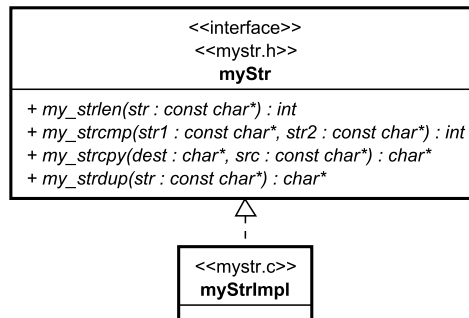# Read this first!!

When doing the exercises, it is advised that you make notes! These notes will be useful for the assignment you have to write later in the course.

Exercise 4.1 my_strcmp, my_strlen, my_strcpy, my_strdup

Look at the standard library functions strlen, strcmp, strcpy, strdup your functions should have the same signatures!!

```
                    <<interface>>
                     <<mystr.h>>
                        myStr
    ──────────────────────────────────────────
    + my_strlen(str : const char*) : int
    + my_strcmp(str1 : const char*, str2 : const char*) : int
    + my_strcpy(dest : char*, src : const char*) : char*
    + my_strdup(str : const char*) : char*
                         △
                         ┊
                    <<mystr.c>>
                     myStrImpl
    ──────────────────────────────────────────
```

1. Implement `my_strlen` first
2. Implement a function `my_strcmp`. You are **not allowed to** use built-in library functions to implement this function.
3. Make three tests of the function, using constant strings as parameters:
   - Compare "Hello World" with "Hello World"
   - Compare "Hello World" with the empty string
   - Compare two empty strings
   - Compare "Hello World" with "Hello, world"
4. Implement your own versions of the functions `strcpy`, and `strdup` (be inspired by Kernighan and Ritchie). You are **not allowed to** use built-in library functions to implement these functions.
5. Make some basic tests of the functions.