

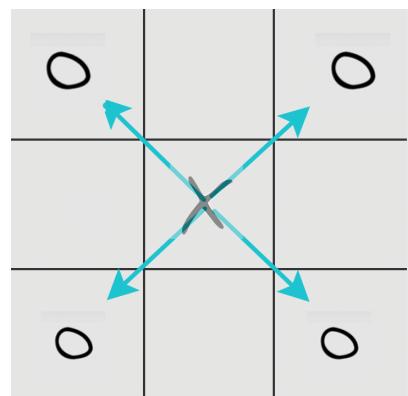
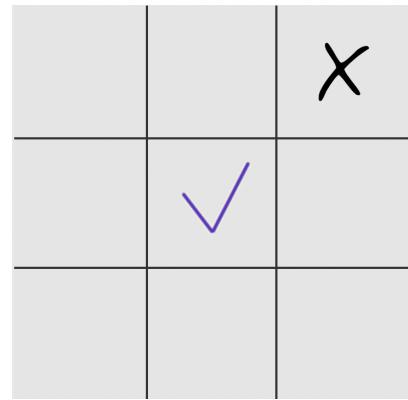
Projet Morpion Programmer la stratégie de l'ordinateur

```
morpion = [[' ', ' ', ' '],[' ', ' ', ' '],[' ', ' ', ' ']]  
  
def case_vide(grille,x,y):  
    """  
        @return True si grille[x][y] == ' '  
        False sinon  
    """  
  
def demander_joueur(texte):  
    """demande une saisie du joueur (sous forme  
d'un entier entre 0 et 2)  
tant que la saisie n'est pas conforme, demander  
à nouveau de saisir une valeur  
@return: val : int, valeur saisie par l'utilisateur  
"""  
  
def jeu_de_defense(grille,car_attaquant):  
    """recherche la première case libre qui permet  
de bloquer l'adversaire, si celui-ci possède 2  
caractères alignés.  
si aucune coordonnée de blocage possible,  
retourne None  
@return: jouer_case : un couple d'entiers où  
jouer_case[0] désigne la ligne et jouer_case[1] la colonne à jouer pour  
bloquer, ou None sinon  
"""  
  
def cherche_case_vide(grille,liste_X,liste_Y):  
    """recherche les coordonnées de la première  
case libre trouvée dans la grille à partir des listes de coordonnées  
donnée en argument.  
La coordonnée d'une case est à construire avec  
(liste_X[0],liste_Y[0]), ...  
@return: jouer_case : un couple d'entiers où x désigne la ligne et y la  
colonne à jouer pour bloquer, ou None sinon  
"""  
  
def jouer(grille,car,x,y):  
    """  
        place le caractère car dans la case (x,y)  
    """
```

1. Soyez vigilant dès le départ

Si votre adversaire commence par la case centrale, occupez un coin. Cela vous permettra de tendre un piège et d'augmenter vos chances de victoire ou, au moins, d'obtenir une égalité.

```
>>> x,y = cherche_case_vide(morpion,[0,0,2,2],  
[0,2,0,2])  
>>> jouer(morpion,'O',x,y)
```



2. Occuper le centre

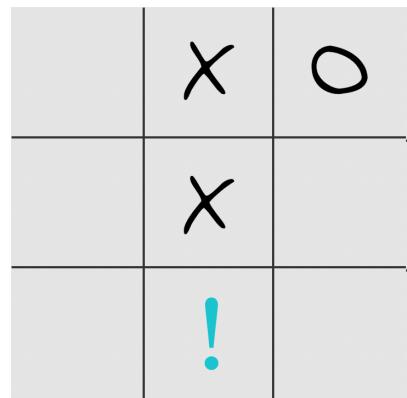
Si l'adversaire commence par un coin, prenez le centre. Cela vous donnera plus d'espace pour manœuvrer lors des prochains tours.

```
>>> jouer(morpion,'0',1,1)
```

3. Bloquez votre adversaire

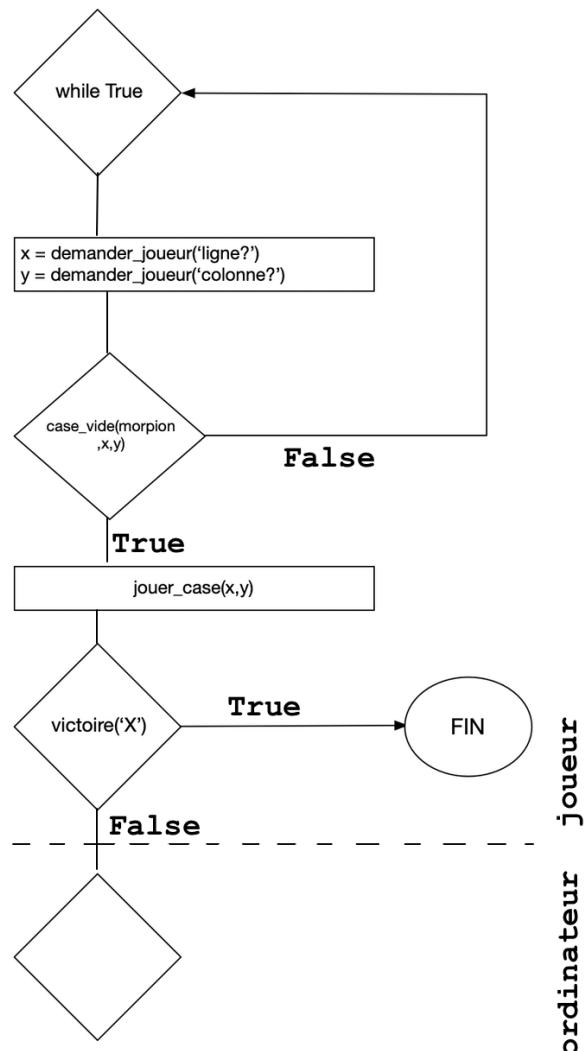
Soyez toujours prêt à vous défendre. Si votre adversaire place deux croix de suite, bloquez-les lors du prochain tour. En contrecarrant constamment ses tentatives de victoire, vous augmentez vos chances de tracer votre propre chemin vers la victoire.

```
>>> jeu_de_defense(... , ... ...)
```



> Compléter le tableau avec les instructions

demander au joueur le numero de la ligne à jouer
demander au joueur le numero de la colonne à jouer
questionner si la case du centre est vide
rechercher une case vide dans l'un des coins
placer le caractère 'O' en x,y
questionner si la case x,y est vide
rechercher les coordonnées pour bloquer le joueur 'X' s'il a 2 'X' alignés



> Continuer le diagramme décisionnel