

CPSC-121
Project #4

The purpose of this assignment is to get experience with a dynamically allocated memory and OOP.

The goal of this program is to create a graph of the class grade distribution as displayed in the sample output below. Additionally, the number of students is *dynamic*. That means the user will enter the number of students at the beginning of the program during its runtime and then provide the input for the actual data. Here is a sample program run. The text in bold is input from a user.

Please enter number of students: **4**

Please enter student name and grade:

Mary Smith 88

John Doe 100

Chris Bell 92

Tina Lee 95

Average: 93.75

Grade distribution:

0- 9:

10- 19:

20- 29:

30- 39:

40- 49:

50- 59:

60- 69:

70- 79:

80- 89: *

90- 99: **

100: *

The project requirements:

It is an important part of your grade that you design your application according to the following requirements.

Step 1. Create a pseudocode to list all the actions to execute along with the order these actions should occur. Include your pseudo code on the top of your cpp file in the comment section after your name and Project#.

Step 2. You must use a class and a dynamic array of objects. Use 3 files for this project as follows Student.h, Student.cpp, main.cpp.

Step 3. Create a user defined function to get a number of students.

Step 4. Create a user defined function to generate a dynamic array of objects. And store the values into the array of objects by calling the class member functions.

Step 5. Create a user defined function to calculate and display the class average grade.

Step 6. Create a user defined function to calculate grade distribution and display it along with the stars as shown in the example above.

Step 7. Remember to de-allocate dynamic memory and avoid dangling pointers.

Projects Submission Policy:

Your electronic file (PDF) must be submitted by the deadline listed on Titanium. The PDF should include your name, project#, code comments, UML diagram, C++ code and a screenshots of your program output. You should be able to explain any part of your program. The feedback on your program will be given in person during the lab hours. The grade of the project will be based on the following rule:

- Program works correctly: 5
- Program is mostly correct and has minor problem: 4
- Program has multiple significant errors (combination of incorrect output or calculation errors or missed project requirements): 3

The following kinds of submissions cannot be evaluated, and will be assigned a zero score:

- Email or late submissions.
- Source code that cannot be compiled successfully.
- Input/output that is falsified or does not match the submitted source code.
- Submissions that are plagiarized.