Assignment 10 - Maximum Value in a Bag

Due May 9 by 11:59pm   **Points** 20   **Submitting** a file upload   **File Types** h and cpp   **Available** until May 12 at 12:01am

---

This assignment was locked May 12 at 12:01am.

ST. CHARLES
COMMUNITY COLLEGE

CPT-182 - Programming in C++

Programming Assignment - Maximum Value in a Bag (**20** Points)

(Number in Question Bank: Assignment **10.1**)

Program Overview

In this assignment, you are going to write a template class, **Bag**. A **Bag** is a simple container that can hold items. It is **unordered**. We **cannot** sort or order items in a **Bag**. A **Bag** is homogenous; that is, it can only hold items of a single type.

Your program should read in items in bags from an input file, and write the maximum value in each bag to the output file.

The **Bag** Class

1) The class is a template class.

2) In this assignment, you will **not** be given the exact ~~private~~ data fields in the class. Based on the functions required (see below), it is your responsibility to **professionally** determine the data fields in the class.

3) The class has a **default constructor** that initializes the bag to an empty state.

4) The class has the following functions:

   - **empty()**. The function returns **true** if **nothing** is stored in the bag; **false** otherwise.
   - **size()**. The function returns the number of items stored in the bag.
   - **push()**. The function takes an item (of what type?) as its only argument. The functions adds the item to the bag.
   - **max()**. The function returns the maximum value stored in the bag.

5) If **pointers** are included in the **private** data fields, then you should overload the deep-copy assignment operator, copy constructor, and destructor in the class.

6) For each function defined in this class, it is your responsibility to correctly determine the function return type, function argument(s), whether the argument(s) should be passed by value, reference, or **const** reference, and whether the function should be a **const** function, **static** function, and/or **friend** function. **Failing** to do these correctly will result in **losing points**.

7) Please do **not** forget that for a template class, you should **not** write a **.cpp** file to implement the class. Instead, you need to write everything in the **.h** file (header file).

The Input File

- The input file is a **plain text file** (filename: **bags.txt**).

- In each row of the input file, the first data field is the type of bag, either **"integer"**, **"double"**, or **"string"**. The second data field is a positive integer representing the size of the bag. After that, values in the bag follow in the same line. The values are **not** sorted in any way (see below).

integer	5	−3 6 23 15 −2
⬆ The bag stores integers.	⬆ Size of the bag is **5**.	⬆ These are the **5** items in the bag.

- You **cannot** assume (or guess) the number of bags in the input file. In other words, no matter how many bags are stored in the input file, your program should correctly process all of them.

- There may be empty lines at the beginning, in the middle, and/or at the end of the input file. Your program should smartly skip those empty lines and process only the rows containing data.

- Please refer to the **sample input file** to better understand the input file format.

The Output File

- The output file is a **plain text file** (filename: `results.txt`).

- When your program finds the maximum value in a bag, it writes the maximum value to the output file. Each value should be a separate line in the output file.

- Please refer to the **sample output file** to better understand the output file format.

Sample Input and Output File (Click to Download)

Sample Input File ⧉ (https://drive.google.com/uc?export=download&id=1OdQE7d3sT_02P6tuidzk1njIhMjbMUcz)**Sample Output File** ⧉

Assignment Submission and Grading (Please Read)

- Please upload all your `.h` (if any) and `.cpp` files (**not** the ~~entire Microsoft Visual Studio project folder~~) on Canvas.

- Before the assignment deadline, you can submit your work <u>unlimited times</u>. However, only your <u>latest submission</u> will be graded.

- At least **20**% of your code should be **comments**. All variable, function (if any), and class (if any) names should "make good sense". You should let the grader put **least effort** to understand your code. Grader will **take off points**, even if your program passed all test cases, if he/she has to put extra **unnecessary** effort to understand your code.

- Please **save a backup copy** of all your work in your computer hard drive.

- Your program will be graded (tested) using another valid input file (still named `bags.txt`) to check whether it can generate the expected (correct) output file (with correct format and correct output values in it). As long as the input file is valid, your program should generate a correct output file. In other words, your program should work for **any** valid input file, **not** just the sample input files provided in the assignment instructions.

- In this class, you can assume that the input file (input data) is always **valid** and **has correct format**. You do **not** need to deal with ~~invalid input~~ or ~~error handling~~.

- Your work will be graded after the assignment deadline. All students will receive their assignment grades at (almost) the same time.