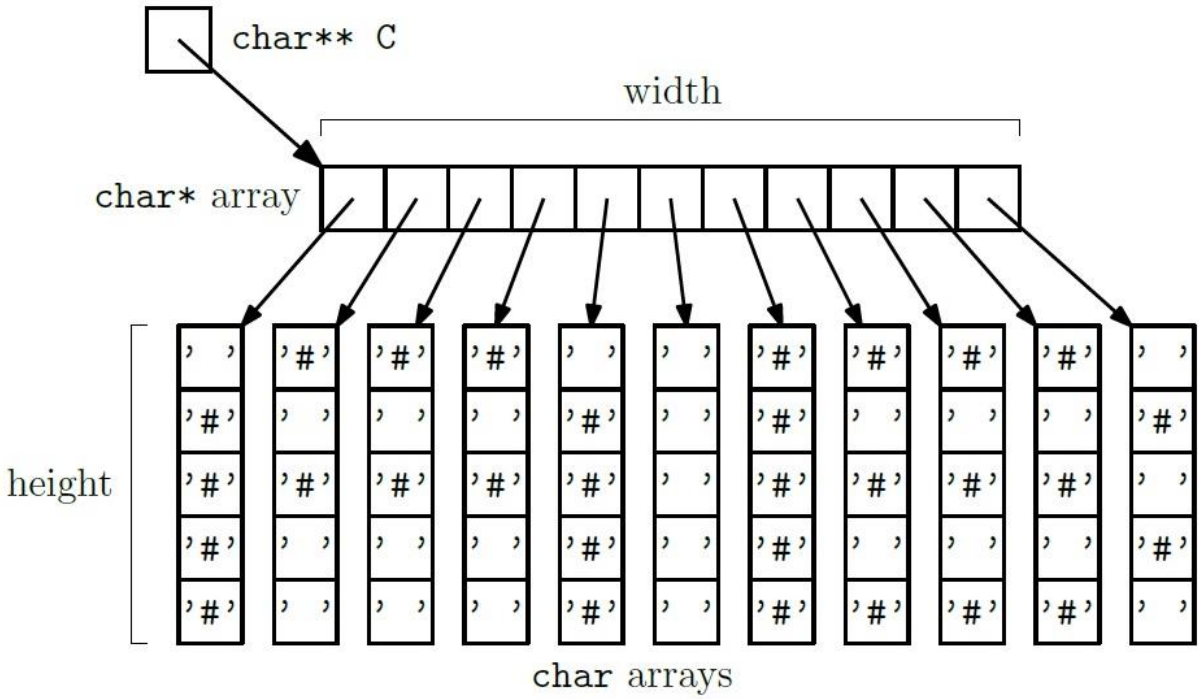
CSCI 2380 **Spring 2025** **Mr. Gustavo Dietrich**

Homework 3: Dynamic array resizing **Due Date:** 2/23/25

In this homework, you'll complete the implementation of the class that represents "ASCII art" images and operations on them, represented as two-dimensional char arrays allocated on the heap (using the **new** operator).



The following files are given to you:

1. A C++ header file (canvas.h) declaring the **Canvas** class.

2. A C++ source file (main.cpp) containing a main() function with tests.

Create a new C++ source file named **canvas.cpp** that implements the class declared in canvas.h so that canvas.cpp and the provided files compile into a program that runs with no failed tests.

Submit just the source code of **canvas.cpp**. You don't need to submit the main.cpp file because I will use my own canvas.h and main.cpp files to evaluate your canvas.cpp code. Do NOT make changes to my files.

Review the examples discussed in class and the textbook to get an idea of what you need to do.

Do not hesitate to use the corresponding topic in Discussions to post your questions/doubts about this assignment. I will reply as soon as I can.

IMPORTANT:

Make sure your program compiles and executes in full (it should pass all the tests included in main()).

You must submit ONLY ONE solution per team.

Your program must be well commented, use meaningful identifiers, and use indentation to improve its readability.

Your program must have the following comments at the top:

```
//*****************************************************************
// Team #                CSCI 2380        Spring 2025        Homework # 3
// First and Last Name
// First and Last Name
//
//*****************************************************************
```

When done, submit your solution through Blackboard using the "Assignments" tool. Do Not email it.
Paste the link to your final solution along with your source code in the textbox opened when you click on Create Submission before you click on Submit.

The following is the basic criteria to be used to grade your submission:

You start with 100 points and then lose points as you don't do something that is required.

-2: Did not pass test (each)
-40: Incorrect implementation of the grid (the array of pointers MUST point to COLUMNS)
-10: Incorrect implementation of Canvas(int width)
-10: Incorrect implementation of Canvas(char x)
-20: Incorrect implementation of Canvas(string s)
-10: Incorrect implementation of int width()
-10: Incorrect implementation of string to_string()
-10: Incorrect implementation of void replace(char old_char, char new_char)
-20: Incorrect implementation of void add(char x)
-10: Incorrect implementation of ~Canvas()
-40: Program does not compile

-10: Program produces memory leak
-10: Missing/too few comments
-5: Unnecessary statements in your code
-20: Incorrect/missing source code
-20: Incorrect/missing link to your solution
-100: The code submitted is not your creation (you got it from a web site or another person)
-100: No team solution
-100: No team contribution
-10: Late

Important: more points may be lost for other reasons not specified here.