

# **Compte rendu Math**

## **Générateur congruentiel linéaire et test du chi2**

# 1- Principe du Générateur congruentiel linéaire

```
def suite(liste,nbE,a,c,m,Xn):#on construit la suite
    i=0
    while(i<=nbE):#tant que i n'a pas atteint le nombre d'elements demandés...
        Xn=(a*Xn+c)%m#... chaque élément est saisi
        liste.append(Xn)#on l'ajoute
        i=i+1

def trouverPeriode(liste):#pour trouver une periode, on va prendre les deux premieres valeurs et parcourir la liste en essayant de les retrouver les mêmes configurations
    valeurInit=liste[0]#on prend la premiere valeur de la liste
    valeurInit2=liste[1]#on prend la seconde valeur de la liste
    for index in range(1,len(liste)):#on parcourt la liste à partir du second élément
        if liste[index]==valeurInit:#si on trouve une valeur identique à la premiere valeur...
            if liste[index+1]==valeurInit2:#... alors regardons si celle qui suit est la même que la seconde ...
                print(index)#... alors c'est qu'il existe une période, on prend l'index actuel pour connaître sa période
                exit()#on sort de cette boucle

a=25#on initialise les paramètres de base.
c=16
m=256

X0=125
liste=[]
liste.append(X0)
suite(liste,1000,a,c,m,X0);
print(liste)
trouverPeriode(liste)
```

Graine X0	Période
10	1
50	4
96	16
13	32
125	32

## À partir de ce travail, comment définiriez vous la notion de « bon » générateur pseudoaléatoire ?

C'est une formule capable de générer un grand nombre de valeurs aléatoires avec la plus grande période, c'est à dire de générer des valeurs aléatoires sans qu'il existe une suite de valeurs qui se répète trop rapidement.

## Est ce suffisant ??

Non, ça ne l'est pas il existe d'autres paramètres à vérifier comme par exemple les différents tests du chi2.

Ici, la période est clairement insuffisante. La période est beaucoup trop petite pour un grand nombre de valeur pour qu'il soit caractérisé de « bon » générateur pseudoaléatoire .

## 2- Générateur congruentiel linéaire : algorithme RANDU

```
def suite(Liste,nbE,a,c,m,Xn):  
    i=0  
    while(i<=nbE):  
        Xn=(a*Xn+c)%m  
        liste.append(Xn)  
        i=i+1  
  
def trouverPeriode(Liste):  
    valeurInit=liste[0]  
    valeurInit2=liste[1]  
    for index in range(1,len(liste)):  
        if liste[index]==valeurInit:  
            if liste[index+1]==valeurInit2:  
                print(index)  
                exit()  
  
a=65539  
c=0  
m=math.pow(2,31)  
  
X0=10  
liste=[]  
liste.append(X0)  
suite(liste,1000,a,c,m,X0);  
print(liste)  
trouverPeriode(liste)  
  
#On ne trouve aucune periode
```

### Trouvez vous une période ?

On ne trouve pas de période.

### 3- test statistique du chi2 :

```
def suite(liste,nbE,a,c,m,Xn):
    i=0
    while(i<=nbE):
        Xn=(a*Xn+c)%m
        liste.append(Xn)
        i=i+1

def DivisionListe(listeDep,listFin,m):#on va diviser chaque valeur par 256
    for index in range(0,len(listeDep)):#pour chaque valeur ...
        listeFin[index]=listeDep[index]/m
        #.... on divise la valeur par 256 de la liste de départ pour ensuite l'ajouter dans la seconde liste

def trouverPeriode(liste):
    valeurInit=liste[0]
    valeurInit2=liste[1]
    for index in range(1,len(liste)):
        if liste[index]==valeurInit:
            if liste[index+1]==valeurInit2:
                print(index)
                exit()

a=25
c=16
m=256

X0=50
liste=[]
liste.append(X0)
suite(liste,10000,a,c,m,X0)
print(liste)
trouverPeriode(liste)
liste2=DivisionListe(liste,m)
#Pour la suite, on crée une fonction rangerListe() avec la liste à trier en paramètre ainsi qu'une liste à deux dimensions
#On regarde chaque valeur puis on l'attribue à la liste[i][] correspondante.
#Par conjecture : le résultat sera le même que dans l'exo 1 étant donné que ce sont les mêmes paramètres de départ.
#b)Les résultats sont identiques
```

#### Le résultat était-il prévisible ?

Par conjecture : le résultat sera le même que dans l'exo 1 avec  $X_0=50$  étant donnée que ce sont les mêmes paramètres de départ. Il y aura donc une période de 4 soit 4 valeurs qui se répètent.

#### b- Refaites le test du chi2 dans le cas des différentes valeurs de graines du tableau 1

Les résultats sont identiques.

#### c- testez maintenant le cas de RANDU et concluez sur sa qualité.

Pour RANDU, on peut conjecturer que le test statistique va se passer normalement sans avoir de période. RANDU passera donc le test du chi2 mais est-ce suffisant ? Clairement pas.