

CSC3002: Introduction to Computer Science

Assignment 2

There are four questions in Assignment 2. Please find the zip file of the QT project for Assignment 2 in the BB system. **And please refer to the test program in “main.cpp” for each question to implement your programs and test them accordingly.**

Question 1:

Even though comments are essential for human readers, the compiler simply ignores them. If you are writing a compiler, you therefore need to be able to recognize and eliminate comments that occur in a source file.

Write a function

```
void removeComments(istream & is, ostream & os);
```

that copies characters from the input stream **is** to the output stream **os**, except for characters that appear inside C++ comments. Your implementation should recognize both comment conventions:

- Any text beginning with **/*** and ending with ***/**, possibly many lines later.
- Any text beginning with **//** and extending through the end of the line.

The real C++ compiler needs to check to make sure that these characters are not contained inside quoted strings, but you should feel free to ignore that detail. The problem is tricky enough as it stands.

One sample input file: (You can create the sample input file by yourself to check your program)

```

/*
 * This program generates a list of the powers of two up to an exponent limit
 * entered by the user.
 */
#include <iostream>
using namespace std;

/* Function prototypes */
int raiseToPower(int n, int k);

/* Main program */
int main() {
    int limit; // define the limit of the powers
    cout << "This program lists powers of two." << endl;
    cout << "Enter exponent limit: ";
    cin >> limit;
    for (int i = 0; i <= limit / 2; i++) { // for each power, we print it out
        cout << "2 to the " << i << " = " << raiseToPower(2, i) << endl;
    }
    return 0;
}

/*
 * Function: raiseToPower
 * Usage: int p = raiseToPower(n, k);
 * -----
 * Returns the integer n raised to the kth power.
 */
int raiseToPower(int n, int k) {
    int result = 1;
    for (int i = 0; i < k; i++) {
        //i = i + k;
        result *= n;
    }
    /* return result;
}

```

The sample output file will be

```

#include <iostream>
using namespace std;

int raiseToPower(int n, int k);

int main() {
    int limit;
    cout << "This program lists powers of two." << endl;
    cout << "Enter exponent limit: ";
    cin >> limit;
    for (int i = 0; i <= limit / 2; i++) {
        cout << "2 to the " << i << " = " << raiseToPower(2, i) << endl;
    }
    return 0;
}

int raiseToPower(int n, int k) {
    int result = 1;
    for (int i = 0; i < k; i++) {
        result *= n;
    }
}

```

Note that you code should be able to deal with the case with only “/*” (the corresponding “*/” does not exist) in which all characters following “/*” will be removed.

Question 2

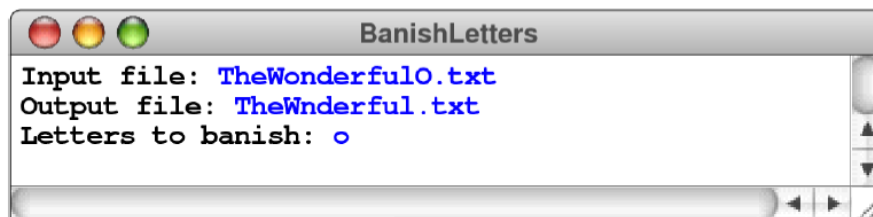
Write a program that asks the user for an input file, an output file, and a string of letters to be eliminated. The program should then copy the input file to the output file, deleting any of the letters that appear in the string of censored letters, no matter whether they appear in uppercase or lowercase form.

As an example, suppose that you have a file containing the first few lines of Thurber's novel, as follows:

TheWonderful0.txt

```
Somewhere a ponderous tower clock slowly
dropped a dozen strokes into the gloom.
Storm clouds rode low along the horizon,
and no moon shown. Only a melancholy
chorus of frogs broke the soundlessness.
```

If you run your program with the input



it should write the following file:

TheWnderful.txt

```
Smewhere a pnderus twer clck slwly
drppd a dzen strks int the glm.
Strm cluds rde lw alng the hrzn,
and n mn shwn. nly a melanchly
chrus f frgs brke the sundlessness.
```

If you try to get greedy and banish all the vowels by entering **aeiou** in response to the prompt, the contents of the output file would be

```
Smwhr pndrs twr clck slwly
drppd dzn strks nt th glm.
Strm clds rd lw lng th hrzn,
nd n mn shwn. nly mlnchly
chrs f frgs brk th sndlssnss.
```

Question 3

In the third century B.C.E., the Greek astronomer Eratosthenes developed an algorithm for finding all the prime numbers up to some upper limit N . To apply the algorithm, you start by writing down a list of the integers between 2 and N . For example, if N were 20, you would begin by writing the following list:

2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

You then circle the first number in the list, indicating that you have found a prime. Whenever you mark a number as a prime, you go through the rest of the list and cross off every multiple of that number, since none of those multiples can itself be prime. Thus, after executing the first cycle of the algorithm, you will have circled the number 2 and crossed off every multiple of 2, as follows:

② 3 ~~4~~ 5 ~~6~~ 7 ~~8~~ 9 ~~10~~ 11 ~~12~~ 13 ~~14~~ 15 ~~16~~ 17 ~~18~~ 19 ~~20~~

To complete the algorithm, you simply repeat the process by circling the first number in the list that is neither crossed off nor circled, and then crossing off its multiples. In this example, you would circle 3 as a prime and cross off all multiples of 3 in the rest of the list, which results in the following state:

② ③ ~~4~~ 5 ~~6~~ 7 ~~8~~ ~~9~~ ~~10~~ 11 ~~12~~ 13 ~~14~~ ~~15~~ ~~16~~ 17 ~~18~~ 19 ~~20~~

Eventually, every number in the list will either be circled or crossed out, as shown in this diagram:

② ③ ~~4~~ ⑤ ~~6~~ ⑦ ~~8~~ ~~9~~ ~~10~~ ⑪ ~~12~~ ⑬ ~~14~~ ~~15~~ ~~16~~ ⑰ ~~18~~ ⑲ ~~20~~

The circled numbers are the primes; the crossed-out numbers are composites. This algorithm is called the *sieve of Eratosthenes*.

Write a program that uses the sieve of Eratosthenes to generate a list of the primes between 2 and 1000.

Question 4

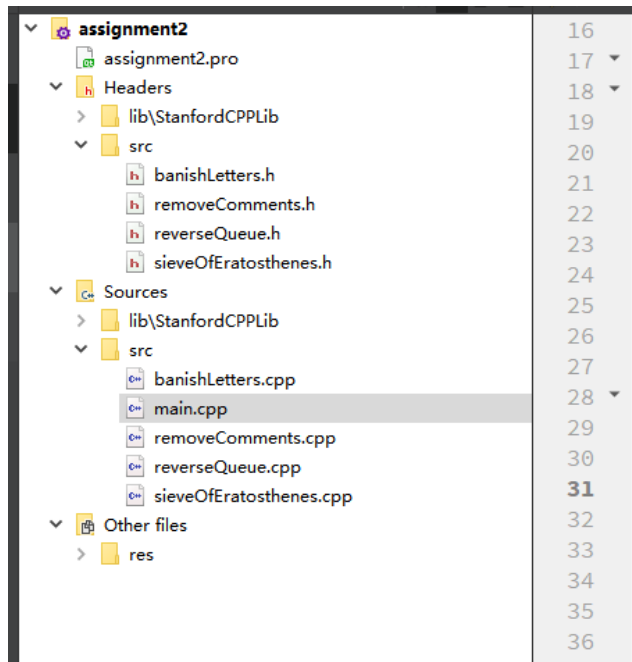
Write a function

```
void reverseQueue(Queue<string> & queue);
```

that reverses the elements in the queue. Remember that you have no access to the internal representation of the queue and must therefore come up with an algorithm—presumably involving other structures—that accomplishes the task.

Assignment Submission

Please find the attached assignment1.zip in the BB system which contains an QT project created for assignment2 and you can edit the corresponding files for above questions to complete the tasks.



After you test all your questions, zip the whole project

名称	修改日期	类型	大小
lib	2020/10/2 23:41	文件夹	
res	2020/10/2 23:41	文件夹	
src	2020/10/2 23:52	文件夹	
assignment2.pro	2020/10/2 23:51	Qt Project file	12 KB

in one file named XXX.zip (XXX is your student ID and your name is not required) and then submit it to the BlackBorad (BB) system. Please pay attention to this because only source files cannot be executed on our computer which will affect your scores!!!!!!!!!!!!

Special Notes

1. Please refer to the grading criteria documents for detailed descriptions
2. Do not rename functions or change its type signature.
3. We are expecting ****beautiful code layout**** and ****consistent code style****.
4. We recommend you to use Qt creator to write the code. Though ****not required****, we hope to can eliminate warnings as much as you can. They are definitely useful for you to get rid of some apparent mistakes.
5. Some good code style guide to follow:
 - LLVM Style: <https://llvm.org/docs/CodingStandards.html>
 - GNU Style: <https://gcc.gnu.org/codingconventions.html>
 - Google Style: <https://google.github.io/styleguide/cppguide.html>

We do not require you to follow any specific coding style but please make it beautiful and consistent.