

# LIFAP1 – TD 12 : Mémoire texte ... puis graphique

*Objectifs* : Terminer le codage du Mémoire version texte  
Commencer à traiter la version graphique.

## Version texte

1. Écrire en C/C++ une procédure demandant au joueur de choisir deux cases et d'afficher le contenu de ces deux cases en les restituant dans la grille complète.
2. Écrire en C/C++ une fonction de vérification du choix de l'utilisateur. Si les deux cases choisies sont identiques la fonction renverra 0 sinon elle renverra 1.
3. Simuler le jeu à deux joueurs jusqu'à ce que toutes les paires aient été trouvées.

## Version GrAPiC

Afin de faciliter le codage de l'application en TP sous GrAPiC, nous allons écrire ici quelques sous-programmes.

Dans la version graphique, vous allez utiliser des images plutôt que les mots de la version texte. Les fichiers d'images seront copiés dans le dossier *data* de Grapic et seront nommés de la manière suivante : `Image_Memory_XX.jpg` avec `XX` un entier compris entre 01 et 18.

Afin de pouvoir positionner aléatoirement les images dans la grille de jeu, il est impératif de pouvoir générer automatiquement le nom du fichier qui sera associé à l'image contenue dans la grille. Pour éviter les doublons (de paires de cartes) il faudra veiller à ce que l'indice choisi de manière aléatoire n'ait pas déjà été sélectionné. On utilisera pour cela un tableau `t_indice_image` tableau de `MAX_IMAGES` cases de type entier.

1. Écrire en C/C++ une fonction qui à partir du tableau `t_indice_image` choisit aléatoirement et retourne un indice non utilisé dans ce tableau.
2. Écrire en C/C++ une procédure qui à partir d'un entier passé en paramètre génère le nom du fichier image associé.
3. Écrire en C/C++ une procédure qui positionne dans la grille du mémoire deux fois l'image dont le nom a été créé précédemment et marque à 1 l'indice utilisé dans le tableau `t_indice_image`.