

TD Machine #5 (xxième séance de TD) de CS101

- 14 dec 2022. Durée 1h30
- Pas de compte-rendu. Il est fortement recommandé de prendre des notes personnelles.
- Version 2022.1
- *author* Laure Gonnord

Préparation

Prévoir une clé USB pour récupérer vos travaux.

À vos marques

- Le TDM est réalisé sur les machines de TP Linux de la salle B141. **utilisez la VM "Laure Gonnord-compilation"**.
- Essayer le plus possible de faire seul·e (on apprend moins vite en binôme)
- Ne pas hésiter à demander de l'aide (on apprend!)
- Créer un répertoire pour les TP de C nommé CS101, et travailler dans un sous répertoire nommé TDM5.
- Note : vous pouvez utiliser cloud.esisar.grenoble-inp.fr pour sauvegarder vos programmes.
en l'absence de réseau: prévoir une clé USB

Prêts

- Recopier dans un fichier nommé tdm5.c le programme suivant:

```
#include <stdio.h>
#include <stdlib.h>

void mange_et_reecritl(void)
{
    char c;
    c = getchar();
    while ( c != EOF )
    {
        putchar(c);
        c = getchar();
    }
}

int main(void)
{
    mange_et_reecritl();
    return 0;
}
```

- Prévoir le comportement de ce programme.
- Compiler, tester. (Noter que le programme attend des caractères entrés par l'utilisateur·rice, et que EOF peut être obtenu en tapant CONTROL-D)

Entrées-sorties et redirections

Dans le manuel de votre terminal (bash), on peut trouver une explication des redirections: (cf <http://manpagesfr.free.fr/man/man1/bash.1.html> en français), donnée simplifiée ci-dessous:

Redirection d'entrée

Lorsque l'on applique une redirection d'entrée, le fichier dont le nom résulte du développement du mot sera ouvert en lecture [...] en tant qu'entrée standard (l'entrée standard est "le clavier").

Le format général des redirections d'entrée est le suivant : <mot

- Déduire le comportement d'une redirection de sortie ...

Application

- Prévoir le comportement de `./tdm5 > foo.txt`. Exécuter, et regarder (on peut utiliser la commande `cat nomdunfichier` pour regarder le contenu d'un fichier texte sans l'éditer, dans un terminal).
- En utilisant un éditeur, créer un fichier `bar.txt` contenant des chaînes de caractères:

```
truc  
bidule  
plouf
```

- Prévoir le comportement de `./tdm5 <bar.txt >foo2.txt`. Comparer avec la commande `cp` Unix.

Écriture de fonctions de chaînes.

Ne pas oublier de tester correctement, à l'aide de l'entrée standard ou de redirections avec des fichiers judicieusement choisis (le mot texte est alors soit l'entrée standard, soit le fichier lu par l'intermédiaire d'une redirection).

- Sur le même modèle, écrire une fonction qui compte le nombre de caractères lus, et comparer à la commande Unix `wc`.
- Écrire une fonction qui à la fin de la lecture imprime les lettres latines jamais lues dans le fichier (on pourra utiliser un tableau de taille 26 pour "se rappeler des caractères déjà vus").
- Écrire une fonction qui réimprime le texte à l'envers (on pourra se demander comment mémoriser les caractères lus).
- Écrire une fonction qui recherche une chaîne dans le texte.
- Écrire une fonction qui dit si les parenthèses du texte sont bien imbriquées (en ignorant les autres caractères).

Bonus

- Coder le tri fusion.