CS 301 Data Structures and Algorithms

Homework 4

In this Homework, you are going to write a program that implements a ***doubly linked list***.

The data must be read from the ***Letters.txt*** and ***Sequence.txt*** the file names must be taken as arguments from commandline.

Letters.txt contains 26 English Letters:

A

B

C

D

…

Z11

Sequence.txt contains a list of numbers.

3,-3,19,-19,18,1,-2,3,-18,17,1,-3,-13,14

You need to

1. Create a doubly linked list by reading the 26 letters from Letters.txt.
2. Sequence.txt file contains a sequence of ***relative*** letter positions. The initial position is at the beginning of the list which is 'A'. The first number in the sequence is '3'. So the current position will move forward for three steps. Then the first letter to output is 'D' and now the current position is at 'D' as well. The second number in the sequence is '-3'. Then the second letter to output is the third element ***backward*** from ***the current position*** which is 'A'. You need to output all letters indicated in sequence.txt

Your need to

1. Create a ***doubly linked list***
2. Implement putItem() method that puts each letter at the end of the doubly linked list. Then use this method to create the doubly linked list.
3. Correctly read the files and interpret the information
4. Print out the letters

Requirements:

1. ***[will be 0 if it does not compile or crash]*** The homework must be done in C++ and compatible to C++11. It is REQUIRED to use g++ and provided Makefile to compile and run the code. Your submission must be in a .zip or a .tar.gz file that includes the source file hw2.cpp, hw2.h, and the datafile. You are NOT allowed to use Standard Template Library to create the linked list. You are NOT allowed to make any modifications to the Makefile or the data file. It is REQUIRED to use an ItemType class and a NodeType struct to solve this homework.
2. [5%] The Following identification information must be included at the beginning of your cpp file.
   //Name: XXXXXXX
   //NetID: ab1234
   //Email: XXXX@csueastbay.edu
3. [40%] Must implement Doubly Linked List class
4. [45%] Must use relative positions to calculate the output using next and back points in the doubly linked list and correctly output the results
5. [10%] Correct I/O of the data file. The content of the files must be read via file I/O.