Assignment 2: Adjusting Arrays

Write a class named **ArrayList** that has an attribute to hold array of integers. Initially it has a capacity of 1 and is “empty”. At any time, this array can be partially full, so the class needs to track of its current occupied size and the total capacity (true size).

1. Add the following to this class **(90 points)**:
   a. A default constructor. It initializes the array to have all 0’s. **(10 points)**
   b. A function “push(int m)” that will add integer ‘m’ to the end of the array. If the array is already full, it must first double the size of the array before adding this number.
      i. **Note:** Double the array by creating a new array that is twice the capacity, copying all the elements of this array to the new one, and the making the new array as the “current” one. Be sure to free up any memory as necessary! **(35 points)**
   c. A function “erase(int m)”. This function deletes the leftmost occurrence of the number ‘m’ from the array by shifting everything else to its right by one position to the left.
      i. **Note:** the size of the occupied part of the array reduces by 1. If the array’s occupancy reduces to less than 50%, make the array half as big similar to how you expanded it. **(35 points)**
   d. A function “toString()” prints the contents of the array in order within “[ ]”. **(10 points)**

2. Provide a header file (*.h) for the class and all its operations/functions **(15 points)**
3. Provide the main.cpp (**10 points**)
4. makefile (**10 points)**

   It is highly recommended to write your own main function that tests your class in all possible ways (i.e. adding elements, deleting from the middle and either ends, allowing the array to expand and contract, etc.).

What to submit

Please submit all your source code (*.h, *.cpp) and a makefile as a single zipped file named “**assignment-5.zip**”.