

# QUALITE DE FABRICATION

Auteur : Marie-Laurence Brivezac

TI-83 Premium CE &amp; 82 Advanced

**Mots-clés :** Loi binomiale.**Fichiers associés :** QUALITE.8xp

## 1. Objectifs

Simuler la loi binomiale à l'aide de l'algorithmique et des listes sur TI-8x.

Ce travail peut-être réalisé indifféremment sur le logiciel TI Smartview CE ou la calculatrice. On considère ici les manipulations de base connues.

## 2. Énoncé

A la sortie de la chaîne de fabrication d'une usine, un test qualité sur des articles publicitaires a déterminé que 3 % des pièces fabriquées sont défectueuses. Ces articles sont livrés par lots de 4.

Proposer une simulation de la constitution des lots qui donne le nombre d'articles défectueux dans un lot.

En déduire la probabilité d'avoir au plus un article défectueux par lot.

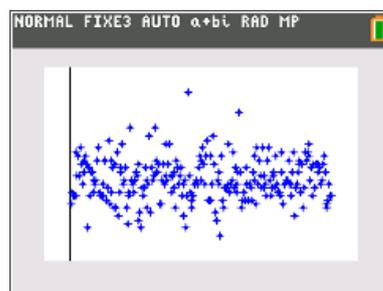
## 3. Commentaires

La constitution d'un lot suit un schéma de Bernoulli. Ce travail peut se simuler sur tableur ou avec un algorithme. Le choix est fait ici d'utiliser l'algorithmique conjointement à la gestion des données dans des listes.

## 4. Conduite de l'activité

### Simulation du choix d'un article

D'après le texte l'article est défectueux avec une probabilité de  $\frac{3}{100}$ , la fonction **NbreAléat** donne un nombre aléatoire uniformément réparti sur  $[0 ; 1[$  (second écran).



*Illustration de la génération de 250 nombres aléatoires avec un nuage de points.*

Ainsi, le choix d'un article selon l'instruction **NbreAléat <3/100** renvoie la valeur 1 ou 0 selon que l'article est défectueux ou non. Cette expression booléenne sera donc la base de la simulation.

## Simulation d'un lot

Quatre instructions **NbreAléat <3/100** successives et indépendantes constituent un lot.

## Simulation de 250 lots

*Remarque* : il est possible de réduire le nombre de lots pour la mise au point, afin de diminuer le temps d'exécution.

### L'algorithme

```
Initialiser les listes du programme à 0.
Pour I allant de 1 à 250 Faire
  Si nbrealéatoire < 3/100 alors L1(I):=1
  Si nbrealéatoire < 3/100 alors L2(I):=1
  Si nbrealéatoire < 3/100 alors L3(I):=1
  Si nbrealéatoire < 3/100 alors L4(I):=1
Fin Pour
Pour I allant de 1 à 250 Faire
  Affecter L5(I) de la
  Valeur L1(I)+L2(I)+L3(I)+L4(I)
Fin Pour
Affecter la valeur 0 à N
Pour I allant de 1 à 250 Faire
  Si L5(I) < 2 alors
    Augmenter N de 1
  Fin Si
Fin Pour
Afficher N/250
```

### Le programme en langage TI-8x

```
EffToutListes
For(I,1,250)
NbrAléat<(3/100)→L1(I)
NbrAléat<(3/100)→L2(I)
NbrAléat<(3/100)→L3(I)
NbrAléat<(3/100)→L4(I)
End
L1+L2+L3+L4→L5
0→N
For(I,1,250)
If L5(I)<2
Then
N+1→N
End
End
Disp "PROBABILITE AU", "PLUS UN
DEFAULT", N/250
```

### Remarques :

- (1) La somme des listes  $L1+L2+L3+L4$  ajoute les termes de même rang, c'est-à-dire qu'on affecte
  - à  $L5(1)$  la somme  $L1(1) + L2(1) + L3(1) + L4(1)$ ,
  - à  $L5(2)$  la somme  $L1(2) + L2(2) + L3(2) + L4(2)$ , etc.. Ainsi la boucle Pour est réalisée.
- (2) L'instruction EffToutListes est disponible dans le catalogue 2nde [0], taper trig (Premium CE) pour accéder aux instructions commençant par E.

### Exécution :

A l'exécution observer les listes obtenues à l'aide du menu STAT.  
Relancer le programme pour observer la fluctuation d'échantillonnage.

