_DU_TYPE NOMBRE
30  n2 EST_DU_TYPE NOMBRE
31  n3 EST_DU_TYPE NOMBRE
32  q EST_DU_TYPE NOMBRE
33  p EST_DU_TYPE NOMBRE
34  Mo EST_DU_TYPE NOMBRE
35  Max EST_DU_TYPE NOMBRE
36  Maxpo EST_DU_TYPE NOMBRE
37  Su EST_DU_TYPE NOMBRE
38  DEBUT_ALGORITHME
39  AFFICHER "Nombre de transitions au 1er test : "
40  LIRE n1
41  AFFICHER n1
42  AFFICHER "Nombre de transitions au 2nd test : "
43  LIRE n2
44  AFFICHER n2
45  AFFICHER "Nombre de transitions au 3ème test : "
46  LIRE n3
47  AFFICHER n3
48  AFFICHER "Nombre de chaînes testées : "
49  LIRE q
50  AFFICHER q
51  AFFICHER " "
52  SI (n1!=0) ALORS
53    DEBUT_SI
54      POUR i ALLANT_DE 0 A 99
55        DEBUT_POUR
56          u[i] PREND_LA_VALEUR 0
57          FIN_POUR
58      POUR t ALLANT_DE 1 A q
59        DEBUT_POUR
60          AB PREND_LA_VALEUR random()
61          AC PREND_LA_VALEUR 1-AB
62          BA PREND_LA_VALEUR random()
63          BC PREND_LA_VALEUR 1-BA
64          CA PREND_LA_VALEUR random()
65          CB PREND_LA_VALEUR 1-CA
66          pC PREND_LA_VALEUR ((AB*BA)-1)/(AB*(BA-CA)-CB*(BA+1)-CA-1)
67          pB PREND_LA_VALEUR (1-pC*(CA+1))/(BA+1)
68          pA PREND_LA_VALEUR 1-pB-pC
69          POUR o ALLANT_DE 1 A 1
70            DEBUT_POUR
71              r PREND_LA_VALEUR floor(3*random())
72              A PREND_LA_VALEUR 0
73              B PREND_LA_VALEUR 0
74              C PREND_LA_VALEUR 0
75              POUR i ALLANT_DE 1 A n1
76                DEBUT_POUR
77                  SI (r==0) ALORS
78                    DEBUT_SI
79                      n PREND_LA_VALEUR random()
80                      A PREND_LA_VALEUR A+1
81                      SI (n<AB) ALORS
82                        DEBUT_SI
83                          rr PREND_LA_VALEUR 1
84                          FIN_SI
85                          SINON
86                            DEBUT_SINON
87                              rr PREND_LA_VALEUR 2
88                              FIN_SINON
89                          FIN_SI
90                      SI (r==1) ALORS
91                        DEBUT_SI
92                          n PREND_LA_VALEUR random()
93                          B PREND_LA_VALEUR B+1
94                          SI (n<BA) ALORS
95                            DEBUT_SI
96                              rr PREND_LA_VALEUR 0
97                              FIN_SI
98                              SINON
99                                DEBUT_SINON
100                                  rr PREND_LA_VALEUR 2
101                                  FIN_SINON
102                              FIN_SI
103                          SI (r==2) ALORS
104                            DEBUT_SI

```



```

105     n PREND_LA_VALEUR random()
106     C PREND_LA_VALEUR C+1
107     SI (n<CA) ALORS
108         DEBUT_SI
109         rr PREND_LA_VALEUR 0
110         FIN_SI
111         SINON
112             DEBUT_SINON
113             rr PREND_LA_VALEUR 1
114             FIN_SINON
115         FIN_SI
116     SI (rr==0) ALORS
117         DEBUT_SI
118         r PREND_LA_VALEUR 0
119         FIN_SI
120     SI (rr==1) ALORS
121         DEBUT_SI
122         r PREND_LA_VALEUR 1
123         FIN_SI
124     SI (rr==2) ALORS
125         DEBUT_SI
126         r PREND_LA_VALEUR 2
127         FIN_SI
128     FIN_POUR
129     TT PREND_LA_VALEUR A+B+C
130     fA PREND_LA_VALEUR A/TT
131     fB PREND_LA_VALEUR B/TT
132     fC PREND_LA_VALEUR C/TT
133     mA PREND_LA_VALEUR abs((pA-fA)/pA)
134     mB PREND_LA_VALEUR abs((pB-fB)/pB)
135     mC PREND_LA_VALEUR abs((pC-fC)/pC)
136     M PREND_LA_VALEUR (mA+mB+mC)*100/3
137     S PREND_LA_VALEUR S+M
138     FIN_POUR
139     POUR i ALLANT_DE 0 A 99
140     DEBUT_POUR
141     SI (M>=i ET M<(i+1)) ALORS
142         DEBUT_SI
143         u[i] PREND_LA_VALEUR u[i]+1
144         FIN_SI
145     FIN_POUR
146     FIN_POUR
147     POUR i ALLANT_DE 0 A 99
148     DEBUT_POUR
149     SI (u[i]!=0) ALORS
150         DEBUT_SI
151         TRACER_POINT (i,u[i]/q)
152         FIN_SI
153     FIN_POUR
154     FIN_SI
155     SI (n2!=0) ALORS
156     DEBUT_SI
157     POUR i ALLANT_DE 0 A 99
158     DEBUT_POUR
159     u[i] PREND_LA_VALEUR 0
160     FIN_POUR
161     POUR t ALLANT_DE 1 A q
162     DEBUT_POUR
163     AB PREND_LA_VALEUR random()
164     AC PREND_LA_VALEUR 1-AB
165     BA PREND_LA_VALEUR random()
166     BC PREND_LA_VALEUR 1-BA
167     CA PREND_LA_VALEUR random()
168     CB PREND_LA_VALEUR 1-CA
169     pC PREND_LA_VALEUR ((AB*BA)-1)/((AB*(BA-CA)-CB*(BA+1)-CA-1)
170     pB PREND_LA_VALEUR (1-pC*(CA+1))/(BA+1)
171     pA PREND_LA_VALEUR 1-pB-pC
172     POUR o ALLANT_DE 1 A 1
173     DEBUT_POUR
174     r PREND_LA_VALEUR floor(3*random())
175     A PREND_LA_VALEUR 0
176     B PREND_LA_VALEUR 0
177     C PREND_LA_VALEUR 0
178     POUR i ALLANT_DE 1 A n2
179     DEBUT_POUR
180     SI (r==0) ALORS

```

```

181     DEBUT_SI
182     n PREND_LA_VALEUR random()
183     A PREND_LA_VALEUR A+1
184     SI (n<AB) ALORS
185         DEBUT_SI
186         rr PREND_LA_VALEUR 1
187         FIN_SI
188         SINON
189             DEBUT_SINON
190             rr PREND_LA_VALEUR 2
191             FIN_SINON
192         FIN_SI
193     SI (r==1) ALORS
194         DEBUT_SI
195         n PREND_LA_VALEUR random()
196         B PREND_LA_VALEUR B+1
197         SI (n<BA) ALORS
198             DEBUT_SI
199             rr PREND_LA_VALEUR 0
200             FIN_SI
201             SINON
202                 DEBUT_SINON
203                 rr PREND_LA_VALEUR 2
204                 FIN_SINON
205             FIN_SI
206     SI (r==2) ALORS
207         DEBUT_SI
208         n PREND_LA_VALEUR random()
209         C PREND_LA_VALEUR C+1
210         SI (n<CA) ALORS
211             DEBUT_SI
212             rr PREND_LA_VALEUR 0
213             FIN_SI
214             SINON
215                 DEBUT_SINON
216                 rr PREND_LA_VALEUR 1
217                 FIN_SINON
218             FIN_SI
219     SI (rr==0) ALORS
220         DEBUT_SI
221         r PREND_LA_VALEUR 0
222         FIN_SI
223     SI (rr==1) ALORS
224         DEBUT_SI
225         r PREND_LA_VALEUR 1
226         FIN_SI
227     SI (rr==2) ALORS
228         DEBUT_SI
229         r PREND_LA_VALEUR 2
230         FIN_SI
231     FIN_POUR
232     TT PREND_LA_VALEUR A+B+C
233     fA PREND_LA_VALEUR A/TT
234     fB PREND_LA_VALEUR B/TT
235     fC PREND_LA_VALEUR C/TT
236     mA PREND_LA_VALEUR abs((pA-fA)/pA)
237     mB PREND_LA_VALEUR abs((pB-fB)/pB)
238     mC PREND_LA_VALEUR abs((pC-fC)/pC)
239     M PREND_LA_VALEUR (mA+mB+mC)*100/3
240     S PREND_LA_VALEUR S+M
241     FIN_POUR
242     POUR i ALLANT_DE 0 A 99
243         DEBUT_POUR
244         SI (M>=i ET M<(i+1)) ALORS
245             DEBUT_SI
246             u[i] PREND_LA_VALEUR u[i]+1
247             FIN_SI
248         FIN_POUR
249     FIN_POUR
250     POUR i ALLANT_DE 0 A 99
251         DEBUT_POUR
252         SI (u[i]!=0) ALORS
253             DEBUT_SI
254             TRACER_POINT (i,u[i]/q)
255             FIN_SI
256         FIN_POUR

```

```

257  FIN_SI
258  SI (n3!=0) ALORS
259  DEBUT_SI
260  POUR i ALLANT_DE 0 A 99
261  DEBUT_POUR
262  u[i] PREND_LA_VALEUR 0
263  FIN_POUR
264  POUR t ALLANT_DE 1 A q
265  DEBUT_POUR
266  AB PREND_LA_VALEUR random()
267  AC PREND_LA_VALEUR 1-AB
268  BA PREND_LA_VALEUR random()
269  BC PREND_LA_VALEUR 1-BA
270  CA PREND_LA_VALEUR random()
271  CB PREND_LA_VALEUR 1-CA
272  pC PREND_LA_VALEUR ((AB*BA)-1)/((AB*(BA-CA)-CB*(BA+1)-CA-1)
273  pB PREND_LA_VALEUR (1-pC*(CA+1))/(BA+1)
274  pA PREND_LA_VALEUR 1-pB-pC
275  POUR o ALLANT_DE 1 A 1
276  DEBUT_POUR
277  r PREND_LA_VALEUR floor(3*random())
278  A PREND_LA_VALEUR 0
279  B PREND_LA_VALEUR 0
280  C PREND_LA_VALEUR 0
281  POUR i ALLANT_DE 1 A n3
282  DEBUT_POUR
283  SI (r==0) ALORS
284  DEBUT_SI
285  n PREND_LA_VALEUR random()
286  A PREND_LA_VALEUR A+1
287  SI (n<AB) ALORS
288  DEBUT_SI
289  rr PREND_LA_VALEUR 1
290  FIN_SI
291  SINON
292  DEBUT_SINON
293  rr PREND_LA_VALEUR 2
294  FIN_SINON
295  FIN_SI
296  SI (r==1) ALORS
297  DEBUT_SI
298  n PREND_LA_VALEUR random()
299  B PREND_LA_VALEUR B+1
300  SI (n<BA) ALORS
301  DEBUT_SI
302  rr PREND_LA_VALEUR 0
303  FIN_SI
304  SINON
305  DEBUT_SINON
306  rr PREND_LA_VALEUR 2
307  FIN_SINON
308  FIN_SI
309  SI (r==2) ALORS
310  DEBUT_SI
311  n PREND_LA_VALEUR random()
312  C PREND_LA_VALEUR C+1
313  SI (n<CA) ALORS
314  DEBUT_SI
315  rr PREND_LA_VALEUR 0
316  FIN_SI
317  SINON
318  DEBUT_SINON
319  rr PREND_LA_VALEUR 1
320  FIN_SINON
321  FIN_SI
322  SI (rr==0) ALORS
323  DEBUT_SI
324  r PREND_LA_VALEUR 0
325  FIN_SI
326  SI (rr==1) ALORS
327  DEBUT_SI
328  r PREND_LA_VALEUR 1
329  FIN_SI
330  SI (rr==2) ALORS
331  DEBUT_SI
332  r PREND_LA_VALEUR 2

```

```

333     FIN_SI
334     FIN_POUR
335     TT PREND_LA_VALEUR A+B+C
336     fA PREND_LA_VALEUR A/TT
337     fB PREND_LA_VALEUR B/TT
338     fC PREND_LA_VALEUR C/TT
339     mA PREND_LA_VALEUR abs((pA-fA)/pA)
340     mB PREND_LA_VALEUR abs((pB-fB)/pB)
341     mC PREND_LA_VALEUR abs((pC-fC)/pC)
342     M PREND_LA_VALEUR (mA+mB+mC)*100/3
343     S PREND_LA_VALEUR S+M
344     FIN_POUR
345     POUR i ALLANT_DE 0 A 99
346     DEBUT_POUR
347     SI (M>=i ET M<(i+1)) ALORS
348     DEBUT_SI
349     u[i] PREND_LA_VALEUR u[i]+1
350     FIN_SI
351     FIN_POUR
352     FIN_POUR
353     POUR i ALLANT_DE 0 A 99
354     DEBUT_POUR
355     SI (u[i]!=0) ALORS
356     DEBUT_SI
357     TRACER_POINT (i,u[i]/q)
358     FIN_SI
359     FIN_POUR
360     FIN_SI
361 FIN_ALGORITHME

```

## 5/ Cryptage

```

1  VARIABLES
2  Réaliser EST_DU_TYPE CHAINE
3  pAB EST_DU_TYPE NOMBRE
4  pAC EST_DU_TYPE NOMBRE
5  pBA EST_DU_TYPE NOMBRE
6  pBC EST_DU_TYPE NOMBRE
7  pCA EST_DU_TYPE NOMBRE
8  pCB EST_DU_TYPE NOMBRE
9  k EST_DU_TYPE NOMBRE
10 DEBUT_ALGORITHME
11 AFFICHER "Réaliser : C (cryptage) ou D (décryptage)"
12 LIRE Réaliser
13 AFFICHER "Clef"
14 LIRE k
15 AFFICHER " "
16 SI (Réaliser=="C") ALORS
17     DEBUT_SI
18     AFFICHER "Cryptage : "
19     LIRE pAB
20     LIRE pAC
21     LIRE pBA
22     LIRE pBC
23     LIRE pCA
24     LIRE pCB
25     pAB PREND_LA_VALEUR (pAB*k)/(k+1)
26     pBA PREND_LA_VALEUR (pBA*k)/(k+1)
27     pCA PREND_LA_VALEUR (pCA*k)/(k+1)
28     pAC PREND_LA_VALEUR 1-pAB
29     pBC PREND_LA_VALEUR 1-pBA
30     pCB PREND_LA_VALEUR 1-pCA
31     AFFICHER "pAB = "
32     AFFICHER pAB
33     AFFICHER "pAC = "
34     AFFICHER pAC
35     AFFICHER "pBA = "
36     AFFICHER pBA
37     AFFICHER "pBC = "
38     AFFICHER pBC
39     AFFICHER "pCA = "
40     AFFICHER pCA

```

```
41  AFFICHER "pCB = "
42  AFFICHER pCB
43  FIN_SI
44  SI (Réaliser=="D") ALORS
45  DEBUT_SI
46  AFFICHER "Décryptage : "
47  LIRE pAB
48  LIRE pAC
49  LIRE pBA
50  LIRE pBC
51  LIRE pCA
52  LIRE pCB
53  pAB PREND_LA_VALEUR (pAB*(k+1))/k
54  pBA PREND_LA_VALEUR (pBA*(k+1))/k
55  pCA PREND_LA_VALEUR (pCA*(k+1))/k
56  pAC PREND_LA_VALEUR 1-pAB
57  pBC PREND_LA_VALEUR 1-pBA
58  pCB PREND_LA_VALEUR 1-pCA
59  AFFICHER "pAB = "
60  AFFICHER pAB
61  AFFICHER "pAC = "
62  AFFICHER pAC
63  AFFICHER "pBA = "
64  AFFICHER pBA
65  AFFICHER "pBC = "
66  AFFICHER pBC
67  AFFICHER "pCA = "
68  AFFICHER pCA
69  AFFICHER "pCB = "
70  AFFICHER pCB
71  FIN_SI
72  FIN_ALGORITHME
```