

Project: implement a grade manager using Dynamic arrays in CPP

Pre:

Shortcut google docs to increase the font size of the selected document: **ctrl+shift+.**

This is how to use shortcuts for c++ in vs code:

1. Type `g++ main.cpp -o -main.exe` to compile `o` is a flag for naming it
 - a. Then instead of typing `main.exe` to run it
 - b. Just type `main.cpp` and you start typing in your current projects text editor
2. Shortcut for terminal open/close: `ctrl+j`

Upload `main.cpp` here with comments that walkthrough the challenges and things you learned.

```
#include <iostream>

class GradeManagerDynamicArray {
private:
    char* grades;
    int length;
    int capacity;

public:
    // Constructor: Initializes the dynamic array with a given
    capacity (default: 2)
    GradeManagerDynamicArray(int cap = 2): capacity(cap), length(0) {
```

```

        grades = new char[capacity];
    }

    // Function to add a new grade
    void addGrade(int grade) {
        if (length == capacity) {
            resize();
        }
        grades[length++] = grade; //increment grades array length by 1
and increment
    }

    void displayGrades() {
        std::cout << "TEST: Call to displayGrades is GOOD!" <<
std::endl; // Debugging calling displayGrades() ✓

        if (length == 0) { // I am now typing here!d====(¯\_/¯*)b
            std::cout << "No grades to display!" << std::endl;
        }
        for (int i = 0; i < length; i++) {
            std::cout << grades[i] << " ";
        }
        std::cout << std::endl;
    }

    void resize() {
        capacity *= 2; //double the capacity
        char* newArr = new char[capacity]; //allocate new array with
new capacity

        //copy existing grades to the new array
        for (int i = 0; i < length; i++) {
            newArr[i] = grades[i];
        }

        delete[] grades; //Free the old array memory
        grades = newArr; // point to the new array
    }

};

```

```

int main() {
    GradeManagerDynamicArray myGradeManager;
    int choice;
    char grade;

    while (choice != 3)
    {
        // Display menu options
        std::cout << "\n=== Grade Manager ===\n";
        std::cout << "1. Display grades\n2. Add a grade\n";
        std::cout << "Choose an option: ";
        std::cin >> choice;
        std::cout << std::endl;

        if (choice == 1) {
            std::cout << "TEST: Display grades was selected!" <<
std::endl;
            myGradeManager.displayGrades();
        }
        else if (choice == 2) {
            std::cout << "Enter a grade: ";
            std::cin >> grade;
            myGradeManager.addGrade(grade);
        }
        else if (choice == 3) {
            std::cout << "Exiting...\n";
            break;
        }

        std::cin.clear();
        std::cin.ignore(1000, '\n');
    }

    return 0;
}

```