

Runtrack C : Jour 2

Job 01

Vous devez implémenter la fonction `divide` (dans un fichier `divide.c`) qui va prendre en paramètre un pointeur sur `int`, et qui devra diviser la valeur de l'`int` par 2.




```
void divide(int *val);
```

Fonctions autorisées : **aucune**.

Job 02

Vous devez implémenter la fonction **`swap`** (dans un fichier **`swap.c`**) qui prend en paramètres deux pointeurs sur `int`, et qui doit échanger leurs valeurs.



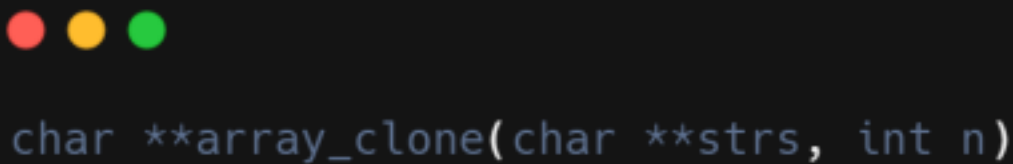
```
void swap(int *a, int *b);
```

Fonctions autorisées : **aucune**.



Job 03

Vous devez implémenter la fonction **array_clone** (dans un fichier **array_clone.c**) qui doit prendre en paramètre un tableau de chaînes de caractères, et un int n. Elle devra renvoyer un tableau de chaîne de caractères qui contient des copies des chaînes de caractères du tableau entrant, jusqu'à la nième, ou jusqu'au premier pointeur NULL présent. Le tableau sortant devra se terminer par un pointeur NULL.



```
char **array_clone(char **strs, int n)
```

Fonctions autorisées : **malloc**.

Job 04

Vous devez implémenter la fonction **sort** (dans un fichier **sort.c**) qui prend en paramètre un tableau de chaînes de caractères (se terminant par un pointeur NULL) et qui devra les trier dans l'ordre ASCII croissant. Un tri à bulles est suffisant.



```
char **sort(char **strs);
```

Fonctions autorisées : **aucune**.

Job 05

Vous devez implémenter une fonction **func_to_array** (dans un fichier



func_to_array.c), qui prend en paramètres un tableau de chaînes de caractères, ainsi qu'un pointeur sur fonction, et applique cette fonction sur toutes les chaînes.

Les fonctions qui peuvent être appliquées sont variées : une fonction `strtoupper`, `strtolower`, ou même `my_putstr`.

```
void func_to_array(char **strs, void (*fun)(char*))
```

Fonctions autorisées : **aucune**.

Job 06

Vous devez implémenter une fonction **`delete_str_array`** (dans un fichier **`delete_str_array.c`**) qui prend en paramètre un pointeur sur tableau de chaînes de caractères, qui doit libérer toutes les chaînes du tableau, le tableau lui même, et mettre son pointeur à NULL.

```
void delete_str_array(char ***strs);
```

Fonctions autorisées : **free**.

Job 07

Vous devez créer un programme (**dans un ou plusieurs fichier(s) .c**) qui prend en arguments des chaînes de caractères, les fait passer en minuscules, et les écrit sur la sortie standard, une par ligne.



```
./job7.exe "HELLO" "world" 123
```

```
hello  
world  
123
```

Fonctions autorisées : **malloc**, **free**, **write**.

Job 08

Vous devez créer un programme (**dans un ou plusieurs fichier(s) .c**) qui prend en arguments des chaînes de caractères, les trim, les trie dans l'ordre croissant puis les affiche en une seule ligne, avec un seul espace séparant chaque mots, et en terminant par un retour à la ligne.



```
./job8.exe "world" "HELLO"  
HELLO world
```

Fonctions autorisées : **malloc**, **free**, **write**.

Rendu



Le projet devra être rendu sur votre **github**, dans un repository nommé **runtrack_c**. Le repo doit contenir un dossier pour chaque jour de la runtrack, nommés "Jour01", "Jour02", "Jour03", "Jour04", "Jour05".

Chacun de ces dossiers devra contenir les jobs dans le dossier respectif :
"Job01", "Job02", etc.

Compétences visées

- C

Base de connaissances

- https://www.w3schools.com/c/c_getstarted.php
- <https://openclassrooms.com/fr/courses/19980-apprenez-a-programmer-en-c>
- <https://www.my-mooc.com/fr/mooc/c-programming-getting-started>

