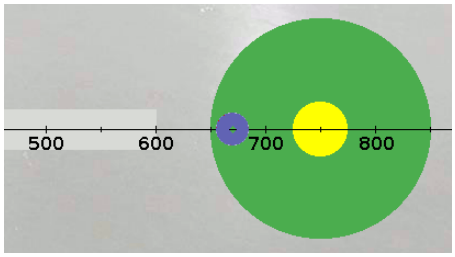
Le palet glissant

Situation :
Un jeune informaticien a écrit un programme qui simule le lancer d'un palet que l'on fait glisser en lui donnant une impulsion initiale.
L'objectif est bien sûr d'essayer d'atteindre le centre de la cible.



Étape 1 : Un premier déplacement programmé

1) Ouvrir le programme **Palet_glissant.py**.

➔ Exemple 1 : le palet est placé initialement au point d'abscisse **`depart=130`** (ligne 26)
A la ligne 31, le joueur a choisi **`impulsion=23`**.
a) Exécuter le programme pour visualiser le déplacement du palet.
b) Trouver, en tâtonnant, la valeur de **`impulsion`** pour que le palet atteigne le centre de la cible.
Impulsion trouvée : ……

➔ Exemple 2 : le palet est placé initialement au point d'abscisse **`depart=38`**. (ligne 26 à modifier donc)
De la même façon, quelle **`impulsion`** donnée au palet pour qu'il atteigne le centre de la cible ?
Impulsion trouvée : ……

2) La cible est composée de deux disques concentriques de rayons respectifs 25 et 100.

Compléter l'algorithme ci-contre et le traduire en Python dans le programme.

Aide technique : l'abscisse du palet est obtenue avec l'instruction **`palet.xcor()`**

```
Si  …….<abscisse palet<……. :
      afficher « Le palet est  sur la zone jaune »
Sinon:
      Si …….<abscisse palet<……. ou …….<abscisse palet<……. :
            afficher « ……. »
      Sinon
            afficher « Perdu »
```

3) **La stratégie gagnante :**

a) Exprimer en fonction de la variable **`depart`** (abscisse initiale du palet) la distance, en pixels, qui sépare le palet du centre de la cible d'abscisse 750 : ……

b) Exprimer en fonction de **`impulsion`**, la distance totale parcourue par le palet après exécution de la boucle **`for`**
……

c) Pour atteindre le centre de la cible, la distance exprimée en a) doit être égale à la distance exprimée en b).
Comment calculer alors **`impulsion`** pour être certain de gagner ? ……
……

d) Vérification sur un exemple : le palet est placé initialement au point d'abscisse **`depart=217`**.
`impulsion` gagnante pour atteindre le centre de la cible : ……
Vérifier votre valeur en exécutant le script.

Étape 2 : un autre déplacement programmé

Le programmeur souhaite modifier le code de déplacement du palet et propose la modification ci-contre.

```
for k in range(10):
    palet.forward((10-k)*impulsion)
```

En vous inspirant de la question 3), trouver une méthode de calcul de **`impulsion`** qui permette d'atteindre le centre de la cible à tous les coups.

……
……
……