



Piscine C++ - d03

My String

Sullivan “Sully” Veres veres_s@epitech.eu

Abstract: Ce document est le sujet du d03

Table des matières

I	Remarques générales	2
II	Exercice 0	3
III	Exercice 1	4
IV	Exercice 2	5
V	Exercice 3	6
VI	Exercice 4	7
VII	Exercice 5	8
VIII	Exercice 6	9
IX	Exercice 7	10
X	Exercice 8	11
XI	Exercice 9	12
XII	Exercice 10	13
XIII	Exercice 11	14
XIV	Exercice 12	15
XV	Exercice 13	16
XVI	Exercice 14	17
XVII	Exercice 15	18
XVIII	Exercice 16	19

Chapitre I

Remarques générales


- Si vous faites la moitié des exercices car vous avez du mal, c'est normal. Par contre, si vous faites la moitié des exercices par flemme et vous tirez à 14h, vous AUREZ des surprises. Ne tentez pas le diable.
- Les noms de fichiers qui vous sont imposés doivent être respectés A LA LETTRE, de même que les noms de fonctions.
- Les répertoires de rendus sont ex00, ex01, ...
- Lisez attentivement les exemples, ils peuvent requérir des choses que le sujet ne dit pas...
- Lisez ENTièrement le sujet d'un exercice avant de le commencer !
- RÉFLECHISSEZ. Par pitié.
- Notez bien qu'aucun de vos fichiers ne doit contenir de fonction "main" sauf si le contraire est explicite. Nous utiliserons notre propre fonction "main" pour compiler et tester votre code.



Mettez ce sujet à jour très régulièrement car une rumeur dit que de nouveaux exercices pourraient apparaître jusqu'à 4h avant le rendu...
Vous êtes prévenus !

Chapitre II

Exercice 0


	Exercice : 00	points : 1
My_String		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex00		
Compilateur : gcc	Flags de compilation: -Wall -Wextra -Werror	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Créer un module String. Ce module devra posséder :

- Un membre `char * str`
- Une fonction d'initialisation :
`void StringInit(String* this, char const * s);`
Cette fonction assigne au membre `str` la valeur de `s`
- Une fonction de destruction :
`void StringDestroy(String* this);` Cette fonction doit servir à détruire proprement une instance d'un module `String`.

Chapitre III

Exercice 1

	Exercice : 01	points : 1
Assign		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex01		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Vous devez faire deux fonctions membres assign :

- `void assign_s (String *this, String const * str);`
le contenu de la string est égal au contenu de la string passée en paramètre
- `void assign_c (String *this, char const * s);`
le contenu de la string est égal au contenu du `char*` passé en paramètre




Rappel : Je vous rapelle que les fonctions membres ne doivent pouvoir être appellées qu'à partir d'une instance de `String` .



- Pensez à affecter vos pointeurs sur fonction.
- Attention à ne pas laisser de fuite mémoire =

Chapitre IV

Exercice 2


	Exercice : 02	points : 1
Append		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex02		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Vous devez faire deux fonctions membres `append` :

- `void append_s(String* this, String const* ap);`
Copie le contenu de la `String` 'ap' passée en paramètre à la fin de la `String` .
- `void append_c(String* this, char const* ap);`
Copie le contenu du `char*` 'ap' passé en paramètre à la fin de la `String` .

Chapitre V

Exercice 3


	Exercice : 03	points : 1
At		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex03		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Vous devez faire une fonction membre `at` :

- `char at(String* this, size_t pos);`
Renvoie le `char` qui est à la position '`pos`' dans notre `String` . Si la position est incorrecte, renvoie -1

Chapitre VI

Exercice 4

	Exercice : 04	points : 1
Clear		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex04		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Vous devez écrire une fonction membre `clear` :


- `void clear(String* this);`
Vide le contenu de la `String`



Attention a votre pointeur.

Chapitre VII

Exercice 5


	Exercice : 05	points : 1
Size		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex05		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Vous devez écrire une fonction membre `size`

- `int size(String* this);`
Retourne la taille de la chaîne.
Si le pointeur de la chaîne est `NULL` , retourne -1

Chapitre VIII

Exercice 6


	Exercice : 06	points : 1
Compare		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex06		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Vous devez faire deux fonctions membres `compare` :

- `int compare_s(String *this, const String* str);`
Compare le contenu de la `String` avec la `String` passée en paramètre.
Les résultats sont équivalents à la fonction `strcmp` de la libc.
- `int compare_c(String *this, char const* str);` Compare le contenu de la `String` avec le `char*` passé en paramètre.
Les résultats sont équivalents à la fonction `strcmp` de la libc.

Chapitre IX

Exercice 7


	Exercice : 07	points : 1
Copy		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex07		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Vous devez faire une fonction membre `copy` :

- `size_t copy(String* this, char* s, size_t n, size_t pos);`
Copie dans `'s'` , `'n'` caractères de la `String` à partir de la position `'pos'` .
Elle retourne le nombre de caractères qui ont été copiés.

Chapitre X

Exercice 8


	Exercice : 08	points : 1
c_str		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex08		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Vous devez faire une fonction membre `c_str` :

- `char const* c_str (String * this);`
Renvoie le buffer contenu dans la `String` .

Chapitre XI

Exercice 9


	Exercice : 09	points : 1
empty		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex09		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Vous devez faire une fonction membre `empty` :

- `int empty (String* this);`
Retourne 1 si la chaîne est vide, -1 sinon.

Chapitre XII

Exercice 10

	Exercice : 10	points : 1
Find		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex10		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		


Vous devez faire deux fonctions membres `find` :

- `int find_s (String* this, const String *str, size_t pos);`
Cherche la première occurrence de la `String str` dans notre string à partir de la position `pos` .
- `int find_c (String* this, char const* str, size_t pos);`
Cherche la première occurrence du `char* str` dans notre string à partir de la position `pos` .

Renvoie la position à laquelle l'occurrence de `str` a été trouvée. -1 si la chaîne n'a pas été trouvée. Si la chaîne est trop grande, renvoyez -1. Si la position est invalide, renvoyez -1.

Chapitre XIII

Exercice 11

	Exercice : 11	points : 1
Insert		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex11		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Vous devez implémenter deux fonctions membres `insert` :

- `void insert_c(String * this, size_t pos, char const* str);`
Copie le contenu de `str` dans la `string`, à la position `pos`.
- `void insert_s(String * this, size_t pos, String const* str);`
Copie le contenu de `str` dans la `string`, à la position `pos`.


Ces fonctions agrandissent la taille de la chaîne. Si '`pos`' est supérieure à la taille de la chaîne, alors vous faites une insertion à la fin de la chaîne.



Attention au '`\0`'.

Chapitre XIV

Exercice 12


	Exercice : 12	points : 1
to_int		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex12		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Vous devez faire une fonction membre `to_int` :

- `int to_int(String * this);`
Transforme la `String` en `int` Cette fonction aura le même comportement que la fonction `atoi(3)` .

Chapitre XV

Exercice 13


	Exercice : 13	points : 2
Split		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex13		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Vous devez faire deux fonctions membres `split` :

- `String* split_s(String* this, char separator);` Retourne un tableau de string qui correspond à la string découpée grace au délimiteur `'separator'` .
- `char** split_c(String* this, char separator);` Retourne un tableau de chaine de caractères qui correspond à la string découpée grâce au délimiteur `'separator'` .

Chapitre XVI

Exercice 14

	Exercice : 14	points : 5
Aff		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex14		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Vous devez faire une fonction membre `aff` (lolilol) :

- `void aff(String * this);` Cette fonction affiche le contenu de la `string` sur la sortie standard.



Attention, j'ai pas dit de mettre un retour à la ligne !




Attention bis, donc `printf` n'est pas forcément une bonne idee !

Remarque : Et oui, c'est la fonction qui vaut le plus de points :)

Chapitre XVII

Exercice 15


	Exercice : 15	points : 2
Join		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex15		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Vous devez ecrire les fonctions membres `join_c` et `join_s` :

- `void join_c(String* this, char delim, char const** tab);`
 Cette fonction membre assignera à la `String` une chaîne de caractères composée de toutes les chaînes de caractères du tableau `tab`, séparées par le délimiteur `delim`. Le tableau sera toujours `NULL terminated`
- `void join_s(String* this, char delim, String* tab);` Cette fonction membre assignera à la `String` une chaine de caractères composée de toutes les `String` du tableau `tab`, séparées par le délimiteur `delim`. Le tableau sera toujours terminé par une `String` vide.

Chapitre XVIII

Exercice 16

	Exercice : 16	points : 3
Substr		
Répertoire de rendu: (DÉPOT SVN - piscine_cpp_d03-promo-login_x)/ex16		
Compilateur : gcc	Flags de compilation: -Wextra -Werror -Wall	
Makefile : Non	Règles : n/a	
Fichiers a rendre : String.h, String.c		
Remarques : n/a		
Fonctions Interdites : Aucune		

Vous devez écrire la fonction membre `substr` :

- `String* substr(String *this, int offset, int length);` Extrait une sous chaîne commençant à `offset` et de longueur `length` et la retourne sous la forme d'une nouvelle instance de `String`. Si `offset` est négatif, il représente le nombre de caractères en partant de la fin. Si `length` est négatif, il représente le nombre de caractères à copiée à gauche de l'offset. Si la sous chaîne spécifiée est en partie en dehors de `String`, on ne retourne que la partie présente dans `String`.