

# UE INFORMATIQUE

Programmation 1 : Fondamentaux

Laureen Fabre

# FICHIERS

- Pour lire et écrire dans un fichier il vous faudra la bibliothèque *stdlib*.
- Pour lire et écrire dans un fichier, il faut d'abord l'ouvrir.  
Pour cela il faudra utiliser la fonction *fopen*.

Cette fonction prend en paramètre :

- Le nom du fichier à ouvrir (doit être placé au même endroit que votre .exe)
- Le mode d'ouverture du fichier

# MODES D'OUVERTURE

- `"r"` : **lecture seule**. Vous pourrez lire le contenu du fichier, mais pas y écrire. Le fichier doit avoir été créé au préalable.
- `"w"` : **écriture seule**. Vous pourrez écrire dans le fichier, mais pas lire son contenu. Si le fichier n'existe pas, il sera créé.
- `"a"` : **mode d'ajout**. Vous écrirez dans le fichier, en partant de la fin du fichier. Vous ajouterez donc du texte à la fin du fichier. Si le fichier n'existe pas, il sera créé. Ce mode d'ajout peut être utile si vous voulez seulement ajouter des informations à la fin du fichier.
- `"a+"` : **ajout en lecture / écriture à la fin**. Vous écrivez et lisez du texte à partir de la fin du fichier. *Si le fichier n'existe pas, il sera créé.*
- `"r+"` : **lecture et écriture**. Vous pourrez lire et écrire dans le fichier. Le fichier doit avoir été créé au préalable.
- `"w+"` : **lecture et écriture, avec suppression du contenu au préalable**. Le fichier est donc d'abord vidé de son contenu, vous pouvez y écrire, et le lire ensuite. Si le fichier n'existe pas, il sera créé.

# FICHIERS

- La fonction **fopen** retourne un FILE, c'est lui que l'on manipulera :

```
FILE * fichier = fopen("test.exe", "r+");
```

- Le fichier ne s'est pas bien ouvert si *fichier* == *NULL*
- Pour fermer un fichier, il faudra utiliser **fclose**.  
La fonction prend en paramètre votre FILE :

```
fclose(fichier) ;
```

# ÉCRITURE

- Plusieurs fonctions pour écrire dans un fichier :

- **fputc** écrit un seul caractère :

*int fputc(int caractere, FILE\* fichier);*

Exemple :

*fputc('A',fichier);*



Les fonctions « put » renvoient EOF si il y a une erreur

- **fputs** écrit une chaîne de caractères :

*char\* fputs(const char\* chaine, FILE\* fichier);*

Exemple :

*fputs("Salut les CIR1\n", fichier);*

- **fprintf** écrit une chaîne formatée :

Exemple :

*int nb;*

*scanf("%d", &nb);*

*fprintf(fichier, "il a %d euros", nb);*

# LECTURE

- Plusieurs fonctions pour lire dans un fichier :

- **fgetc** lit un caractère :

*int fgetc(FILE\* fichier);*

Exemple :

    int charCourant = fgetc(fichier);

Pour tout lire il faut boucler TANT QUE *charCourant* != EOF

- **fgets** lit une chaîne de caractères jusqu'au premier retour à la ligne :

*char\* fgets(char\* chaine, int nbreDeCaracteresALire, FILE\* pointeurSurFichier);*

Exemple :

    char chaine[200] = "":

    fgets(chaine, 200, fichier);

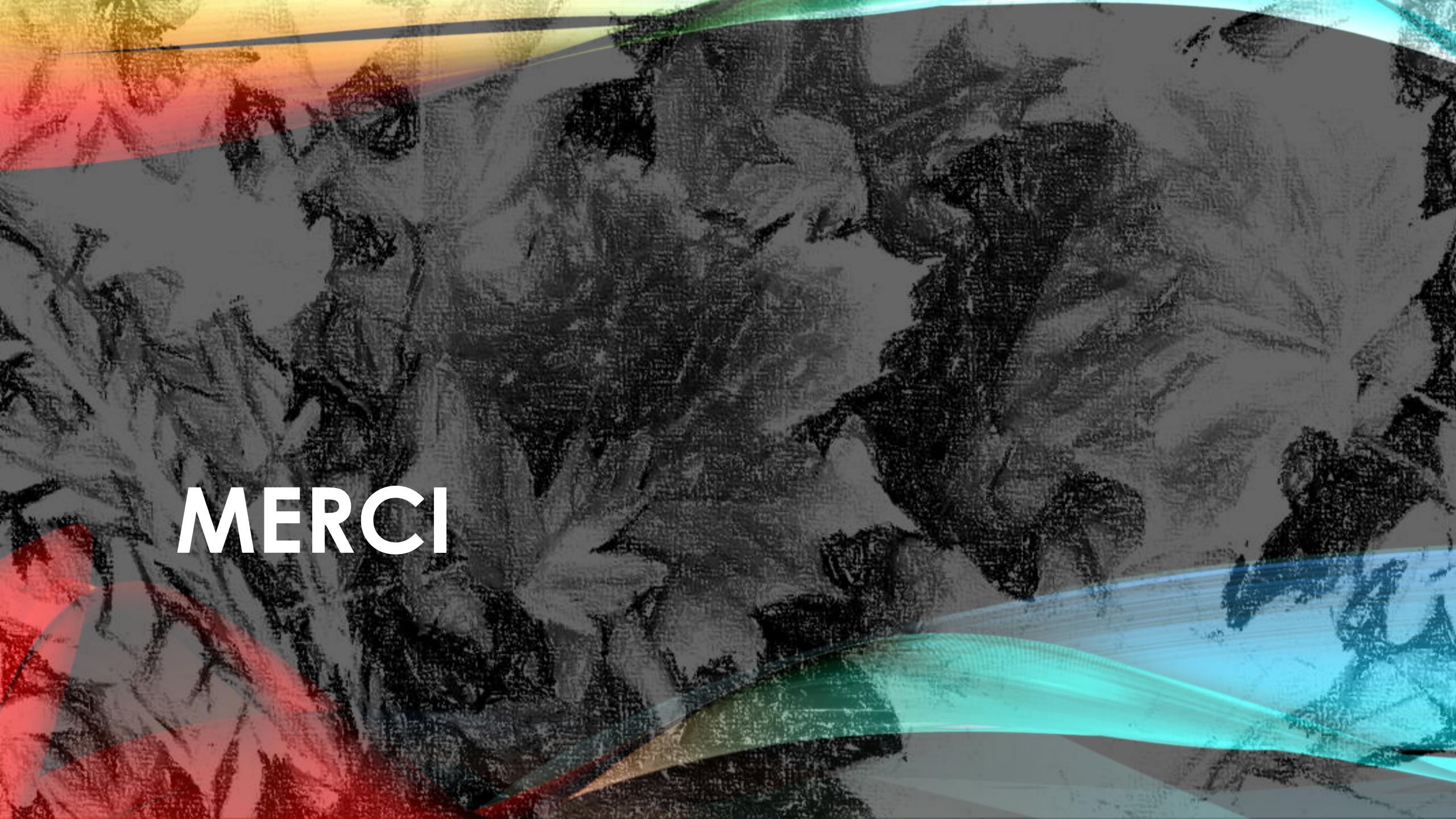
Pour tout lire il faut boucler TANT QUE *fgets(...)* != NULL

- **fscanf** lit une chaîne formatée :

Exemple :

    int score[3] = {0};

    fscanf(fichier, "%d %d %d", &score[0], &score[1], &score[2]);



MERCI