

TPA2 : Tic tac toe

Le but de ce TP en autonomie est programmer le jeu du Tic tac toe : un joueur contre l'ordinateur.

Nous allons progresser pas à pas :



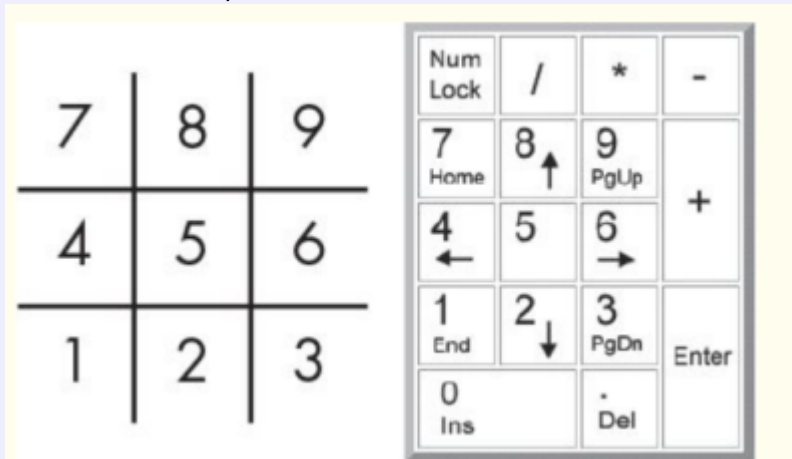
- tout d'abord : les fonctions de base
- puis la stratégie et les fonctions associées
- le programme principal



Le jeu est représenté dans le programme par une liste de 10 ' '. Les neuf cases du jeu se trouvent aux indices de 1 à 9, l'indice 0 ne sert pas.

Affichage

voilà ce qu'on doit afficher comme représentation du jeu:



* les cases sont numérotées comme sur le clavier: jeu[1] en bas à gauche ... jeu[9] en haut à droite



O



O



X



Fonctions de base

```
def trace_jeu(jeu):  
    '''cette fonction affiche le jeu passé en paramètre'''
```

```
def saisie_lettre_joueur():  
    '''saisie au clavier de la lettre choisie par le joueur 'X' ou 'O'  
    retourne une liste avec en premier la lettre du joueur puis en second la  
    lettre de l'ordinateur'''
```

```
def qui_commence():  
    ''' choisit par un random qui commence à jouer '''
```

```
def rejoue():  
    '''cette fonction retourne True si le joueur veut rejouer, False sinon'''
```

```
def joue(jeu, lettre, lieu):  
    '''met dans la liste jeu la lettre à la place lieu'''
```

```
def a_gagne(jeu, lettre):  
    '''retourne True si le joueur représenté par la lettre a gagné le jeu'''
```

```
def copieJeu(jeu):  
    '''retourne une copie de la liste jeu'''
```

```
def place_libre(jeu, place):  
    ''' retourne True si la place est libre sur le jeu'''
```

```
def saisie_coup_suivant(jeu):  
    ''' saisit le coup suivant du joueur'''
```

```
def choix_coup_dans_liste(jeu, liste_coups):  
    '''retourne un coup possible de la liste liste_coups (choisi au hasard si  
    plusieurs possibles)'''
```

```
def jeu_complet(jeu):  
    ''' retourne True s'il n'y a plus de place dans le jeu'''
```

Stratégie

Stratégie de l'ordinateur



- regarde d'abord si il y a un coup qui peut le faire gagner
- sinon si il y a un coup qui peut faire gagner le joueur alors il le bloque
- sinon, s'il y a un coin de libre, y joue
- sinon, si le centre est libre, y joue
- sinon joue sur un des côtés

```
def coup_ordinateur(jeu, lettre_ordinateur):  
    '''détermine quel coup doit jouer l'ordinateur'''
```

Programme principal

Algo général:

* **BOUCLE EXTERNE:**

- demande au joueur quelle lettre il veut
- décide qui commence
- **BOUCLE**
 - si c'est le joueur
 - montre le jeu
 - saisit le coup du joueur
 - place la lettre du joueur dans le jeu
 - regarde si le joueur a gagné : partie finie
 - regarde si le jeu est plein : match nul et partie finie
 - si ni l'un ni l'autre c'est au tour de l'ordinateur: on recommence la BOUCLE
 - sinon (ordinateur)
 - détermine quel coup doit jouer l'ordinateur
 - place la lettre de l'ordinateur dans le jeu
 - regarde si l'ordinateur a gagné partie finie
 - regarde si le jeu est plein : match nul et partie finie
 - si ni l'un ni l'autre c'est au tour du joueur: on recommence la BOUCLE
 - si partie finie: demande au joueur s'il veut faire une autre partie:
 - si oui on recommence la BOUCLE EXTERNE
 - sinon on s'arrête



From:

<https://wiki.centrale-med.fr/informatique/> - **WiKi informatique**

Permanent link:

https://wiki.centrale-med.fr/informatique/tc_info:tpa1

Last update: **2019/12/09 16:49**

