

**CMPS 150 – Fall 2005**  
**Programming Assignment #7**  
2005.10.31

**Date Assigned:** Monday, October 31, 2005  
**Due Date:** 10:00 PM, Sunday, November 6, 2005

**NOTE !!!!**  
*pa7 is worth 3% of  
your overall grade*

The coded solution to the following problem is to be done by you and only you. You may ask for help from the class teaching assistants and the instructors, but you may not ask for C++ help from anyone else. You may use your notes, C++ texts, online tutorials, etc., but the code must be your own.

If you have a problem with your class account, compiling or debugging your code, or if you are not certain if you have submitted correctly, come see a TA or instructor as soon as possible.

1) **Include the following information as comments in the header of your source code:**

**Author:** *Your-Name*  
**CLID:** *Your-login-ID*  
**Class:** *CMPS 150 Section Your-Section-Number*  
**Assignment:** *pa7*  
**Date Assigned:** *Monday, October 31, 2005*  
**Due Date:** *10:00 PM, Sunday, November 6, 2005*  
**Description:** *A brief description of the purpose of the program.*

**Certification of Authenticity:**  
I certify that this assignment is entirely my own work.

2) **While in your class folder, enter the C++ code for the following description into a file named pa7.cc**

3) **Problem Description:**

**Section 3** You have a new TA (Hollie Boudreaux) – new submission CLID: **hmb7226**

This program will use the features of C++ that we have learned thus far in CMPS150, focusing primarily on reading input data from a file until EOF and writing void functions.

Your program is to be a menu-driven program, giving the user options of different types of processing of a file containing either text data or numeric data. Your menu must have the following options on it:

1. Count All Occurrences of Vowels (upper and lowercase)
2. Count All Occurrences of Consonants
3. Count All Characters
4. Find Employee with Highest Hourly Rate
5. Find Employee with Lowest Hourly Rate
6. Quit

You must check that a valid menu selection has been given. If it is invalid, an error message should be displayed and the menu/input prompt repeated until valid data is given. The program must process each selection (except Quit) using a call to a void function. Main should only "drive" the program.

### Processing

Selection of option 1, 2 or 3 will use an input file named "pa7.text", while options 4 and 5 will allow the user to enter the name of the input file. Remember to use `.c_str()` when opening a file using a `string` variable.

NOTE: When counting characters, do NOT count white space.

### Input Files

The input file "pa7.text" will contain a "bunch of" text. A sample is shown below.

The input file whose name is entered by the user will contain employee record data. There will be an ID (string with NO spaces), hourly rate (floating point number) and hours worked (floating point number) for each employee – in that order. Although you must read all three(3) pieces of data for each employee, you are only concerned with the hourly rate. A sample is shown below.

### Sample Text File

```
Before formally discussing predefined functions
in C++, let us review a concept from a college
algebra course. In algebra, a function can be
considered a rule or correspondence between
values, called the function's arguments, and the
unique value of the function associated with the
arguments. Thus, if  $f(x)=2x+5$ , then  $f(1)=7$ ,
 $f(2)=9$ , and  $f(3)=11$ , where 1, 2, and 3 are the
arguments of  $f$ , and 7, 9, and 11 are
corresponding values of the function  $f$ .
```

### Sample Employee File

*order of data:*

**id**  
**hourly rate**  
**hours**

```
ABC123 7.25 30
DEF456 9.90 20.25
GHK789 20.50 40
LMN234 12.85 10.75
QRS345 8.00 35
TUV678 10.45 31.5
XYZ567 6.95 28.0
```

Again, you **MUST** name your source file 'pa7.cc' and you **MUST** store it in your class directory (cs150x).

See next page for Sample Run !!!

**Sample Run:**

1. Count All Vowels (upper and lower case)
2. Count All Consonants
3. Count All Characters
4. Find Employee with Highest Hourly Rate
5. Find Employee with Lowest Hourly Rate
6. Quit

Choice: 1

There are 127 vowels in pa7.text !

1. Count All Vowels (upper and lower case)
2. Count All Consonants
3. Count All Characters
4. Find Employee with Highest Hourly Rate
5. Find Employee with Lowest Hourly Rate
6. Quit

Choice: 2

There are 200 consonants in pa7.text !

1. Count All Vowels (upper and lower case)
2. Count All Consonants
3. Count All Characters
4. Find Employee with Highest Hourly Rate
5. Find Employee with Lowest Hourly Rate
6. Quit

Choice: 3

There are 376 characters in pa7.text !

1. Count All Vowels (upper and lower case)
2. Count All Consonants
3. Count All Characters
4. Find Employee with Highest Hourly Rate
5. Find Employee with Lowest Hourly Rate
6. Quit

Choice: 4

Enter the name of the employee file: emp.data

Error opening input file: emp.data -- Option Terminated!

1. Count All Vowels (upper and lower case)
2. Count All Consonants
3. Count All Characters
4. Find Employee with Highest Hourly Rate
5. Find Employee with Lowest Hourly Rate
6. Quit

Choice: 4

Enter the name of the employee file: emp.dat

Maximum hourly rate found in "emp.dat": \$20.50

CMPS 150 – Fall 2005 -- Programming Assignment #7

1. Count All Vowels (upper and lower case)
2. Count All Consonants
3. Count All Characters
4. Find Employee with Highest Hourly Rate
5. Find Employee with Lowest Hourly Rate
6. Quit

Choice: 5

Enter the name of the employee file: emp.dat  
Minimum hourly rate found in "emp.dat": \$6.95

1. Count All Vowels (upper and lower case)
2. Count All Consonants
3. Count All Characters
4. Find Employee with Highest Hourly Rate
5. Find Employee with Lowest Hourly Rate
6. Quit

Choice: 6

Thanks -- and Bye !!

---

**4) Additional Requirements:**

- Use comments as appropriate. Refer to “Programming Style Sheet” on the CMPS150 website.
- Your program must use good names for all variables and named constants. (Good names are names that are descriptive of the values stored or the function performed.)
- Adhere to style requirements. See “Programming Style Sheet” on the CMPS 150 web site.

**5) Name your source file 'pa7.cc' and store it in your class directory (cs150x).**

**6) Compile your program and test it (see Some Unix Help for quick assistance).**

To compile:

```
g++ -o pa7run pa7.cc
```

To run (execute):

```
pa7run
```

**7) After it is debugged and running correctly, submit pa6.cc (the source file only) electronically by 10:00 PM, Sunday, November 6, 2005 to receive full credit.**

```
submit -d pa7.cc
```

The CLID for the TA of your section is the name to put in. This is one of the following:

<b>Section 3</b> You have a new TA – therefore a new submission CLID: <b>hmb7226</b>
--

<u>Section</u>	<u>TA</u>	<u>CLID of TA</u>
Section 3.....	Hollie .....	hmb7226
Section 4.....	Anca.....	axd9917
Section 5.....	Mitun .....	mxb2169
Section 6.....	Jason .....	jbm8240

You will be asked to enter the assignment name and the CLID of the person it is to go to. The assignment name is:

**assn7**

REMINDER: You can turn in programs up to 24 hours late for 75% credit, or up to 48 hours late for 50% credit.

**NOTE: Programs that do NOT compile will receive a grade of zero !!!**