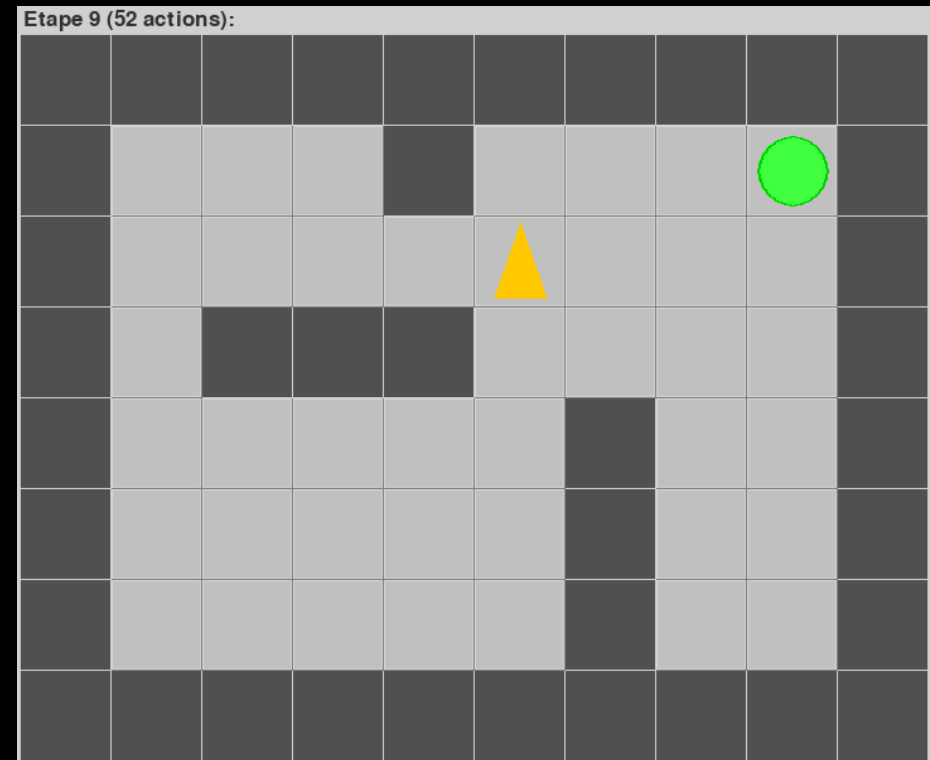


## Loan SENS



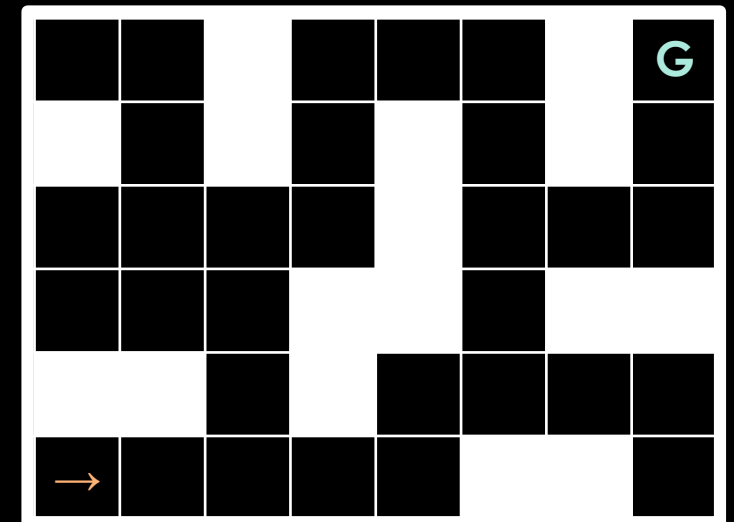
# OBJECTIF

- Aider le personnage à s'échapper du labyrinthe !
- -> Programmer une **intelligence artificielle**



# CONVENTION

- Le personnage ne peut pas traverser un mur
- Il existera forcément un chemin entre le joueur et l'objectif
- Le personnage débute dans le **coin inférieur gauche** orienté à droite
- L'objectif est dans le **coin supérieur droit**
- La taille du labyrinthe est de 8 x 6



# PRIMITIVES

= instructions que l'on peut utiliser en rapport avec le jeu

- *Game*:

- `moveForward()` : Avance le joueur devant lui (si la case n'est pas occupée par un mur)
- `turnLeft()` / `turnRight()` : Effectue une rotation du joueur de 90° sur la gauche/droite
- `isFreeForward()` / `isGoalForward()` -> `bool` : Vérifie si la case devant est libre/l'objectif
- `direction()` -> `str(right/up/left/down)` : Donne la direction du joueur

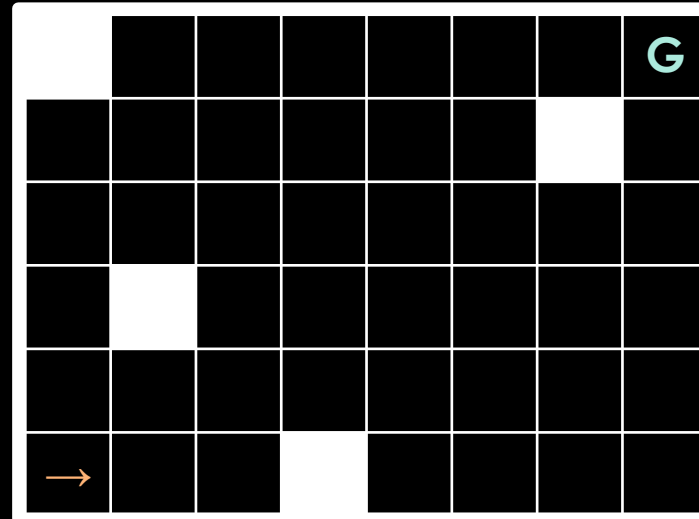
# CODE EXÉCUTÉ

- Etape (fonction `step()`)
  - Sera exécuté en boucle
  - Si le joueur est sur l'objectif à la fin d'une étape, la partie se termine avec une **victoire**
  - Après 1000 étapes sans terminer sur l'objectif, la partie est **perdue**
- `debug(<nom de la carte>, <pas à pas>=False)`: permet de tester ton code sur une carte en particulier. Montrera l'état du labyrinthe à la fin de chaque étape
- `test([<nom de carte>])`: test ton code sur les différentes cartes disponibles

# PROJET

- Fichiers utile:
  - **ia.py** -> code de ton IA
  - **main.py** -> code pour tester ton IA
- Dossier utile:
  - **maze** -> contient différent labyrinthe, tu peux aussi y créer les tiens. Leur nom est le nom du fichier

```
X.....  
.....X.  
.....  
.X.....  
.....  
....X....
```



# SCORE

- Objectif principal : Guider le joueur vers l'objectif
- Objectif secondaire : Le faire de manière efficace !
  - Toute les actions coutent des points d'action (PA), votre but est d'en avoir le **moins** possible en ayant atteint l'objectif

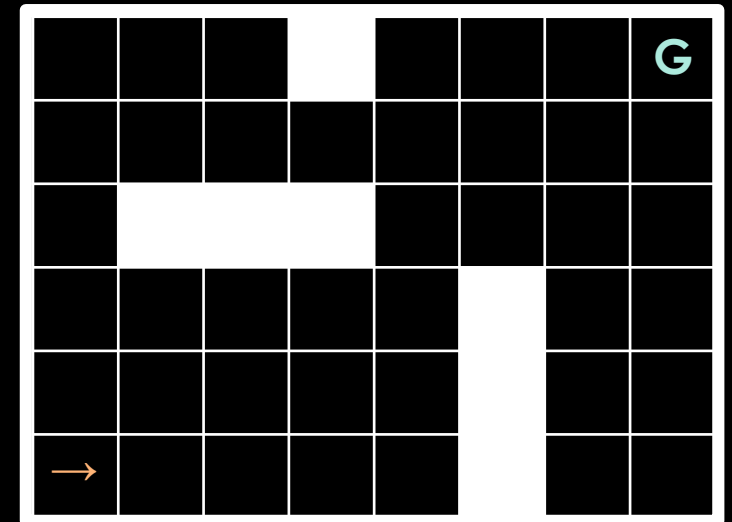
moveForward	turn...	is...forward	direction
5	2	1	0

- Le score sera de 0% si le labyrinthe n'est pas résolu
- Si il est résolu, le score dépend du nombre de PA (plus le nombre de PA est grand, plus le score sera bas)



# EXAMPLE

```
def step(game):  
    if game.isFreeForward() or game.isGoalForward():  
        game.moveForward()  
    else:  
        if game.direction() == "up":  
            game.turnRight()  
        else:  
            game.turnLeft()
```

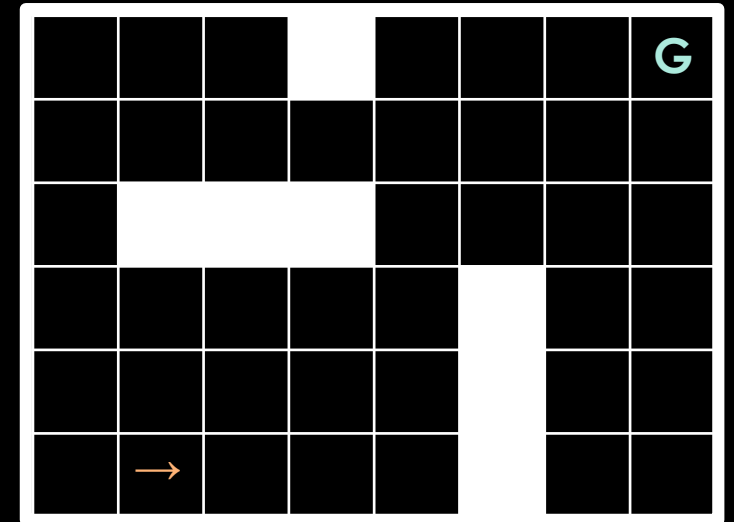


Etapes	Point d'actions
0	0



# EXAMPLE

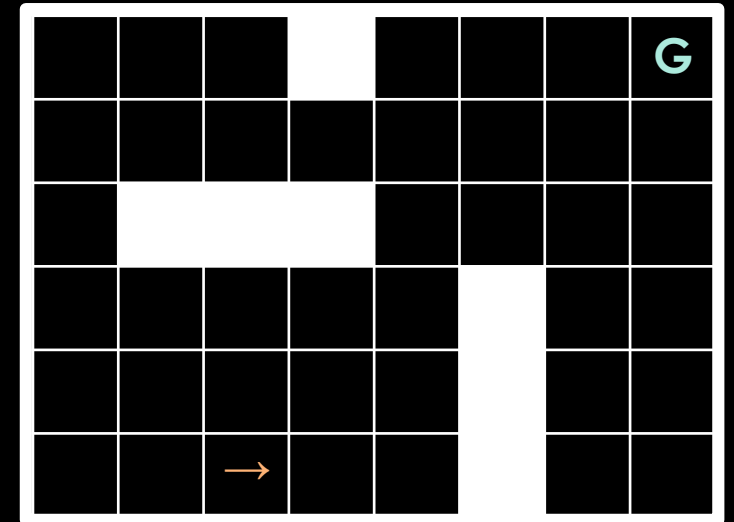
```
def step(game):
    if game.isFreeForward() or game.isGoalForward():
        game.moveForward()
    else:
        if game.direction() == "up":
            game.turnRight()
        else:
            game.turnLeft()
```



Etapes	Point d'actions
1	6

# EXAMPLE

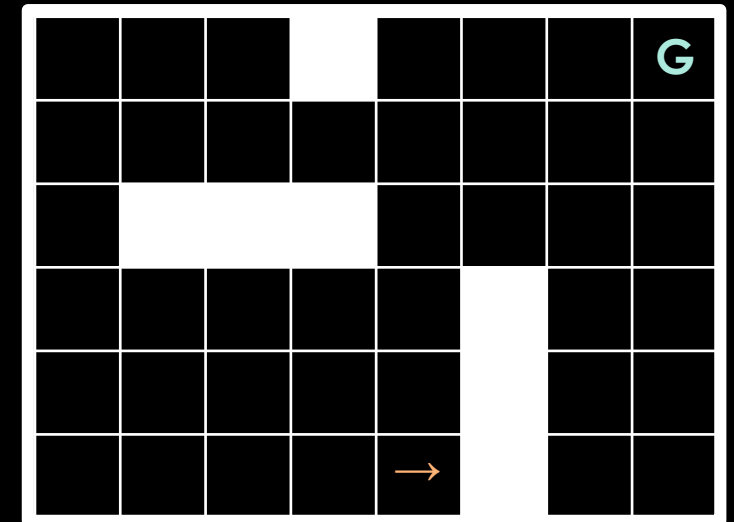
```
def step(game):
    if game.isFreeForward() or game.isGoalForward():
        game.moveForward()
    else:
        if game.direction() == "up":
            game.turnRight()
        else:
            game.turnLeft()
```



Etapes	Point d'actions
2	12

# EXAMPLE

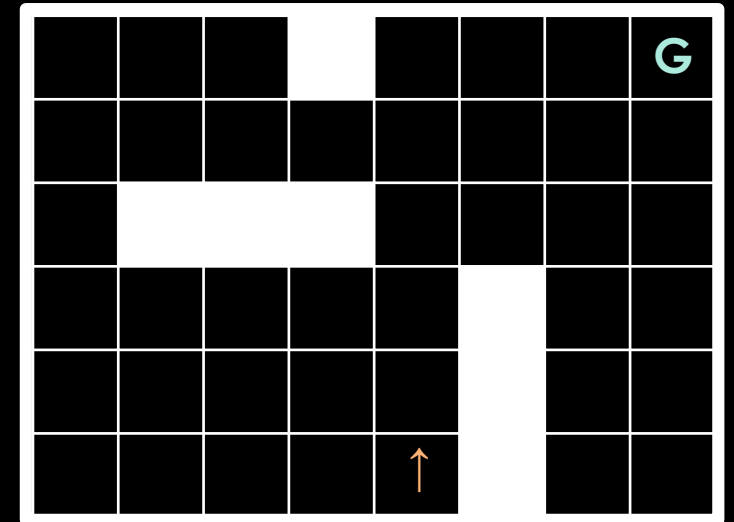
```
def step(game):
    if game.isFreeForward() or game.isGoalForward():
        game.moveForward()
    else:
        if game.direction() == "up":
            game.turnRight()
        else:
            game.turnLeft()
```



Etapes	Point d'actions
4	24

# EXAMPLE

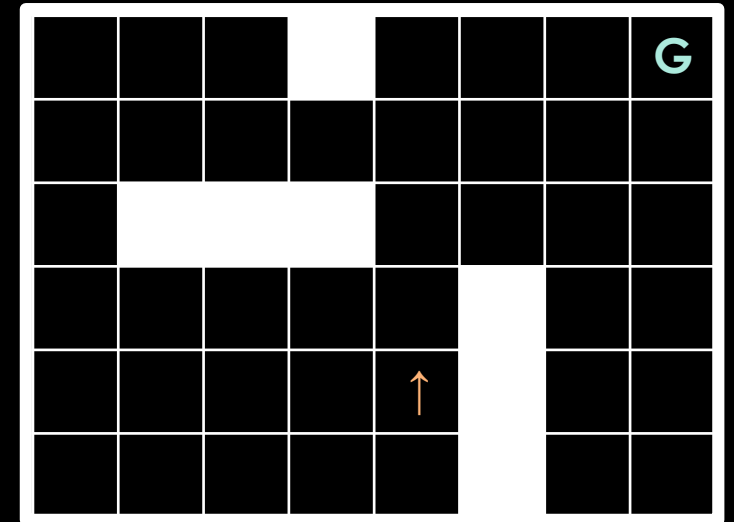
```
def step(game):
    if game.isFreeForward() or game.isGoalForward():
        game.moveForward()
    else:
        if game.direction() == "up":
            game.turnRight()
        else:
            game.turnLeft()
```



Etapes	Point d'actions
5	28

# EXAMPLE

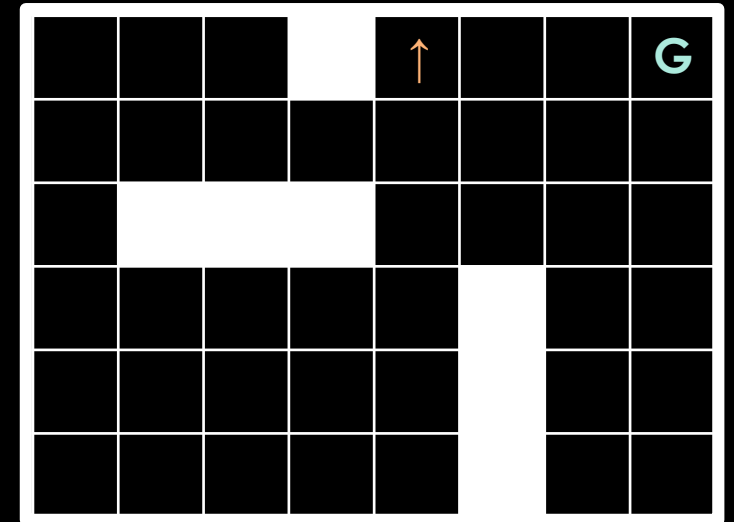
```
def step(game):  
    if game.isFreeForward() or game.isGoalForward():  
        game.moveForward()  
    else:  
        if game.direction() == "up":  
            game.turnRight()  
        else:  
            game.turnLeft()
```



Etapes	Point d'actions
6	34

# EXAMPLE

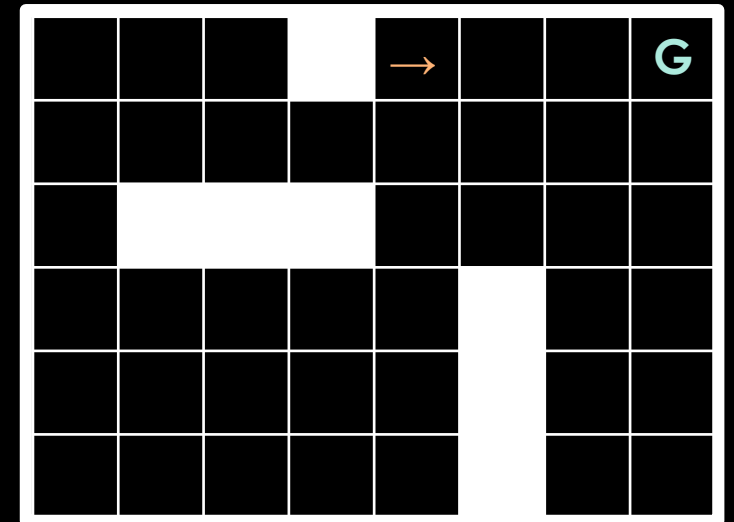
```
def step(game):
    if game.isFreeForward() or game.isGoalForward():
        game.moveForward()
    else:
        if game.direction() == "up":
            game.turnRight()
        else:
            game.turnLeft()
```



Etapes	Point d'actions
10	58

# EXAMPLE

```
def step(game):
    if game.isFreeForward() or game.isGoalForward():
        game.moveForward()
    else:
        if game.direction() == "up":
            game.turnRight()
        else:
            game.turnLeft()
```

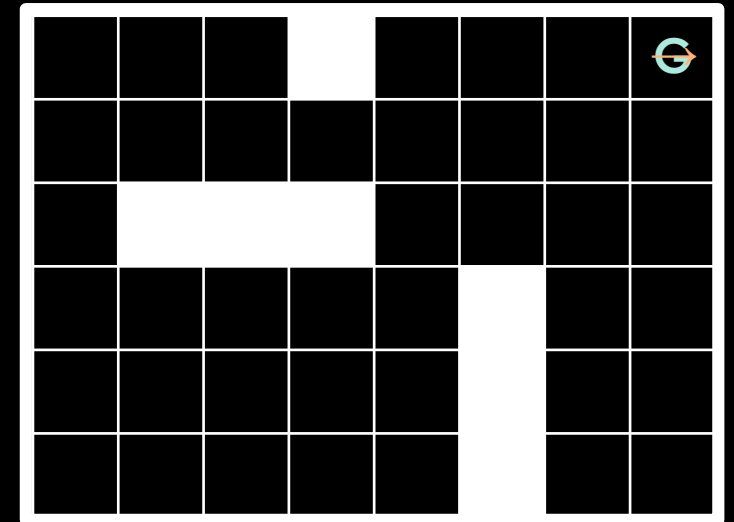


Etapes	Point d'actions
11	62



# EXAMPLE

```
def step(game):
    if game.isFreeForward() or game.isGoalForward():
        game.moveForward()
    else:
        if game.direction() == "up":
            game.turnRight()
        else:
            game.turnLeft()
```




Etapes	Point d'actions
14	81

# PISTES

- Réfléchissez à ce que tu veux que le joueur fasse avant de le programmer
- Avec les contraintes qu'on a posé, il y a certaines astuces que l'on peut exploiter...
- A vous de toi de jouer maintenant ! :)

# SITE (INGINIOUS)

-> Donnera un score à ton code sur plusieurs labyrinthes (cachés)


INGInious
> [DOJO] Labyrinthe
> Concours Labyrinthe IA
Liste des cours
Loan Prof

## Concours Labyrinthe IA

Copie ici le contenu de ton fichier `ia.py`

Votre réponse a passé les tests ! Votre note est de 91.31%.

[Soumission #5e1388b1e10b245845ab90e7]

### Victoire !

**Nombre de labyrinthe résolu:**  
9 / 9

**Score:**  
91.31 %

Voici les résultats du test :

- easy1 :	OK (191 actions en 17 étapes)	Score : 87.65%
- easy2 :	OK (191 actions en 17 étapes)	Score : 87.65%
- easy3 :	OK (239 actions en 21 étapes)	Score : 91.36%
- medium1 :	OK (251 actions en 23 étapes)	Score : 90.87%
- medium2 :	OK (215 actions en 19 étapes)	Score : 92.5%
- medium3 :	OK (191 actions en 17 étapes)	Score : 93.89%
- hard1 :	OK (373 actions en 34 étapes)	Score : 93.43%
- hard2 :	OK (371 actions en 33 étapes)	Score : 94.0%
- hard3 :	OK (241 actions en 22 étapes)	Score : 90.45%

### Code de l'IA

Ceci n'est pas un debugger, tester votre code localement avant de le soumettre

### Informations

Auteur(s)	Loan
Date limite	Pas de date limite
Etat	Réussi
Note	91.31%
Poids de la note	1.0
Nombre d'essais	2
Limite de soumission	Pas de limite

### Soumission en tant que

> Loan Prof

Classe : Default classroom

Scoreboard: Concours Labyrinthe

### Pour évaluation

i Meilleure soumission

> 06/01/2020 14:21:21 - 91.31%

### Historique des soumissions

06/01/2020 14:21:21 - 91.31%
06/01/2020 14:20:41 - 44.44%

# SITE SCOREBOARD

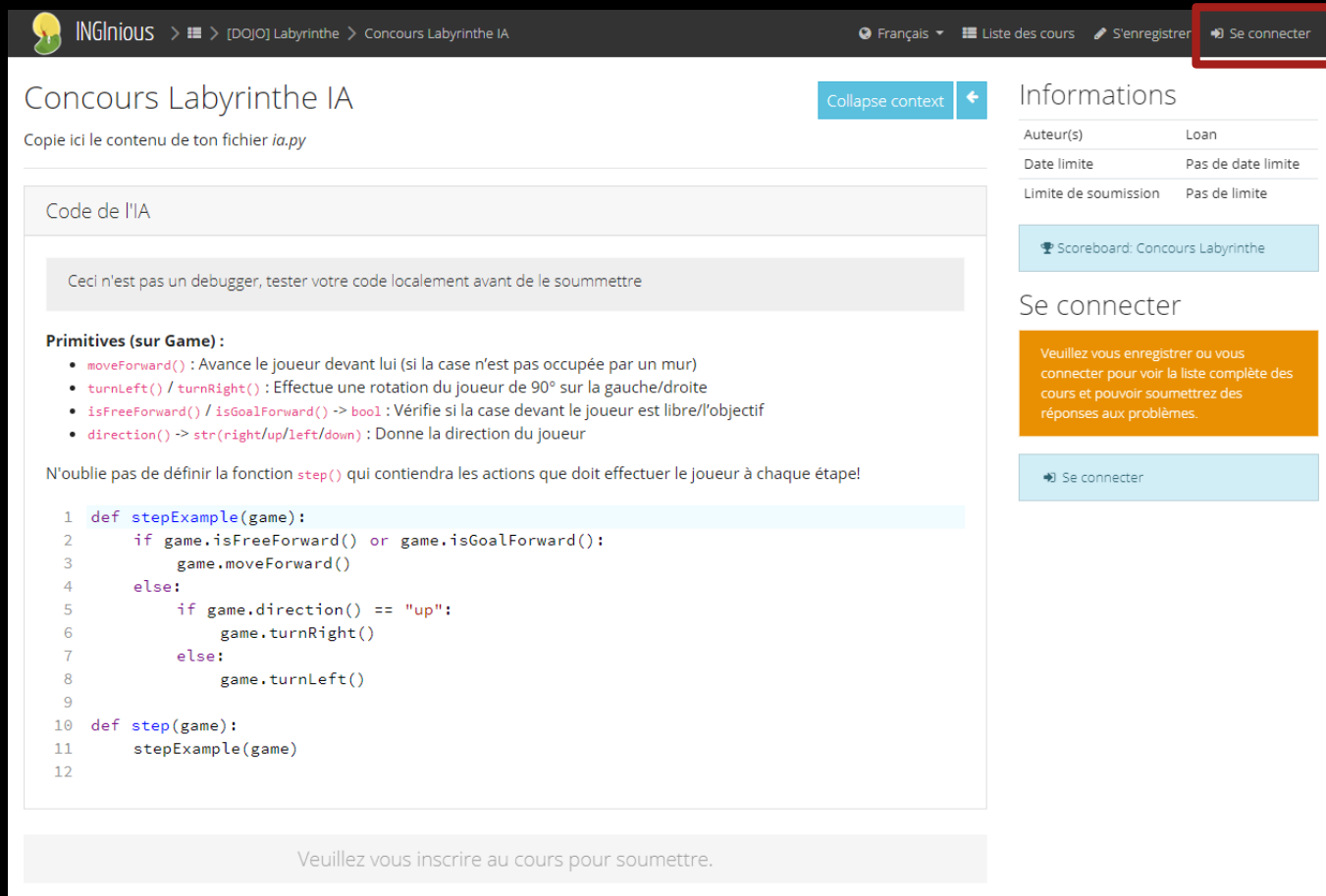
Te permet de consulter le score des différents utilisateurs ayant participé.  
Seras tu capable de battre mon score ?



The screenshot shows the INGINIOUS website interface. The top navigation bar includes the INGINIOUS logo, a breadcrumb trail: [DOJO] Labyrinthe > Concours Labyrinthe, and links for 'Liste des cours' and 'Loan Prof'. Below the navigation bar, the title 'Concours Labyrinthe' is displayed. A table shows the top two scores: Loan Prof with a score of 91.31 and Bob with a score of 44.44.

Concours Labyrinthe	
Student(s)	Score
1	Loan Prof
2	Bob

# SITE SE CONNECTER



INGInious > [DOJO] Labyrinthe > Concours Labyrinthe IA

Français Liste des cours S'enregistrer Se connecter

## Concours Labyrinthe IA

Copie ici le contenu de ton fichier `ia.py`

Code de l'IA

Ceci n'est pas un debugger, tester votre code localement avant de le soumettre

**Primitives (sur Game):**

- `moveForward()` : Avance le joueur devant lui (si la case n'est pas occupée par un mur)
- `turnLeft()` / `turnRight()` : Effectue une rotation du joueur de 90° sur la gauche/droite
- `isFreeForward()` / `isGoalForward()` -> bool : Vérifie si la case devant le joueur est libre/l'objectif
- `direction()` -> str(right/up/left/down) : Donne la direction du joueur

N'oubliez pas de définir la fonction `step()` qui contiendra les actions que doit effectuer le joueur à chaque étape!

```
1 def stepExample(game):
2     if game.isFreeForward() or game.isGoalForward():
3         game.moveForward()
4     else:
5         if game.direction() == "up":
6             game.turnRight()
7         else:
8             game.turnLeft()
9
10 def step(game):
11     stepExample(game)
12
```

Veuillez vous inscrire au cours pour soumettre.

### Informations

Auteur(s)	Loan
Date limite	Pas de date limite
Limite de soumission	Pas de limite

Scoreboard: Concours Labyrinthe

### Se connecter

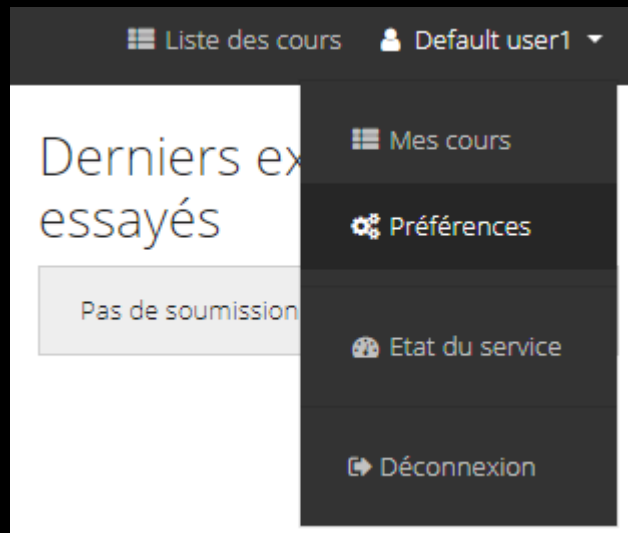
Veuillez vous enregistrer ou vous connecter pour voir la liste complète des cours et pouvoir soumettre des réponses aux problèmes.

Se connecter

Remplace « XX » par  
ce que le prof te dit  
de mettre

# SITE CHANGER CES PRÉFÉRENCES

- Tu peux ensuite modifier mot de passe et le nom affiché sur le site depuis les préférences



### Mon profil

Nom d'utilisateur :

Nom complet :


Adresse email :

Langue :

Mot de passe actuel :

Nouveau mot de passe (min. 6 caractères) :

Confirmez le nouveau mot de passe :

 Sauvegarder mon profil