

IN104 - Solveur de Sudoku - TP2

Ugo Vollhardt

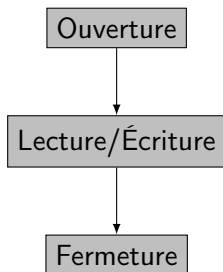
CEA LIST

17 avril 2018

Déroulement prévisionnel :

1. Rappels sur Git, mise en place versionnement, structure donnée et premières fonctions de mises à jours de possibilités.
2. Rappels/introduction à la notion de récursivité, poursuite du travail de la première séance et mise en place des premiers algo de résolution.
3. Fin d'implémentation des premiers algo de résolution + rappels sur sujet au choix.
4. Introduction à la librairie GTK+ pour Python et C, implémentation d'une interface minimale.
5. Séance en autonomie.
6. Revu des points précédents et approfondissement si besoin.

Utilisation d'un fichier



1. **Ouverture** : Bloque l'accès au fichier. Crée un "canal" pour interagir directement avec son contenu.
2. **Lecture/Écriture** : A travers le "canal" modification ou non le contenu du fichier.
3. **Fermeture** : "Libère" le fichier. Si le fichier n'est pas fermé, il restera bloqué !

Ouverture d'un fichier

- **C :**
`FILE* fopen(const char* nomDuFichier, const char* modeOuverture)`
- **Python :**
`file = open(name[, mode])`
- **MatLab :**
`file = fopen(filename,permission)`

Mode d'ouverture

- **r** : Lecture seule, commence au début du fichier
- **r+** : Lecture + Écriture, commence au début du fichier
- **w** : Écriture seule, vide le fichier avant d'écrire ou crée un nouveau fichier
- **w+** : Écriture + Lecture, vide le fichier avant d'écrire ou crée un nouveau fichier
- **a** : Écriture seule, écrit à la suite du contenu du fichier ou crée un nouveau fichier
- **a+** : Écriture + Lecture, écrit à la suite du contenu du fichier ou crée un nouveau fichier

Écriture dans un fichier

- **C :**

`int fputc(int caractere, FILE* pointeurSurFichier)` : Écrit un seul caractère

`int fputs(const char* string, FILE* pointeurSurFichier)` : Écrit une chaîne de caractère

`int fprintf(FILE* pointeurSurFichier,...)` : Même chose que `printf`

- **Python :**

`file.write(str)` : Écrit une String dans le fichier

`file.writelines(sequence)` : Écrit une séquence de String dans le fichier

- **MatLab :**

`fwrite(file,A)` : Écrit le contenu de A

Lecture d'un fichier

- **C :**

`int fgetc(FILE* pointeurDeFichier)` : Lit un seul caractère

`char* fgets(char* chaine, int nbChar, FILE* pointeurSurFichier)` : Lit `nbChar` et les place dans le tableau "chaine"

`fscanf(FILE* pointeurDeFichier,...)` : Même chose que `scanf`

- **Python :**

`file.read([size])` : Lit "size" bits du fichier et les retourne sous la forme d'une string

`file.readline()` : Lit une ligne complète et l'enregistre dans une string

`file.readlines()` : Lit toutes les lignes et les enregistre dans une liste de string

- **MatLab :**

`A = fread(fileID,sizeA)` : Lit `sizeA` données et les place dans `A`

`A = fscanf(fileID,formatSpec)` : Même chose que `scanf` en C

fermeture d'un fichier

- **C :**
`int fclose(FILE* pointeurSurFichier)`
- **Python :**
`file.close()`
- **MatLab :**
`fclose(file)`

Objectifs de la séance

1. finir les objectifs de la séance précédente
2. implémenter la boucle de résolution simple (mise à jours des possibilités constamment jusqu'à résolution de la grille ou impasse)
3. Implémenter une fonction récursive pour les cas où la résolution tombe dans une impasse