

Algorithmique : mesure principale d'un angle orienté

Niveau

Première S, première STI2D.

Situation

On considère l'algorithme suivant :

Entrée : a est un nombre positif, mesure en degrés d'un angle.

Algorithme :

Tant que $a > 180$ faire

$a \leftarrow a - 360$

Sortie : a

1. Que fait cet algorithme ?
2. Modifier l'algorithme de sorte que l'entrée a (et la sortie) soit une mesure en radians.
3. Programmer cet algorithme.
4. Tester le programme avec un angle de mesure $\frac{14\pi}{3}$
5. Modifier l'algorithme pour déterminer la valeur exacte du résultat.

Commentaires

- Un prolongement possible : la mesure de l'angle peut être un nombre négatif
- À la question 2., la réponse attendue est :

Entrée : a est un nombre positif, mesure en **radians** d'un angle.

Algorithme :

Tant que $a > \pi$ faire

$a \leftarrow a - 2\pi$

Sortie : a

- Un tel algorithme aboutit à des calculs en valeurs approchées (sauf sur des langages utilisant le calcul symbolique), c'est pourquoi on demande de modifier l'algorithme à la question 5. Par exemple :

Entrée : a et b des entiers positifs, tels que $\frac{a\pi}{b}$ est une mesure en

radians d'un angle.

Algorithme :

Tant que $a > b$ faire

$a \leftarrow a - 2b$

Sortie : a

Implantations de la question 2.

Casio

```
====MESPRINC====
?→A↵
While A>π↵
A-2π→A↵
WhileEnd↵
A↵
```

Texas Instruments

```
PROGRAM:MESPRINC
:Input A
:While A>π
:A-2π→A
:End
:Disp A
:
```

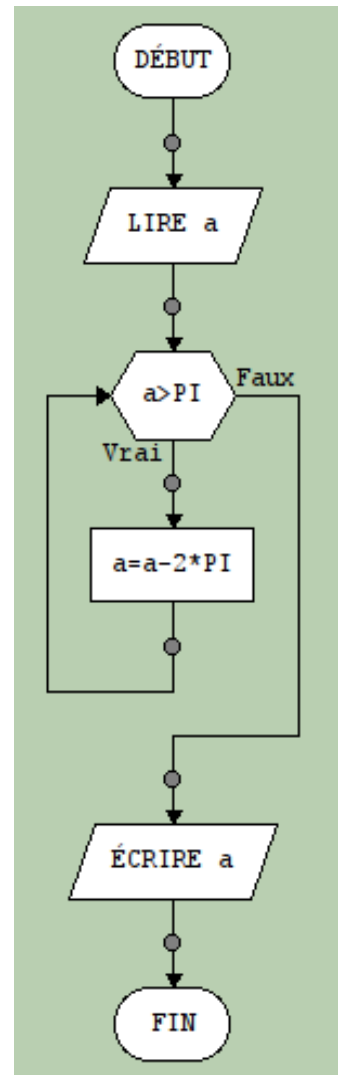
Algobox

Le nombre π a pour syntaxe Math.PI

```
1  VARIABLES
2  a EST_DU_TYPE NOMBRE
3  DEBUT_ALGORITHME
4  LIRE a
5  TANT_QUE (a>Math.PI) FAIRE
6  DEBUT_TANT_QUE
7  a PREND_LA_VALEUR a-2*Math.PI
8  FIN_TANT_QUE
9  AFFICHER a
10 FIN_ALGORITHME
```

Larp

Le nombre π a pour syntaxe PI. Il semble qu'on ne puisse pas utiliser cette syntaxe pour entrer la valeur de a.



Python

Le nombre π a pour syntaxe math.pi, ou plus simplement pi si on l'importe depuis le module « math » au début du programme.

```
from math import pi
```

```
a=input("entrer la valeur de a : ")
```

```
while (a>pi):
    a=a-2*pi
```

```
print a
```
