

Découverte propriété exponentielle Correction

May 16, 2021

1 Découvrir des propriétés de la fonction exponentielle notée exp

On étudie l'évolution d'une colonie de bactéries dans un milieu renouvelé. Le nombre de bactéries, en centaine, en fonction du temps t , en h , est modélisé par la fonction N définie sur l'intervalle $[0; +\infty[$ par : $N(t) = \exp(t)$ (fonction exponentielle). 1°) On vous fournit le programme ci-dessous. * Exécuter-le. * À quoi sert-il ? Il permet d'obtenir la liste des 8 images de la fonction $\exp(x)$ pour les 8 premiers entiers naturels de 0 à 7.

```
[1]: from math import *

def exponentielle():
    liste=[] # Création d'une liste vide
    for i in range(8):
        liste.append(round(exp(i),3)) #round(f,3) arrondi à 0,001 près
    return (liste)

print(exponentielle())
```

[1.0, 2.718, 7.389, 20.086, 54.598, 148.413, 403.429, 1096.633]

2°) À l'aide des résultats précédents, faire les calculs suivants: * $N(2) \times N(3)$ On a 148,415...proche de $N(5)$ * $N(3) \times N(4)$ On a 1096,655...proche de $N(7)$ * $N(1) \times N(5)$ On a 403,386... proche de $N(6)$ * Que remarquez-vous en comparant les résultats au tableau? * Conjecturer une formule donnant le produit $\exp(x) \times \exp(y)$ où x et y sont des nombres réels. On peut donc penser que $\exp(x) \times \exp(y) = \exp(x + y)$

3°) On souhaite tester la propriété conjecturée pour les valeurs entières t appartenant à l'intervalle $[0; 7]$. On établit alors le programme. * Que font les lignes 7,8 et 9 ? Les lignes 7 et 8 permettent d'affecter à la variable i et j un nombre aléatoire entier compris entre 0 et 7 inclus. La ligne 9 sert à affecter à la variable résultat le résultat du calcul mentionné. * À quoi sert la ligne 10 ? On effectue un test pour savoir si il y a égalité entre les valeurs stockées dans la variable résultat et celle provenant du calcul mentionné. * Quel est le rôle de ce programme ? Il sert à valider 8 fois la conjecture établit précédemment.

```
[2]: from math import *
from random import randint

def produit():
    compteur=0
```

```

for k in range(8):
    i=randint(0,8)
    j=randint(0,8)
    resultat=exp(i)*exp(j)
    if resultat==exp(i+j):
        compteur+=1
return (compteur)

print (produit())

```

6

- Pourquoi celui-ci ne nous rend pas 8 à chaque fois qu'on l'exécute? Par ce que nous manipulons ce que l'on appelle des flottants en informatique qui ne correspondent pas exactement aux nombres réels. Aussi, les arrondis ne sont pas forcément similaires...

```

[3]: from math import *
from random import randint

def produit(n):
    epsilon=0.001
    compteur=0
    for k in range(n):
        i=randint(0,8)
        j=randint(0,8)
        resultat=exp(i)*exp(j)
        if resultat-exp(i+j)<epsilon:
            compteur+=1
    return (compteur)

print (produit(1000))

```

1000

- Pourquoi cette modification ligne 11 est-elle fondamentale en informatique ? On règle le problème en acceptant qu'il puisse y avoir un écart très petit mais petit quand même entre la valeur que l'ordinateur va établir et celle exacte.
- Que change le passage de la fonction produit() en fonction produit(n)? Au lieu de ne tester que 8 fois, on peut tester le nombre de fois que l'on souhaite. Regarder où se situe le "n". Modifier dans le programme à la ligne 15 le nombre 1000 par 350 puis exécuter.

4°) À l'aide des résultats de la question 1, faire les calculs suivants: $\frac{N(7)}{N(1)} * \frac{N(5)}{N(4)} * \frac{N(6)}{N(2)}$ * Que remarquez-vous en comparant les résultats au tableau? * Conjecturer une formule donnant le produit $\frac{\exp(x)}{\exp(y)}$ où x et y sont des nombres réels. On trouve la formule $\frac{\exp(x)}{\exp(y)} = \exp(x - y)$

5°) Créer un programme permettant de tester la deuxième propriété pour les valeurs entières t appartenant à l'intervalle $[0; 7]$ une centaine de fois.

```
[4]: from math import *
      from random import randint

      def quotient(n):
          epsilon=0.001
          compteur=0
          for k in range(n):
              i=randint(0,8)
              j=randint(0,8)
              resultat=exp(i)/exp(j)
              if resultat-exp(i-j)<epsilon:
                  compteur+=1
          return (compteur)

      print (quotient(1000))
```

1000

```
[ ]:
```