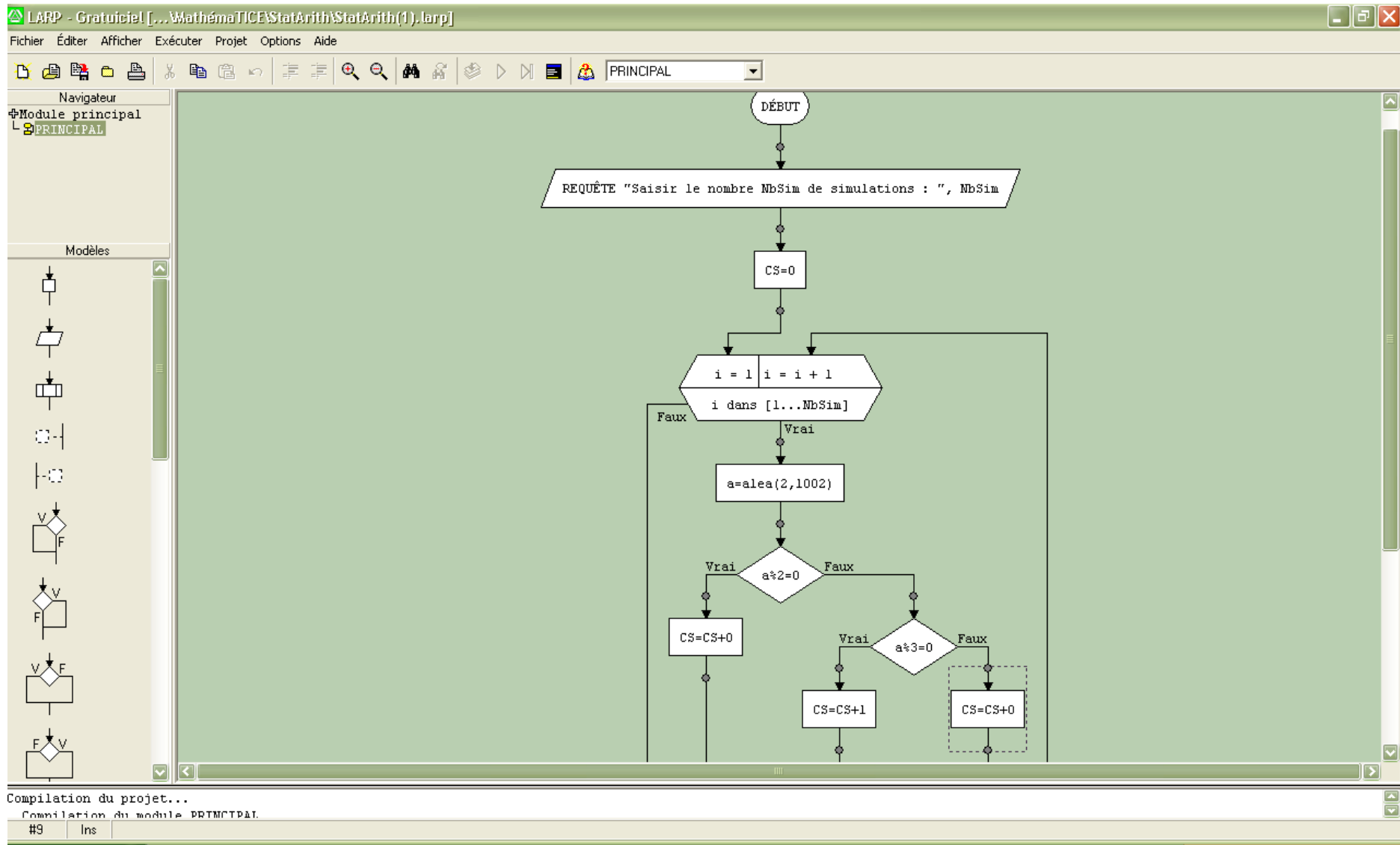


Statistique, probabilités et Arithmétique

1. Sur la probabilité qu'un nombre entier tiré au hasard entre 2 et 1002 ait 3 pour plus petit diviseur premier...



LARP - Gratuitiel [...\WathémaTICE\StatArith\StatArith(1).larp]

Fichier Éditer Afficher Exécuter Projet Options Aide

PRINCIPAL

Module principal
PRINCIPAL

Modèles

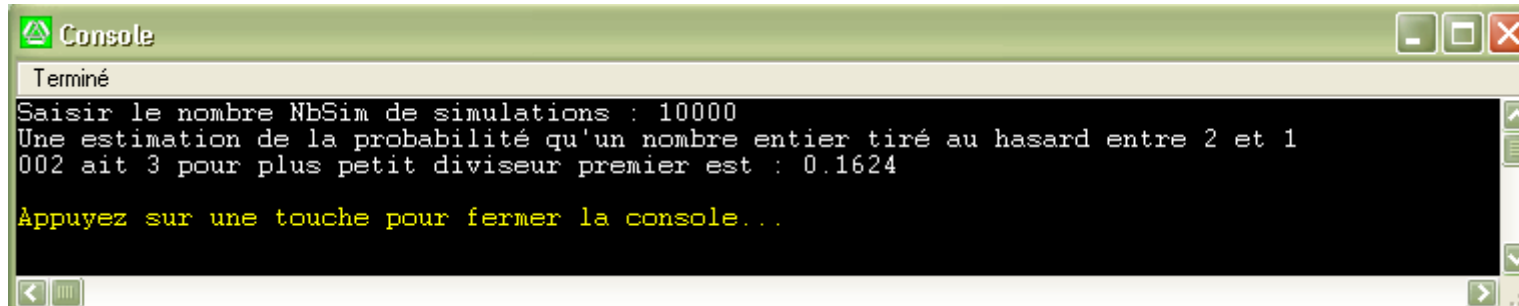
```

graph TD
    Start([i = 1  
i = i + 1  
i dans [1...NbSim]]) -- Vrai --> Alea[a=alea(2,1002)]
    Start -- Faux --> End([Fin])
    Alea --> Mod2{a%2=0}
    Mod2 -- Vrai --> CS0[CS=CS+0]
    Mod2 -- Faux --> Mod3{a%3=0}
    Mod3 -- Vrai --> CS1[CS=CS+1]
    Mod3 -- Faux --> CS0_2[CS=CS+0]
    CS0 --> End
    CS1 --> End
    CS0_2 --> End
    
```

ÉCRIRE "Une estimation de la probabilité qu'un nombre entier tiré au hasard entre 2 et 1002 ait 3 pour plus petit diviseur premier est :",CS/Nbsim

Compilation du projet...
Compilation du module PRINCIPAL.
#9 Ins

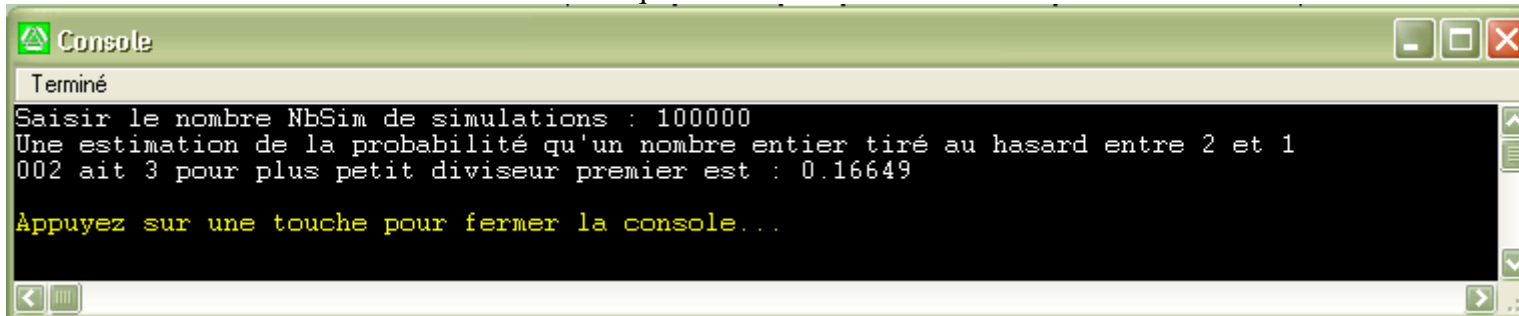
on obtient instantanément :



```
Console
Terminé
Saisir le nombre NbSim de simulations : 10000
Une estimation de la probabilité qu'un nombre entier tiré au hasard entre 2 et 1
002 ait 3 pour plus petit diviseur premier est : 0.1624

Appuyez sur une touche pour fermer la console...
```

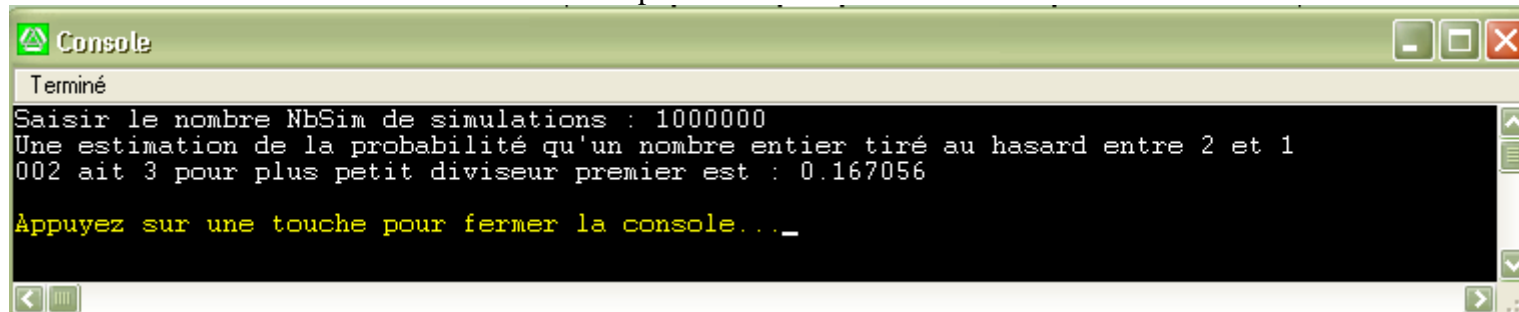
quasi instantanément :



```
Console
Terminé
Saisir le nombre NbSim de simulations : 100000
Une estimation de la probabilité qu'un nombre entier tiré au hasard entre 2 et 1
002 ait 3 pour plus petit diviseur premier est : 0.16649

Appuyez sur une touche pour fermer la console...
```

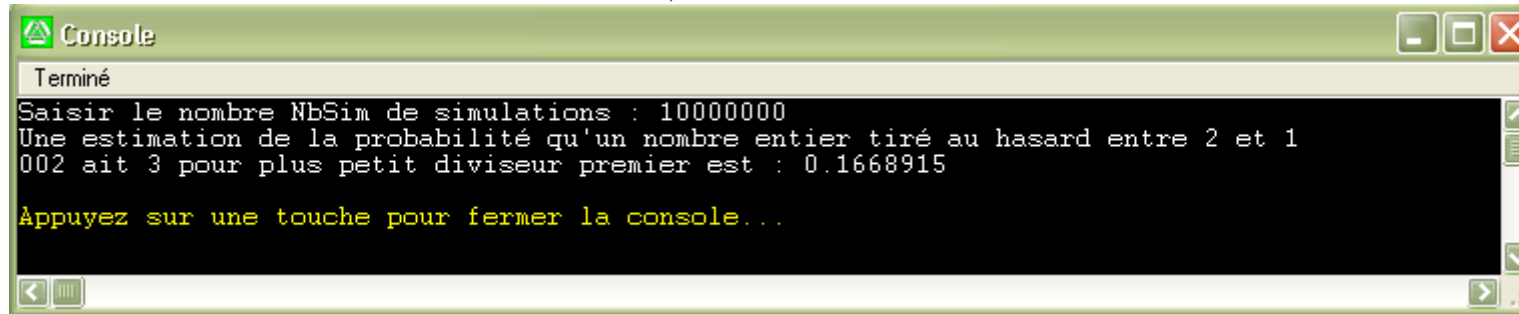
puis au bout de 23 s :



```
Console
Terminé
Saisir le nombre NbSim de simulations : 1000000
Une estimation de la probabilité qu'un nombre entier tiré au hasard entre 2 et 1
002 ait 3 pour plus petit diviseur premier est : 0.167056

Appuyez sur une touche pour fermer la console..._
```

et enfin, au bout de 3 min 52 s :



```
Console
Terminé
Saisir le nombre NbSim de simulations : 10000000
Une estimation de la probabilité qu'un nombre entier tiré au hasard entre 2 et 1
002 ait 3 pour plus petit diviseur premier est : 0.1668915
Appuyez sur une touche pour fermer la console...
```

On doit pouvoir trouver la valeur exacte...

Il suffit de compter combien, parmi les 1001 entiers possibles, il y a d'entiers en question (ayant 3 pour plus petit diviseur premier) et de faire le ratio suivant le 1^{er} principe de Laplace !

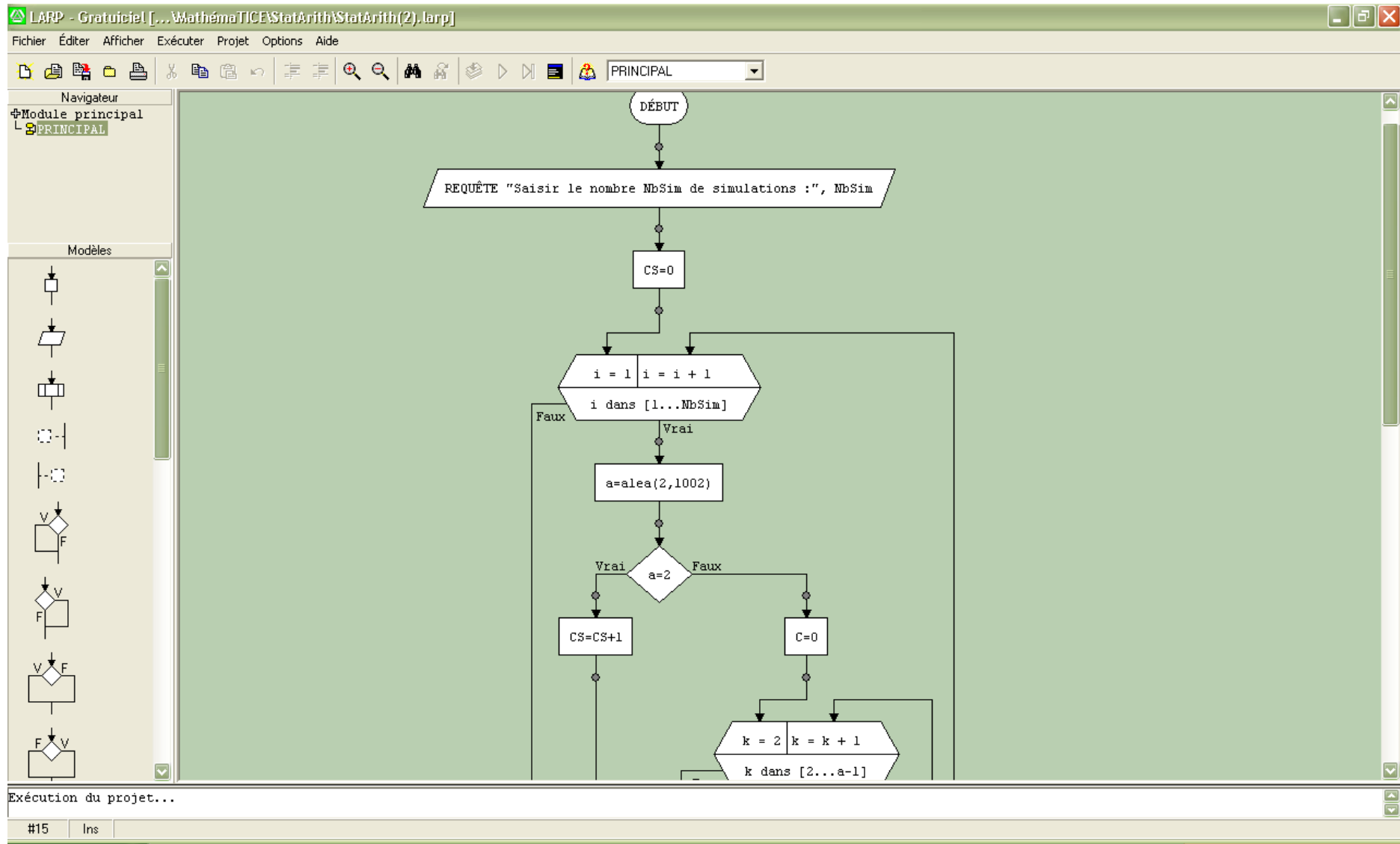
Bref, vous comptez si vous voulez, moi je n'ai pas trop envie...

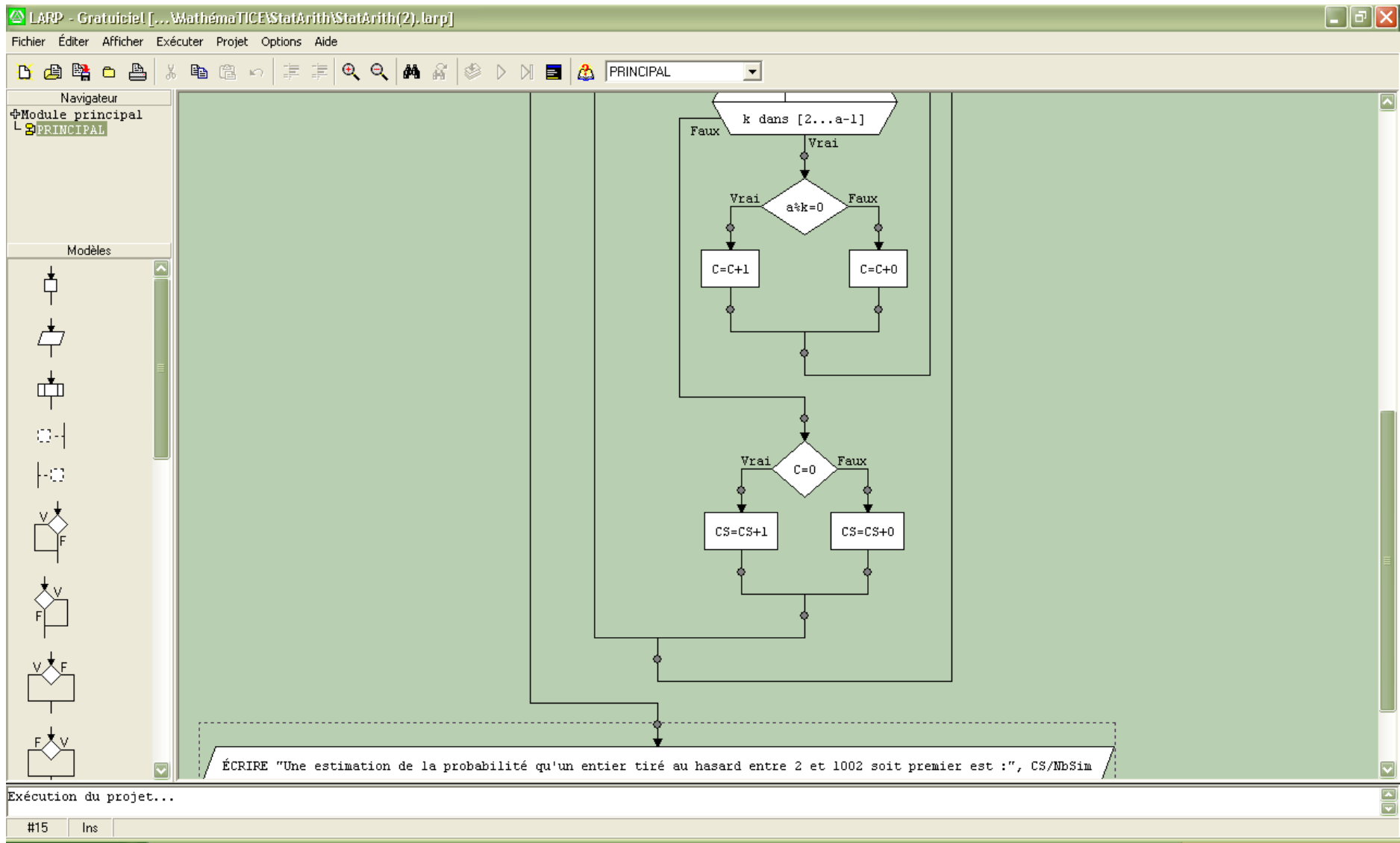
En tout cas je dis que :

Si on élève répond au problème comme cela (cet algorithme avec la conclusion de la console d'exécution ci-dessus), je dis bravo et je valide !!!

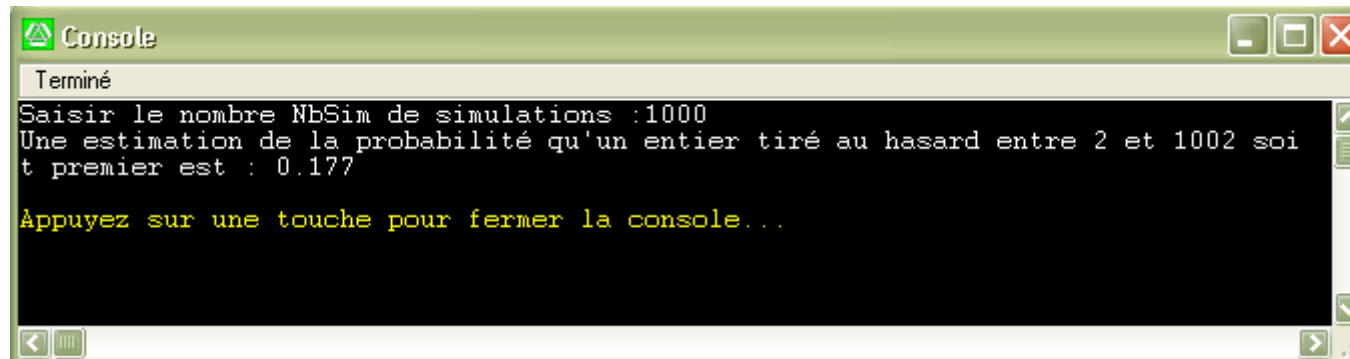
Et on laisse la démo, puisqu'elle existe ici, aux matheux... ou disons ici aux passionnés de résultats exacts et qui sont un peu patients ☺

2. Sur la probabilité qu'un nombre entier tiré au hasard entre 2 et 1002 soit premier...



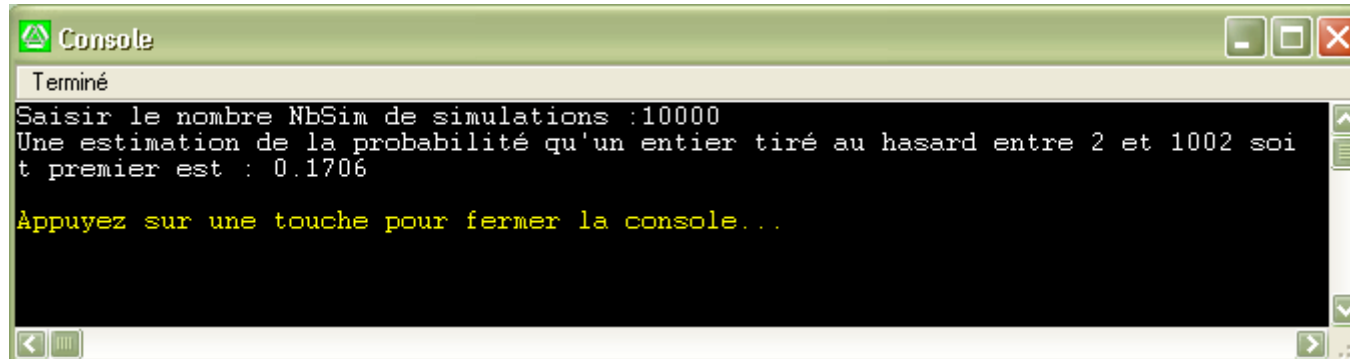


On obtient en 10 s :



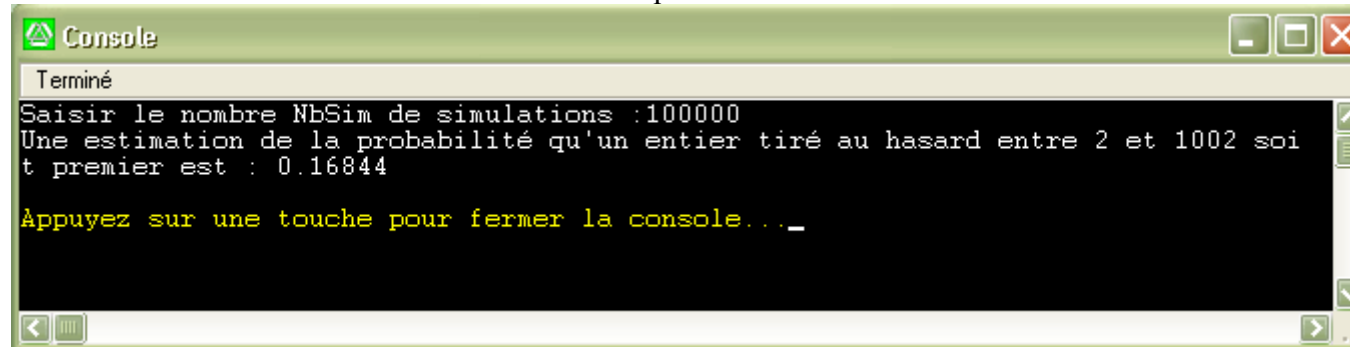
```
Console
Terminé
Saisir le nombre NbSim de simulations :1000
Une estimation de la probabilité qu'un entier tiré au hasard entre 2 et 1002 soit premier est : 0.177
Appuyez sur une touche pour fermer la console...
```

au bout d'1 min 50 s :



```
Console
Terminé
Saisir le nombre NbSim de simulations :10000
Une estimation de la probabilité qu'un entier tiré au hasard entre 2 et 1002 soit premier est : 0.1706
Appuyez sur une touche pour fermer la console...
```

et au bout d'une quinzaine de minutes...



```
Console
Terminé
Saisir le nombre NbSim de simulations :100000
Une estimation de la probabilité qu'un entier tiré au hasard entre 2 et 1002 soit premier est : 0.16844
Appuyez sur une touche pour fermer la console..._
```

sachant que le résultat exact est $168/1001$! (selon toujours le 1^{er} principe de Laplace)

c'est un peu plus long

et je sens que je peux améliorer tout ça

notamment il n'est pas utile de faire toutes les divisions et il suffit de s'arrêter dès que l'une tombe juste
mais bon.... Je vais réfléchir...

3. Sur l'algorithme de la soustraction pour calculer le PGCD...

LARP - Gratuitiel [...mes doc stef\MathémaTICE\StatArith\PGCDSoust.larp]

Fichier Éditer Afficher Exécuter Projet Options Aide

PRINCIPAL

Module principal
PRINCIPAL

Modèles

```
graph TD
    DEBUT([DÉBUT]) --> A[a=alea(2,1002)]
    A --> B[b=alea(2,1002)]
    B --> ECR[ÉCRIRE "a=",a,"et b=",b]
    ECR --> AB{a=b}
    AB -- Vrai --> CA[c=a]
    AB -- Faux --> ABG{a<>b}
    ABG -- Vrai --> CM[c=min(a,b)]
    ABG -- Faux --> BM[b=max(a,b)-min(a,b)]
    BM --> ABG
```

Console

```
Terminé
a= 700 et b= 840
700 140
140 560
140 420
140 280
140 140
PGCD= 140
Appuyez sur une touche pour fermer la console...
```

Exécution du projet...

la suite de l'algorithme avec une autre exécution...

The screenshot shows the LARP software interface. The main window displays a flowchart for calculating the GCD of two numbers. The flowchart starts with a process box $c = \min(a, b)$, followed by $b = \max(a, b) - \min(a, b)$, then $a = c$, and an output box $\text{ÉCRIRE } a, b$. A loop structure is shown with a decision diamond and a process box $\text{PGCD} = c$. The flowchart ends with an output box $\text{ÉCRIRE "PGCD=", } c$ and a terminal box FIN .

The console window, titled "Console", shows the execution results:

```
Terminé
a= 188 et b= 778
188 590
188 402
188 214
188 26
26 162
26 136
26 110
26 84
26 58
26 32
26 6
6 20
6 14
6 8
6 2
2 4
2 2
PGCD= 2
```

The status bar at the bottom indicates "Exécution du projet..." and "Ins".

4. Sur l'algorithme d'Euclide pour calculer le PGCD...

The screenshot shows a software application window titled "LARP - Gratuitiel [...]mes doc stef\MathémaTICE\StatArith\PGCDEucl.larp". The interface includes a menu bar (Fichier, Éditer, Afficher, Exécuter, Projet, Options, Aide), a toolbar with various icons, and a "PRINCIPAL" dropdown menu. On the left, there is a "Navigateur" pane showing a tree structure with "Module principal" and "PRINCIPAL", and a "Modèles" pane with various flowchart symbols. The main workspace displays a flowchart for the Euclidean algorithm:

- Start: DÉBUT
- Process: $a = \text{alea}(2, 1002)$
- Process: $b = \text{alea}(2, 1002)$
- Output: ÉCRIRE "a=", a, "et b=", b
- Process: $r = \max(a, b) \% \min(a, b)$
- Decision: $r = 0$ (Vrai/Faux)
- Path (Vrai): Process $c = a$
- Path (Faux): Decision $r <> 0$ (Vrai/Faux)
- Path (Vrai): Process $a = \min(a, b)$
- Path (Faux): Process $b = r$
- Loop: The $r <> 0$ decision and the $a = \min(a, b)$ / $b = r$ processes form a loop.

Overlaid on the flowchart is a "Console" window titled "Terminé" with the following output:

```
Terminé
a= 624 et b= 782
624 158
158 150
150 8
8 6
6 2
PGCD= 2

Appuyez sur une touche pour fermer la console...
```

At the bottom of the application window, there is a status bar with the text "Exécution du projet..." and a button labeled "Ins".

la suite de l'algorithme avec une autre exécution...

The screenshot shows the LARP (Logiciel de Apprentissage et de Représentation des Algorithmes) software interface. The main window displays a flowchart for the Euclidean algorithm. The flowchart starts with a process box $a = \min(a, b)$, followed by $b = r$, $r = a \% b$, and an output box $\text{ÉCRIRE } a, b$. A loop is formed by a decision diamond (V for true, F for false) that branches back to the $a = \min(a, b)$ box. After the loop, the process box $c = b$ is executed, followed by $\text{PGCD} = c$ and a final output box $\text{ÉCRIRE "PGCD=", } c$.

On the left side, there is a 'Modèles' (Models) panel with various flowchart symbols. The top menu includes 'Fichier', 'Éditer', 'Afficher', 'Exécuter', 'Projet', 'Options', and 'Aide'. The toolbar contains icons for file operations, navigation, and execution. The 'PRINCIPAL' project is selected in the dropdown menu.

A 'Console' window is open in the foreground, showing the following output:

```
Terminé
a= 459 et b= 207
207 45
45 27
27 18
18 9
PGCD= 9
Appuyez sur une touche pour fermer la console..._
```

At the bottom of the interface, the text 'Exécution du projet...' is visible.

4. Sur la loi du PGCD...

Il faut passer à \mathbf{R} : Monsieur Raymondaud ?