

# Programmation C

## TP 9 - Les fonctions

# I - Objectifs

Nous allons créer un **petit jeu de type Morpion**. L'objectif est d'aligner 3 symboles identiques dans une grille de 3X3. Le jeu se joue à deux. Chaque joueur possède un symbole parmi 'X' et 'O'. A tour de rôle, ils dessinent dans une case libre de la grille leur symbole respectif. Celui qui aligne 3 de ses symboles (horizontal, vertical ou diagonal), remporte la victoire. En cas de grille pleine sans alignement de 3 symboles identiques, c'est un match nul!

```
...
|X|O|X|
|X|O|O|
|. |X|O|
Joueur X ou voulez-vous jouer (x y)? 1 1
Case deja occupee!
Joueur X ou voulez-vous jouer (x y)? 2 0
Victoire du joueur X
|X|O|X|
|X|O|O|
|X|X|O|
```

## II - Préparatifs

1- Créer un nouveau projet C du nom de **"tp9\_fonctions"**.

2- Remplacer le code de la fonction main par:

```
printf("TP9: Les fonctions\n");

int fin_partie=0;
char joueur = 'X';
int coord_x, coord_y;
char grille[3][3]=
{
    {'.', '.', '.', '.'},
    {'.', '.', '.', '.'},
    {'.', '.', '.', '.'}
};

do{ } while(fin_partie != 1);

return 0;
```

# III - Instructions

- 1- Créer une fonction qui affiche le contenu de la grille de manière graphique comme dans l'exemple.
- 2- Créer une fonction pour gérer la saisie utilisateur:
  - 2.a- Affichage des instructions: "Joueur X ou voulez-vous jouer (x y)?"
  - 2.b- Lecture de la saisie (coordonnées x et y de la case à joueur)
  - 2.c- Inscription du signe du joueur dans la grille
- 3- Créer une fonction qui vérifie la validité de la saisie utilisateur ( case libre et coordonnées valide)
- 4- Utiliser cette dernière fonction pour redemander à l'utilisateur une saisie temps qu'elle n'est pas valide.
- 5- Utiliser ces fonctions dans la boucle de jeu de la fonction main

## IV - Pour aller plus loin

- A- Créer une fonction qui permet de savoir si le joueur courant a aligné 3 symboles (ligne, colonne ou diagonale)
- B- Créer une fonction qui permet de savoir si la grille est pleine (match null)
- C- Gérer les fins de parties via ces différentes fonctions

# V - SPOIL! (aide)

1- **void dessinerGrille(char grille[][3]):** Faire une boucle qui parcourt les lignes de la grille. Pour chaque lignes afficher les case en ajoutant le signe '|' comme séparateur

2- **void saisieUtilisateur(char grille[][3], int\* coord\_x, int\* coord\_y, char joueur)**

2.a- utiliser un printf et afficher le symbole du joueur: "Joueur X ou voulez-vous jouer (x y)?"

2.b- attention au fflush(stdin) et à l'utilisation des pointeurs

2.c- il faut utiliser le char de la variable 'joueur' pour inscrire le bon symbole dans la grille

3- **int saisieValide(char grille[][3], int coord\_x, int coord\_y):** il faut vérifier que les coordonnées sont bien comprises entre 0 et 2. De plus, il faut vérifier que la case ciblée ne contient pas déjà le symbole 'X' ou 'O'.

4- il faut ajouter à la fonction **saisieUtilisateur** une boucle do while et lui donner comme condition le retour de la fonction **saisieValide**

A- **int victoire(char grille[][3], char joueur):** Vérifier les lignes puis les colonnes puis enfin la diagonale et la diagonale inverse.

B- **int grillePleine(char grille[][3]) :** parcourir simplement la grille à la recherche d'une cases vide ('.')