

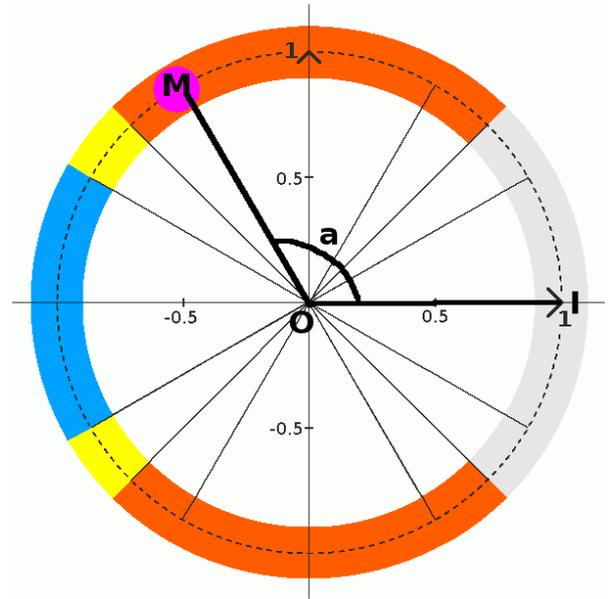
## Situation :

Un programmeur a simulé un jeu qui consiste à lancer une bille qui va rouler dans la couronne extérieure d'un disque et freiner au fur et à mesure pour s'arrêter sur l'une des zones de couleurs. L'objectif est de déterminer la zone sur laquelle va s'arrêter la bille en fonction de la poussée initiale de la bille.

Le centre de la bille se déplace sur le cercle de rayon 1 dans le sens trigonométrique.

La bille est initialement positionnée au point I (1 ; 0)

L'angle orienté  $(\vec{OI}, \vec{OM})$  a pour mesure en radians  $a$



Fichier initial : **roue.py**

## Partie A : Faire rouler la bille

Le déplacement de la bille est donné par le code suivant et elle est positionnée sur le cercle à l'aide de l'angle  $a$

```

a=0                #Valeur initiale de l'angle a
for k in range(n): #Pour k allant de 0 à n-1
    pas=(n-k)*pi/(60*n) #Calcul de pas qui va diminuer au fur et à mesure
    a=a+pas            #L'angle a augmente de pas
    bille.goto(..... ,..... ) #coordonnées (x,y) de la bille en fonction de a
    
```

- 1) Compléter les pointillés en exprimant en fonction de  $a$  les coordonnées cartésiennes  $(x ; y)$  de la bille .
- 2) Saisir à la ligne 28, le code ci-dessus et exécuter le programme en prenant pour  $n$  différentes valeurs entières comprises entre 0 et 1000 (ligne 26 à modifier).
- 3) Trouver, en tâtonnant, la valeur de  $n$  (ligne 26) pour laquelle la bille effectue exactement un tour complet de roue et donc se retrouve au point I de coordonnées (1,0) ? .....

## Partie B : Comprendre le déplacement de la bille

- 1) Pour  $n=5$ , on va exécuter le code ci-dessus manuellement et remplir le tableau suivant qui donne les valeurs des variables  $a$  et  $pas$  au fur et à mesure.

k	0	1	2	3	4
pas (en fonction de $\pi$ )	$\frac{5\pi}{60 \times 5} = \frac{5\pi}{300}$	$\frac{4\pi}{60 \times 5} = \frac{4\pi}{300}$			
a (en fonction de $\pi$ )	$0 + \frac{5\pi}{300} = \frac{5\pi}{300}$				

Remarque : la dernière valeur de  $a$  est celle qui détermine la position d'arrêt de la bille.

2) Dans le cas général ( $n$  quelconque), les valeurs de **pas** (pour  $k$  allant de 0 à  $n-1$ ) sont les suivantes :

$k$	0	1	2	...etc...	$n-1$
<b>pas</b> (en fonction de $\pi$ )	$\frac{n\pi}{60n}$	$\frac{(n-1)\pi}{60n}$	$\frac{(n-2)\pi}{60n}$	...etc...	$\frac{\pi}{60n}$

a) L'angle final  $a$  au moment où la bille s'arrête est la somme des  $n$  termes **pas** du tableau ci-dessus.

On a donc, au moment où la bille s'arrête :  $a = \frac{n\pi}{60n} + \frac{(n-1)\pi}{60n} + \frac{(n-2)\pi}{60n} + \dots + \frac{\pi}{60n}$  Montrer que  $a = \frac{(n+1)\pi}{120}$

b) Calculer alors les différentes valeurs de  $n$  pour que la bille se retrouve aux points demandés.

La bille s'arrête au point de coordonnées ...	(1 ; 0) après exactement 1 tour	(1 ; 0) après exactement 5 tours	(-1 ; 0)	(0 ; 1)
$n$				

### Partie C : Valeur de $n$ choisie aléatoirement.

Remplacer la ligne 26 par : `n=randint(0,1199)` # $n$  nombre entier aléatoire inférieur à 1199.

Lorsque la bille s'arrête, on voudrait voir s'afficher dans la console le gain obtenu suivant la zone de couleur.

Compléter ci-dessous les pointillés et saisir les lignes de script à partir de la ligne 33.

Vérifier que le programme fonctionne correctement.

```

if cos(a)>sqrt(2)/2:           #la bille se retrouve dans la zone .....
    print("Tu peux rejouer !")
if -sqrt(2)/2<cos(a)<sqrt(2)/2: #la bille se retrouve dans la zone .....
    print("PERDU !")
if .....:                   #la bille se retrouve dans la zone bleue foncée
    print("Tu as gagné un petit lot")
if .....:                   #la bille se retrouve dans la zone jaune
    print("BRAVO, tu as gagné un gros lot !")

```