# Thème : Les nombres premiers

## Corrigé de l’activité 2. Algorithmes sur les nombres premiers (3 exercices)

**Exercice 1 : *Tester la primalité (exercice identique à l’exercice 3 de l’activité 1 « Autour des nombres premiers »)***

1. 107 n'est pas divisible par 2, 3, 5, …, 103 : c'est donc un **nombre premier**.

* 227 n'est pas divisible par 2, 3, 5, …, 223 : c'est donc **un nombre premier**.
* 375 est divisible par 5, ce n'est donc pas un nombre premier.
* 377 est divisible par 13, ce n'est donc pas un nombre premier.
* 379 n’est pas divisible par 2, 3, 5, …, 373 : c'est donc un **nombre premier**.
* 571 n'est pas divisible par 2, 3, 5, …, 569 : c'est donc un **nombre premier**.

1. Pour 107, il n'est pas utile de tester tous les nombres premiers de 2 jusqu'à 103. En effet, il est certes nécessaire de tester la division par 2, 3, …

Mais par exemple, le test de la division par 73 est inutile : PartEnt (107/73) = 1.

Les élèves auront tendance à énoncer le fait qu'il faut "s'arrêter" à 107/2.

Il faudra relancer le débat en demandant si l'on ne peut pas faire mieux…

Et ce jusqu'à obtenir que si est composé alors son plus petit diviseur noté est inférieur ou égal à .

Preuve : On suppose que a au moins un autre diviseur que 1 et lui-même. On appelle *le plus petit* de ces diviseurs.

Donc où est un diviseur de . Alors :

* Il existe un second diviseur tel que
* Ce second diviseur est tel que . Donc

. Donc il faut et il suffit de tester la division par tous les nombres premiers inférieurs ou égaux à 7.

1. On fait fonctionner l'algorithme sur le papier :

Pour ,

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  | 19 |  |
| 1er tour de boucle Tant que |  | 2 |
| 2e tour de boucle Tant que |  | 3 |
| 3e tour de boucle Tant que |  | 4 |
| 4e tour de boucle Tant que |  | 5 |
| Sortie de boucle car la condition ne divise pas est vraie et la condition est fausse |  |  |

L'affichage est PREMIER.

|  |  |  |  |
| --- | --- | --- | --- |
| Pour , |  |  |  |
|  |  | 21 |  |
|  | 1er tour de boucle Tant que |  | 2 |
|  | 2e tour de boucle Tant que |  | 3 |
|  | Sortie de boucle car la condition ne divise pas est fausse et la condition est vraie |  |  |

L'affichage est NON PREMIER.

|  |  |  |  |
| --- | --- | --- | --- |
| Pour , |  |  |  |
|  |  | 53 |  |
|  | 1er tour de boucle Tant que |  | 2 |
|  | 2e tour de boucle Tant que |  | 3 |
|  | 3e tour de boucle Tant que |  | 4 |
|  | 4e tour de boucle Tant que |  | 5 |
|  | 5e tour de boucle Tant que |  | 6 |
|  | 6e tour de boucle Tant que |  | 7 |
|  | 7e tour de boucle Tant que |  | 8 |
|  | Sortie de boucle car la condition ne divise pas est vraie et la condition est fausse |  |  |

L'affichage est PREMIER.

1. a) Programme TESTA

Programme TESTA sur TI

Prompt N

If N ≤ 3

Then

If N=2 ou N=3

Then

Disp "N PREMIER"

Else

Disp "N NON PREMIER"

End

Else

2I

While (partEnt(N/I)≠N/I) et (I≤ partEnt(√(N)))

I+1I

End

If partEnt(N/I)=N/I

Then

Disp "N NON PREMIER"

Else

Disp "N PREMIER"

End

End

Programme TESTA sur Casio

"N="?→N

If N≤ 3

Then

If N=2 Or N=3

Then

"N PREMIER"◢

Else

"N NON PREMIER"◢

IfEnd

Else

2→I

While (Intg(N÷I)≠N÷I) And (I≤ Intg(√(N)))

I+1I

WhileEnd

If Intg(N÷I)=N÷I

Then

"N NON PREMIER"◢

Else

"N PREMIER"◢

IfEnd

IfEnd

ClrText

b) En utilisant ce programme, on affiche :

* 19 et 53 sont premiers et 21 n'est pas premier.
* 2011 et 2017 sont premiers et 2013, 2015 et 2019 ne pas premiers.

|  |  |  |  |
| --- | --- | --- | --- |
| c) | TI83 | TI83 CE Premium | Casio Graph 35+ USB |
| TESTA : Durée du test de la primalité de 7 237 031 | 75 s | 58 s | 31 s |

1. a) Modification de l'algorithme pour tester la division par et pour ensuite ne pas tester la division par les autres nombres pairs :

Déclaration des variables : *N* et *I* sont des entiers

Début de l'algorithme

Saisir *N*

Si

Alors

Si

Alors

Afficher "*N* PREMIER"

Sinon

Afficher "*N* NON PREMIER"

Fin Si

Sinon

Si divise

Alors

Afficher " NON PREMIER"

Sinon

prend pour valeur

Tant que (*I* ne divise pas *N*) et (*I* E)

*I* prend la valeur *I* + 2

Fin Tant que

Si *I* divise *N*

Alors

Afficher "*N* NON PREMIER"

Sinon

Afficher "*N* PREMIER"

Fin Si

Fin Si

Fin de l'algorithme

* 1. Programmation sur calculatrice :

Programme TESTB sur TI

Prompt N

If N ≤ 3

Then

If N=2 ou N=3

Then

Disp "N PREMIER"

Else

Disp "N NON PREMIER"

End

Else

If partEnt(N/2)=N/2

Then

Disp "N NON PREMIER"

Else

3I

While (partEnt(N/I)≠N/I) et (I≤ partEnt(√(N)))

I+2I

End

If partEnt(N/I)=N/I

Then

Disp "N NON PREMIER"

Else

Disp "N PREMIER"

End

End

Programme TESTB sur Casio

"N="?→N

If N≤ 3

Then

If N=2 Or N=3

Then

"N PREMIER"◢

Else

"N NON PREMIER"◢

IfEnd

Else

If Intg(N÷2)=N÷2

Then

"N NON PREMIER"◢

Else

3→I

While (Intg(N÷I)≠N÷I) And (I≤ Intg(√(N)))

I+2I

WhileEnd

If Intg(N÷I)=N÷I

Then

"N NON PREMIER"◢

Else

"N PREMIER"◢

IfEnd

IfEnd

ClrText

* 1. L'amélioration du temps d'exécution du programme est très significative. La durée d'exécution du programme est divisée par 2. C'est une très nette amélioration pour de grandes valeurs de *N*.

|  |  |  |  |
| --- | --- | --- | --- |
|  | TI83 | TI83 CE Premium | Casio Graph 35+ USB |
| TESTB : Durée du test de la primalité de 7 237 031 | 38 s | 29 s | 16 s |

**Exercice 2 : *Liste des nombres premiers inférieurs à 1000***

1. L'exécution du programme LISTPREM dure environ :.

|  |  |  |  |
| --- | --- | --- | --- |
|  | TI83 | TI83 CE Premium | Casio Graph 35+ USB |
| LISTPREM : Durée de la construction de la liste LP de tous les nombres premiers jusqu'à 1009 | 120 s | 140 s | 54s |

1. Dans la liste Liste6, on peut lire 2, 3, 5 … 997, 1 009. On a donc bien obtenu les nombres premiers inférieurs ou égaux à .
2. En utilisant la liste Liste6 des nombres premiers inférieurs ou égaux à , on obtient :

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | 1 000 |
|  | 25 | 46 | 62 | 78 | 95 | 109 | 125 | 139 | 154 | 168 |

On ne peut donc pas parler d'équi-répartition des nombres premiers.

En effet, si c'était le cas, chaque centaine comporterait le même nombre de nombres premiers. Or :

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | 1 000 |
|  | 25 | 46 | 62 | 78 | 95 | 109 | 125 | 139 | 154 | 168 |

+21 +16 +16 +17 +14 +16 +14 +15 +14

Il semble que la densité des nombres premiers diminue quand augmente.

**Exercice 3 : *Recherche des diviseurs premiers inférieurs ou égaux à 1000***

1. Voici les programmes pour calculatrices TI et Casio de cet algorithme :

Programme DIVISPRE sur Casio

"N="?→N

1 →I

0 →P

While P<997 And P ≤N

List 6[I] → P

If Intg(N÷P)=N÷P

Then

"DIVISIBLE PAR":P◢

IfEnd

I+1 →I

WhileEnd

ClrText

Programme DIVISPRE sur TI

Prompt N

1 →I

0 →P

While P<997 et P ≤N

L6(I) → P

If partEnt(N/P)=N/P

Then

Disp "DIVISIBLE PAR",P

End

I+1 →I

End

1. L'exécution du programme DIVISPREM, permet d'obtenir :

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 12 | 13 | 985 | 997 | 1 000 | 1 009 |
| Affichage :  DIVISIBLE PAR | 2 ; 3 | 13 | 5 ; 197 | 997 | 2 ; 5 | aucun |

1. Le programme DIVISPREM donnera tous les diviseurs premiers des entiers *N* dont la décomposition en produit de facteurs premiers ne comporte aucun nombre premier strictement supérieur à 997. Par exemple, testons le nombre .

L'affichage est alors : DIVISIBLE PAR 2 ; 3 ; 5 ; 7 ; 11 ; 13 ; 17.

1. Le programme donnera certains diviseurs premiers mais pas tous pour tous les nombres qui admettent au moins un diviseur premier strictement supérieur à 997. Par exemple, testons le nombre . L'affichage est alors : DIVISIBLE PAR 2 ; 3 ; 5. Le programme ne donne donc pas 1 009 qui est un nombre premier.
2. Le programme ne donnera aucun diviseur premier pour tous les nombres dont la décomposition en produit de facteurs premiers ne comporte que des termes strictement supérieurs à 1 009. Par exemple, testons le nombre . Le programme n'affiche alors aucun diviseur premier alors que pourtant 1 013 et 1 019 sont deux nombres premiers.