



C++ Pool - d03

My String

Koalab koala@epitech.eu

Abstract: This document is the subject of d03.

Contents

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

Chapter I

GENERAL RULES


- If you only do the half of the exercises because you're struggling, that's normal. But, if you do the half of the exercises because you're lazy and leave at 2pm, you WILL have surprises. Do not play with fire.
- File names that are required must be respected to the letter, same for function names.
- Turn-in directories are ex00, ex01, ...
- Read the examples CAREFULLY. They might require things the subject doesn't say...
- Read ENTIRELY the subject of an exercise before you start it !
- THINK. Please.
- Note that none of your files must contain a `main` function, unless the contrary is explicitly specified. We will use our own `main` function to compile and test your code.



Update this subject very regularly because a rumor says that new exercises could appear up to 4 hours before turn-in time... You are warned!

Chapter II

Exercise 0

	Exercise : 00	points : 1
My_String		
Turn-in directory: (piscine_cpp_d03)/ex00		
Compiler: gcc	Compilation flags: -Wall -Wextra -Werror	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

Create a `String` module. This module must have:

- A `char * str` member
- An initialization function:

```
void StringInit(String* this, char const * s);
```


This function assigns to the `str` member the value of `s`.
- A destructor function:

```
void StringDestroy(String* this);
```

This function must be used to properly destroy a `String` module instance.

Chapter III

Exercise 1

	Exercise : 01	points : 1
Assign		
Turn-in directory: (piscine_cpp_d03)/ex01		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

You have to add two `assign` member functions:

- `void assign_s (String *this, String const * str);`
The content of the `String` is equal to the content of the `String` passed as parameter.
- `void assign_c (String *this, char const * s);`
The content of the `String` is equal to the content of the `char*` passed as parameter.




Reminder: I remind you that member functions can only be called from a `String` instance.



- Think to affect your function pointers.
- Be careful not to leave any memory leaks.

Chapter IV

Exercise 2


	Exercise : 02	points : 1
Append		
Turn-in directory: (piscine_cpp_d03)/ex02		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

You have to add two `append` member function:

- `void append_s(String* this, String const* ap);`
Copies the content of the `String ap` given as parameter at the end of the `String` .
- `void append_c(String* this, char const* ap);`
Copies the content of the `char* ap` given as parameter at the end of the `String` .

Chapter V

Exercise 3


	Exercise : 03	points : 1
At		
Turn-in directory: (piscine_cpp_d03)/ex03		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

You have to add an `at` member function:

- `char at(String* this, size_t pos);`
Returns the `char` which is at the position `pos` in our `String` . If the position is incorrect, returns -1.

Chapter VI

Exercise 4

	Exercise : 04	points : 1
Clear		
Turn-in directory: (piscine_cpp_d03)/ex04		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

You have to add a `clear` member function:


- `void clear(String* this);`
Empties the content of the `String` .



Be careful with your pointer.

Chapter VII

Exercise 5


	Exercise : 05	points : 1
Size		
Turn-in directory: (piscine_cpp_d03)/ex05		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

You have to add a `size` member function:

- `int size(String* this);`
Returns the size of the string.
If the string pointer is `NULL` , returns -1.

Chapter VIII

Exercise 6


	Exercise : 06	points : 1
Compare		
Turn-in directory: (piscine_cpp_d03)/ex06		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

You have to add two `compare` member functions:

- `int compare_s(String *this, const String* str);`
Compares the content of the `String` with the `String str` given as parameter.
Results are the same as the `strcmp` libc function.
- `int compare_c(String *this, char const* str);`
Compares the content of the `String` with the `char* str` given as parameter.
Results are the same as the `strcmp` libc function.

Chapter IX

Exercise 7


	Exercise : 07	points : 1
Copy		
Turn-in directory: (piscine_cpp_d03)/ex07		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

You have to add a `copy` member function:

- `size_t copy(String* this, char* s, size_t n, size_t pos);`
Copies in 's' , 'n' characters of the `String` from the position `pos` . Returns the number of characters which have been copied.

Chapter X

Exercise 8


	Exercise : 08	points : 1
c_str		
Turn-in directory: (piscine_cpp_d03)/ex08		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

You have to add a `c_str` member function:

- `char const* c_str (String * this);`
Returns the buffer contained in the `String` .

Chapter XI

Exercise 9


	Exercise : 09	points : 1
empty		
Turn-in directory: (piscine_cpp_d03)/ex09		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

You have to add an `empty` member function:

- `int empty (String* this);`
Returns 1 if the string is empty, -1 otherwise.

Chapter XII

Exercise 10

	Exercise : 10	points : 1
Find		
Turn-in directory: (piscine_cpp_d03)/ex10		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		


You have to add two `find` member functions:

- `int find_s (String* this, const String *str, size_t pos);`
Searches the first occurrence of the `String str` in our string starting from the position `pos` .
- `int find_c (String* this, char const* str, size_t pos);`
Searches the first occurrence of the `char* str` in our string starting from the position `pos` .

Returns the position at which the occurrence of `str` has been found. -1 if the string has not been found. If the string is too long, return -1. If the position is invalid, return -1.

Chapter XIII

Exercise 11

	Exercise : 11	points : 1
Insert		
Turn-in directory: (piscine_cpp_d03)/ex11		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

You have to add two `insert` member functions:

- `void insert_c(String * this, size_t pos, char const* str);`
Copies the content of `str` in the `String`, at the position `pos`.
- `void insert_s(String * this, size_t pos, String const* str);`
Copies the content of `str` in the `String`, at the position `pos`.


These functions enlarge the size of the string. If `pos` is greater than the size of the string, then you make an insertion at the end of the string.



Be careful with the `'\0'`.

Chapter XIV

Exercise 12


	Exercise : 12	points : 1
to_int		
Turn-in directory: (piscine_cpp_d03)/ex12		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

You have to add a `to_int` member function:

- `int to_int(String * this);`
Transforms the `String` into an `int` . This function will have the same behavior as the `atoi(3)` function.

Chapter XV

Exercise 13


	Exercise : 13	points : 2
Split		
Turn-in directory: (piscine_cpp_d03)/ex13		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

You have to add two `split` member function:

- `String* split_s(String* this, char separator);`
Returns a table of string corresponding to the string splitted by the delimiter 'separator' .
- `char** split_c(String* this, char separator);`
Returns a table of characters string corresponding to the string splitted by the delimiter 'separator' .

Chapter XVI

Exercise 14

	Exercise : 14	points : 5
Aff		
Turn-in directory: (piscine_cpp_d03)/ex14		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

You have to add an `aff` member function (lol):

- `void aff(String * this);`
This function displays the content of the `String` on the standard output.



Be careful, I never talked about a carriage return!




Be careful one more time, `printf` is NOT necessarily a good idea!

Remark: Yes, this function is worth the most points. :)

Chapter XVII

Exercise 15


	Exercise : 15	points : 2
Join		
Turn-in directory: (piscine_cpp_d03)/ex15		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

You have to add two `join` member functions:

- `void join_c(String* this, char delim, char const** tab);`
This member function will assign to the `String` a string of characters composed by all strings of characters of the table `tab` , separated by the delimiter `delim` . The table will always be `NULL` terminated .
- `void join_s(String* this, char delim, String* tab);`
This member function will assign to the `String` a string of characters composed by all `String` of the table `tab` , separated by the delimiter `delim` . The table will always be terminated by an empty `String` .

Chapter XVIII

Exercise 16

	Exercise : 16	points : 3
Substr		
Turn-in directory: (piscine_cpp_d03)/ex16		
Compiler: gcc	Compilation flags: -Wextra -Werror -Wall	
Makefile: No	Rules: n/a	
Files to turn in : String.h, String.c		
Remarks : n/a		
Forbidden functions : None		

You have to add a `substr` member function:

- `String* substr(String *this, int offset, int length);`
Extracts a sub string starting at `offset` and with the size `length` . The function returns the found sub string as a new instance of `String` . If `offset` is negative, it represents the number of characters starting from the end. If `length` is negative, it represents the number of characters to be copied at the left of the offset. If the specified sub string is in part outside of `String` , you just return the present part in `String` .