

## cisc1110, fall 2010, assignment for unit VIII

---

This is the assignment for unit VIII, “basic algorithms.” You are expected to complete the assignment in the C++ language and submit one “stats.cpp” file. You must complete and submit the assignment on or before the due date of **December 1**. *Remember: Assignments will NOT be accepted more than 7 days late.*

This assignment is worth **5 points**.

### Submission instructions.

- Submit your assignment to me via email: `sklar@sci.brooklyn.cuny.edu`
- Your email subject line should be: **CISC 1110 Lab VIII submission**
- Attach your C++ `stats.cpp` file to your email.
- Make sure your name is in the body of the email message.
- Make sure your name is also included in the header comments at the top of your C++ file.

### Overview.

This assignment involves reading a data file and computing several types of statistics on the data in the file.

This assignment will reinforce your knowledge of file handling and will also give you a chance to test and demonstrate your understanding of basic algorithms.

### Assignment Description.

Start by downloading the sample data file from the class web page (linked on the November 18 section of the syllabus page: <http://www.sci.brooklyn.cuny.edu/~sklar/cisc1110/notes/games.dat>).

Have a look at the file. Note that there are **65** lines in the file:

- Lines that start with a # are considered like “comments” and do not contain data.
- Lines that do contain data are called *data records*. (These are the lines that do NOT have a # as the first character in the line.)

**For each step, below, write pseudo code and include it in a comment block preceding each function.**

**Write a separate function for each step.**

*HINT: you can open and close a data file multiple times in a program.*

1. Write a program that counts and outputs the number of data records and the number of comment lines in the file. (Comment lines begin with the '#' character, and data records do not.)  
(1 point: 0.3 points pseudocode, 0.7 points C++ code)
2. Modify your program so that it computes the average “Impact Factor” for the queries listed in the file. As we did in class (Nov 18), take advantage of the counts you compute in the previous step to first read the comment lines (using `getline()`) and then read the data records (using `>>`).  
(1 point: 0.3 points pseudocode, 0.7 points C++ code)
3. Modify your program so that it finds and displays the query with the highest “Query Popularity.”  
(1 point) (1 point: 0.3 points pseudocode, 0.7 points C++ code)
4. Modify your program so that it finds and displays the query with the lowest “Query Popularity.”  
(1 point) (1 point: 0.3 points pseudocode, 0.7 points C++ code)
5. Modify your program so that it asks the user to enter a query and searches for the user’s query. If the user’s query is found, then display the statistics for that query (impact factor, query popularity and QCI). If the user’s query is not found, then display a “not found” message.  
(1 point: 0.3 points pseudocode, 0.7 points C++ code)