

## **Dossier de tests**

V 02.00

## 0 - Table des matières :

<b>0 - Table des matières :</b>	<b>2</b>
<b>• A/ Test n°1 ; Nombres premiers</b>	<b>3</b>
<b>1 - Le contexte</b>	<b>3</b>
1.1 Introduction	3
1.3 Description des informations à enregistrer pour les tests	3
<b>2 - Les tests</b>	<b>3</b>
2.1 Campagne de test	3
2.2 - Résultats attendus	4
2.3 - Résultats des tests	4
<b>3 - Conclusion :</b>	<b>5</b>
<b>• B/ Test n°2 ; Probabilité de loi normale</b>	<b>6</b>
<b>1 - Le contexte</b>	<b>6</b>
1.1 Introduction	6
1.3 Description des informations à enregistrer pour les tests	6
<b>2 - Les tests</b>	<b>7</b>
2.1 Campagne de test	7
2.2 - Résultats attendus	7
<b>2.3 - Résultats des tests</b>	<b>8</b>
<b>3 - Conclusion :</b>	<b>10</b>

• A/ Test n°1 : Nombres premiers

Responsable de la rédaction : Belaidi Elyas

## 1 - Le contexte

### 1.1 Introduction

Ce document suit la trace des tests effectués sur le site de calcul due au projet SAE, et présente l'environnement dans lequel ils ont été effectués. Le but de ces tests est de déterminer si le module de calcul remplit les conditions nécessaires pour donner la liste des nombres premiers de 1 à N.

### 1.2 Description de la procédure de test

La procédure effectuée répond au système de test unitaire. Ces tests seront segmentés le plus possible afin de déterminer rapidement dans quelle parcelle de code se trouve une erreur. Durant ces tests, il sera effectué une vérification des résultats de calculs simples.

Afin d'obtenir des résultats concrets, sur chaque méthode de calcul, il sera effectué des tests, chacun comprenant une seule variable. Cette variable variera afin d'avoir des résultats fiables.

### 1.3 Description des informations à enregistrer pour les tests

Le résultat pour chaque variable sera calculé avec un calculateur en ligne, puis comparé avec les résultats obtenus avec le module de calcul. Il faut que les résultats obtenus avec le module correspondent avec les valeurs obtenues au préalable pour que le test soit concluant.

## 2 - Les tests

### 2.1 Campagne de test

Produit testé : Module de calcul de nombres premiers Date de début : 21/11/2023

Date de finalisation : 22/11/2023

Tests à appliquer : Calcul de l'effectif des nombres premiers de 1 à n Responsable de la campagne de test : Belaidi Elyas

## 2.2 - Résultats attendus

<b>Outils utilisé :</b> <a href="https://www.dcode.fr/prime">https://www.dcode.fr/prime</a> <b>Version : 1.0</b>		
Classe	N	Résultat attendu (R)
P1	N = -1	R = INVALID
P2	N = 0	R = INVALID
P3	N = 1	R = []
P4	N = 2	R = [2]
P5	N = 35	R = [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31]
P6	N = 472	R = [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163, 167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239, 241, 251, 257, 263, 269, 271, 277, 281, 283, 293, 307, 311, 313, 317, 331, 337, 347, 349, 353, 359, 367, 373, 379, 383, 389, 397, 401, 409, 419, 421, 431, 433, 439, 443, 449, 457, 461, 463, 467]

### 2.3 - Résultats des tests

<b>Identification du test</b> : Calcul de l'effectif des nombres premiers compris entre 1 et n		
<b>Version : 1.0</b>		
<b>Classe</b>	<b>N</b>	<b>Résultat obtenu (R)</b>
P1	N = -1	R = None
P2	N = 0	R = None
P3	N = 1	R = []
P4	N = 2	R = [2]
P5	N = 35	R = [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31]
P6	N = 472	R = [2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163, 167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239, 241, 251, 257, 263, 269, 271, 277, 281, 283, 293, 307, 311, 313, 317, 331, 337, 347, 349, 353, 359, 367, 373, 379, 383, 389, 397, 401, 409, 419, 421, 431, 433, 439, 443, 449, 457, 461, 463, 467]

### 3 - Conclusion :

Les résultats obtenus par les tests de notre module de calcul de l'effectif des nombres premiers compris entre 1 et n sont similaires aux résultats obtenus par un simulateur en ligne fiable. On peut donc en conclure que les tests sont concluants, et que notre module de calcul est fiable.

## • B/ Test n°2 : Probabilité de loi normale

Responsable de la rédaction : Bullock Patrick

### 1 - Le contexte

#### 1.1 Introduction

Ce document suit la trace des tests effectués sur le site de calcul due au projet SAE, et présente l'environnement dans lequel ils ont été effectués. Le but de ces tests est de déterminer si le module de calcul utilisant la méthode de Monte-Carlo pour calculer des probabilité de loi normale.

Le calcul de probabilité de loi normale est une méthode statistique qui évalue la probabilité qu'une variable aléatoire continue. Il repose sur la distribution normale, également appelée courbe en cloche, caractérisée par sa moyenne et son écart-type.

#### 1.2 Description de la procédure de test

La procédure effectuée répond au système de test unitaire. Ces tests seront segmentés le plus possible afin de déterminer rapidement dans quelle parcelle de code se trouve une erreur. Durant ces tests, il sera effectué une vérification des résultats de calculs simples.

Afin d'obtenir des résultats concrets, sur chaque méthode de calcul, il sera effectué des tests.

Les tests comprennent entre 3 et 4 variables :

- Moy : La moyenne
- $\sigma$  : L'écart-type
- T1 : Défini le seuil du calcul  $P(X > T1)$
- T2 : Défini le seuil du calcul  $P(X < T2)$

Ces variables varieront afin d'avoir des résultats fiables.

#### 1.3 Description des informations à enregistrer pour les tests

Le résultat pour chaque variable sera calculé avec un calculateur en ligne, puis comparé avec les résultats obtenus avec le module de calcul. Il faut que les résultats obtenus avec le module correspondent avec les valeurs obtenues au préalable pour que le test soit concluant.

Il est aussi important de noter que le module de calcul développé utilise la fonction random de python et donc que les résultats obtenus pour un même calcul ne seront jamais les mêmes, cependant ils varieront très peu.

## 2 - Les tests

### 2.1 Campagne de test

Produit testé : Module de calcul des probabilité de loi normale

Date de début : 24/11/2023

Date de finalisation : 4/12/2023

Responsable de la campagne de test : Bullock Patrick

### 2.2 - Résultats attendus

**Outils utilisé :** <https://www.123calculus.com/loi-normale-probabilites-page-1-50-170.html>

**Version : 2.0**

Classe	Moy	$\sigma$	T1	Résultat attendu ( $R = P(X > T1)$ )
P1	Moy = 0	$\sigma = 1$	T1 = 0	R = 0.5
P2	Moy = 2	$\sigma = 5$	T1 = 0.5	R = 0.6179
P3	Moy = -1	$\sigma = 10$	T1 = -2	R = 0.5398
P4	Moy = 3	$\sigma = 15$	T1 = 3.5	R = 0.4867

Classe	Moy	$\sigma$	T2	Résultat attendu ( $R = P(X < T2)$ )
P5	Moy = 0	$\sigma = 1$	T2 = 1.5	R = 0.9332
P6	Moy = 2	$\sigma = 5$	T2 = 3	R = 0.5793
P7	Moy = -1	$\sigma = 10$	T2 = 0	R = 0.5398
P8	Moy = 3	$\sigma = 15$	T2 = 2.5	R = 0.4867

Classe	Moy	$\sigma$	T1	T2	Résultat attendu ( $R = P(T1 < X < T2)$ )
P9	Moy = 0	$\sigma = 1$	T1 = 1	T2 = 2	R = 0.1359
P10	Moy = 2	$\sigma = 5$	T1 = 0.5	T2 = 3.5	R = 0.2358
P11	Moy = -1	$\sigma = 10$	T1 = -2	T2 = 1	R = 0.1191
P12	Moy = 3	$\sigma = 15$	T1 = Null	T2 = Null	Erreur
P13	Moy = 0	$\sigma = 1$	T1 = 5	T2 = 0	Erreur

### 2.3 - Résultats des tests

**Identification du test :** Calcul utilisant la méthode de Monte-Carlo pour calculer des probabilité de loi normale

**Version : 2.0**

Classe	Moy	$\sigma$	T1	Résultat obtenu ( $R = P(X > T1)$ )
P1	Moy = 0	$\sigma = 1$	T1 = 0	R = 0.4998
P2	Moy = 2	$\sigma = 5$	T1 = 0.5	R = 0.6175
P3	Moy = -1	$\sigma = 10$	T1 = -2	R = 0.5392
P4	Moy = 3	$\sigma = 15$	T1 = 3.5	R = 0.4885

Classe	Moy	$\sigma$	T2	Résultat obtenu ( $R = P(X < T2)$ )
P5	Moy = 0	$\sigma = 1$	T2 = 1.5	R = 0.9336
P6	Moy = 2	$\sigma = 5$	T2 = 3	R = 0.5790
P7	Moy = -1	$\sigma = 10$	T2 = 0	R = 0.5391
P8	Moy = 3	$\sigma = 15$	T2 = 2.5	R = 0.4859

Classe	Moy	$\sigma$	T1	T2	Résultat obtenu ( $R = P(T1 < X < T2)$ )
P9	Moy = 0	$\sigma = 1$	T1 = 1	T2 = 2	R = 0.1362
P10	Moy = 2	$\sigma = 5$	T1 = 0.5	T2 = 3.5	R = 0.2356
P11	Moy = -1	$\sigma = 10$	T1 = -2	T2 = 1	R = 0.1185
P12	Moy = 3	$\sigma = 15$	T1 = None	T2 = None	“Erreur: Fournissez au moins un des seuils t1 ou t2.”
P13	Moy = 0	$\sigma = 1$	T1 = 5	T2 = 0	“Erreur: t1 doit être strictement inférieur à t2.”



### 3 - Conclusion :

Les résultats obtenus par les tests de notre module de calcul utilisant la méthode de Monte-Carlo pour calculer des probabilité de loi normale sont quasi-similaires aux résultats obtenus par un simulateur en ligne fiable.

Les sont précis à l'échelle  $10^{-3}$  si l'écart type est petit, par contre, plus l'écart-type augmente plus l'imprécision augmente. Cela est dû à la méthode de Monte-Carlo qui place des points aléatoirement dans une zone définie pour calculer son aire, plus la zone est grande et l'écart type en est le facteur majeur, plus le résultat sera imprécis.

On peut pallier ce problème en augmentant le nombre de points, qui ici était de 3.000.000. Avec tout cela, on peut donc en conclure que les tests sont concluants, et que notre module de calcul est fiable.