# Thème : Les nombres premiers

## Activité 2. Algorithmes sur les nombres premiers (3 exercices)

**Exercice 1 : *Tester la primalité***

1. On considère l'algorithme suivant :

***Note :***

La fonction  **est la fonction partie entière :**

A tout réel elle associe l’unique entier relatif qui lui est immédiatement inférieur ou égal.

Exemples :

; ; .

* Sur les calculatrices TI, il s’agit de la commande **partEnt** - ou **int** pour les calculatrices en anglais - (touche math, menu MATH)
* Sur les Casio, il s’agit de la commande **Intg** (touche OPTN, menu NUM).

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

Début de l'algorithme

Saisir *N*

Pour J allant de 2 à E

Si *J* divise *N*

Alors

Afficher "*N* NON PREMIER"

Stopper le programme

Fin Si

Fin Pour

Afficher "*N* PREMIER"

Fin de l'algorithme

Faire fonctionner sur le papier cet algorithme pour et . Qu'affiche-t-il ?

1. a) Programmer cet algorithme sur la calculatrice (Nommer ce programme ESTPREM).
   1. Vérifier que le programme fonctionne pour et puis l'utiliser pour dire parmi les nombres suivants lesquels sont premiers : 2 011 ; 2 013 ; 2 015 ; 2 017 et 2 019.
   2. Chronométrer le temps que met le programme ESTPREM pour tester la primalité de

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

Pré requis : Avoir le programme ESTPREM de test de la primalité de l’exercice 1.

Objectif : Créer un algorithme qui crée la liste des nombres premiers inférieurs ou égaux à 1009. Créer le programme LISTPREM sur la calculatrice.

# Principe

* On utilise une nouvelle liste qui sera la Liste6 (cela laisse libres les Listes 1, 2, 3… souvent utilisées pour des calculs courants). On utilisera une version aménagée en sous-programme du programme ESTPREM pour tester si un entier naturel est premier.
* On commence avec . Si est premier

Alors on écrit dans la liste Liste6 (à la ligne )

Comme est premier on écrit donc à la première ligne de la liste ce qui donne Liste6[1] = 2

* On incrémente[[1]](#footnote-1) de . On a alors

Si est premier

Alors on écrit dans la liste Liste6 (à la ligne )

Comme est premier on écrit donc à la deuxième ligne de la liste ce qui donne Liste6[2] = 3

* On incrémente de . On a alors

Si est premier

Comme n’est pas premier on n’écrit donc rien dans la liste Liste6.

* On incrémente de . On a alors

Si est premier

Alors on écrit dans la liste Liste6 (à la ligne )

Comme est premier on écrit donc à la troisième ligne de la liste Liste6 qui donne Liste6[3] = 5

* On réitère cette démarche tant que

Après l’exécution du programme, on aura dans la liste Liste6 les premiers nombres premiers :

|  |
| --- |
| Liste6 |
| 2 |
| 3 |
| 5 |
| 7 |
| 11 |

# Modification du programme ESTPREM (renommé PREM)

On modifie le programme de test de primalité ESTPREM car on va en avoir besoin comme sous-programme dans le nouvel programme LISTPREM

|  |  |
| --- | --- |
| **Procédure sur calculatrice TI** | **Procédure sur calculatrice Casio** |
| * Créer un nouveau programme PREM * Y copier le programme ESTPREM. Pour cela :   Le curseur étant sur la première ligne (juste après **:** )  2nd sto→  Prgm EXEC ESTPREM  Entrée Entrée   * Enlever avec la touche suppr la première ligne Prompt N * Dans le programme PREM,   + Remplacer la ligne **Disp « NON PREMIER »** par **0 P** (touche annul pour effacer la ligne sans la supprimer)   + Remplacer **Stop** par **Return**   + Remplacer la ligne **Disp « PREMIER »** par **1 P**   + Ecrire sur la dernière ligne : **Return** (se trouve dans prgm CTL) | Renommer le programme ESTPREM en PREM   * Enlever avec la touche DEL la première ligne ‘‘N=’’?→N * Dans le programme PREM,   + Remplacer les trois lignes **‘‘N NON PREMIER’’**◢ par **0 P** (touche annul pour effacer la ligne sans la supprimer)   + Remplacer les deux lignes **Disp « N PREMIER »** par **1 P**   Ecrire sur la dernière ligne : **Return** (se trouve dans prgm CTL) |

Ainsi, le programme PREM reçoit un entier et renvoie la variable qui vaut lorsque est premier et lorsque est composé.

PREM sur TI

For(J,2, partEnt(√(N))

If N/J=partEnt(N/J)

Then

0P

Return

End

End

1P

Return

# Algorithme LISTPREM

Effacer la liste Liste6

reçoit

reçoit

**Tant que** **faire**

Appel du sous-programme PREM

**Si**

**Alors** Liste6[*I*] reçoit

reçoit

**FinSi**

reçoit

**FinTant que**

# Programmation de la calculatrice

On traduit l’algorithme précédent en langage TI ou Casio.

## Ecriture du programme LISTPREM

Saisir le programme « LISTPREM » sur la calculatrice.

|  |  |
| --- | --- |
| Calculatrice TI | Calculatrice Casio |
| * On choisit L6 dans 2nd listes NOMS * prgmPREM s’écrit en tapant prgm EXEC PREM entrer |  |

LISTPREM sur TI

EffListe L6

2N

1I

While N≤1009

prgmPREM

If P=1

Then

NL6(I)

I+1I

End

N+1N

End

LISTPREM sur Casio

ClrList 6

2N

1J

While N≤1009

Prog ”PREM”

If P=1

Then

NList 6[J]

J+1J

IfEnd

N+1N

WhileEnd

ClrText

## Utilisation du programme LISTPREM

1. Exécutez-le en le chronométrant.
2. Vérifiez que vous avez obtenu dans la liste Liste6 les nombres premiers inférieurs ou égaux à en allant dans le menu Statistiques
3. On note le nombre de nombres premiers inférieurs ou égaux à . Utilisez la liste Liste6 des nombres premiers inférieurs ou égaux à pour compléter le tableau ci-dessous :

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 100 | 200 | 300 | 400 | 500 | 600 | 700 | 800 | 900 | 1000 |
|  |  |  |  |  |  |  |  |  |  |  |

Peut-on parler d'équi-répartition des nombres premiers ? Justifier votre réponse.

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

Pré requis : Avoir la liste des nombres premiers inférieurs à 1000 de l’exercice 2.

Objectif : Obtenir un algorithme qui donne la liste des diviseurs premiers inférieurs à 1000 d’un entier naturel non nul quelconque.

# Principe

* On donne un entier naturel non nul .
* On cherche dans la liste des nombres premiers inférieurs ou égaux à ceux qui divisent .
* On commence par créer la liste dans la calculatrice des nombres premiers de à à l’aide du programme LISTPREM (voir l’exercice 2).

# Algorithme DIVISPRE

Saisir

reçoit

reçoit

Tant que et faire

reçoit l’élément *I* de la liste Liste6

Si *N* est divisible par *P*

Alors

Afficher « Divisible par », P

Fin Si

reçoit

FinTant que

# Programmation de la calculatrice

*Travail préalable : Exécutez le programme LISTPREM et assurez-vous que la liste Liste6 contient tous les nombres premiers de à .*

1. Ecrivez le programme « DIVISPRE » dans votre calculatrice.
2. Exécutez-le en cherchant les diviseurs premiers de  
    ;  ;  ; .
3. Quels sont les nombres dont le programme donnera tous les diviseurs premiers ? Testez sur un exemple.
4. Quels sont les nombres dont le programme donnera certains diviseurs premiers mais pas tous ? Testez sur un exemple.
5. Quels sont les nombres dont le programme ne donnera aucun diviseur premier ? Testez sur un exemple.

1. **Incrémenter** une variable, c’est lui ajouter une quantité constante [↑](#footnote-ref-1)