

[Mon accueil](#) / [Mes cours](#) / [Enseignement et formations](#) / [Département d'Informatique](#) / [L3 informatique](#) / [2023-2024](#)  
 / [INFO3203 - Architecture & Système 2023-2024](#) / TDs et TPs / [TD 2 - C](#)

## TD 2 - C

### TRAVAU DIRIGÉ 2

#### DESCRIPTION

Ce TD a pour but de vous donner un peu de temps pour vous entraîner avec le C avant que nous commençons à creuser dans des programmes plus orientés OS :

#### C PROGRAMMES ET COMPILATION

Effectuez les tâches suivantes en utilisant

- un éditeur de votre choix
- le compilateur GDB, si vous avez des problèmes.

#### Requis

Écrivez un programme qui lit une série de chaînes de caractères à partir du clavier, les stocke dans un tableau interne, trie le tableau dans l'ordre croissant et l'imprime. Le programme doit utiliser la mémoire dynamique plutôt que des vecteurs surdimensionnés. Le tri des chaînes de caractères est induit par la fonction de la bibliothèque `strcmp()` : étant donné deux chaînes de caractères `s1`, `s2`, nous considérons que

- `s1 < s2` si `strcmp(s1, s2) < 0` ;
- `s1 > s2` si `strcmp(s1, s2) > 0` ;
- `s1 = s2` si `strcmp(s1, s2) = 0`.

L'utilisateur communique le nombre de chaînes à lire lors de la première opération. Une séquence typique d'I/O de programme est la suivante :

```

Nombre de chaînes de caractères à trier : 3
Chaîne.1 : Tom
Chaîne.2 : Chris
Chaîne.3 : David
  Vecteur trié :
  Chris
  David
  Tom
  
```

#### Piste de mise en œuvre

Le fichier doit être divisé en deux modules (fichiers sources) : un fichier `main.c` contenant uniquement le principal et un fichier `util.c` contenant les fonctions implémentées. Il est également nécessaire de créer un fichier `defs.h` contenant les prototypes des fonctions implémentées dans `util.c`.

Mettre en œuvre les fonctions suivantes :

`char **read_strings(int n)` Lit `n` chaînes de caractères à partir du clavier, les copie dans une structure allouée en mémoire dynamique et renvoie le pointeur vers cette structure. En détail : `read_strings(n)` alloue dans la mémoire dynamique, en utilisant `malloc()`, un vecteur de `n` pointeurs de caractères - c'est-à-dire un vecteur de `n` éléments, chacun d'entre eux étant de type `char *`. Soit `p` le pointeur vers ce vecteur ; pour `i=0, . . . , n-1`, lit une chaîne de caractères à partir du clavier, crée un double dans la mémoire dynamique en utilisant la fonction de bibliothèque `strdup()`, et stocke le pointeur vers le double dans `p[i]`. Retourne `p`, ou `NULL`.

*Note.* La structure de données souhaitée est du type illustré à la figure 1.

`void selsort(char **begin, char **end)` trie par sélection le vecteur chaîne (`=char *`) entre début et fin.

`char **pos_min(char **begin, char **end)` renvoie le pointeur vers la chaîne "minimale" (au sens défini par `strcmp()`) dans le sous-vecteur `begin . . . fin`. Il est utilisé par `selsort()`.

`void swap(char **a, char **b)` échange `*a` et `*b`. Il est utilisé par `selsort()`. Notez bien comment s'effectue cette "permutation de chaînes".

`void print_strings(char **t, int n)` imprime les chaînes de caractères (pointées par) `t[0], . . . , t[n-1]`.

`void delete(char **p, int n)` libère au moyen de `free()` la mémoire associée aux pointeurs `p[0], . . . , p[n-1]`, et enfin celle associée à `p`.

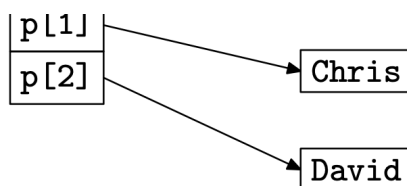


Fig.1 -Exemple de vecteur de chaînes de caractères

## Programme principal.

Utilisez le fichier `main.c` suivant :

```
#include <stdlib.h>
#include <stdio.h>
#include "defs.h" // Pour les prototypes de fonctions implémentées
int main(void)
{
    char **v;
    int n;
    printf("Nombre de chaînes de caractères à trier : ");
    scanf("%d", &n);
    v=read_strings(n);
    if( v!=NULL ) {
        selsort(v,v+n-1);
        printf( "\nVecteur trié:\n" );
        print_strings(v,n);
        delete(v,n);
    } else printf( "Erreur." );
}
```

L'inclusion `#include "defs.h"` permet d'inclure le contenu du fichier `defs.h`, c'est-à-dire les types de fonctions définis dans le module `util.c`, dans le module `main.c`, assurant ainsi leur visibilité. *Ecrivez un makefile pour compiler votre programme avec succès.*

Modifié le: mercredi 7 février 2024, 16:19

[◀ sometext.txt](#)

Aller à...

[TD 1 ▶](#)

## ADMINISTRATION

> Administration du cours

### PRATIQUE

Connecté sous le nom « [Augustin Lucas](#) » ([Déconnexion](#))  
[INFO3203 - Architecture & Système 2023-2024](#)

[Enseignement et Formations](#)

[Étudiants](#)

[Enseignants et Personnels](#)

[Obtenir l'app mobile](#)

[Politiques](#)

### BESOIN D'AIDE ?

Vous êtes étudiant ? enseignant ?  
Vous avez une question concernant le portail ?

[Écrivez à l'assistance](#)