

**CMPS 150 – Fall 2005**  
**Programming Assignment #9**  
2005.11.18

**ABSOLUTELY**  
**NO Late Assignments Accepted !!!**

**Date Assigned:** Friday, November 18, 2005  
**Due Date:** 11:00 PM, Monday, November 28, 2005

**NOTE !!!!**  
*pa9 is worth 4% 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: pa9
Date Assigned:  Friday, November 18, 2005
Due Date:       11:00 PM, Monday, November 28, 2005
Description:    A brief description of the program's purpose
```

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 pa9.cc**

**3) Problem Description:**

This program will use the features of C++ that we have learned thus far in CMPS150, focusing on arrays and string functions.

Your program will NOT be a menu-driven program (really). There are two tasks to be performed. We encourage using functions, but there are no specific function requirements for pa9.

**Processing**

**Task #1:** The first task is to process a file containing exam data (see sample on next page). The first line of the file will contain exactly 20 characters, each of which is either a 'T' (for true) or an 'F' (for false). These characters are the correct answers to the 20 T/F question exam taken by some students. Following the line with the correct ("key") answers, there will be an unknown number of lines, each of which will contain a student ID (no spaces), followed by exactly one space, followed by 20 characters indicating the answers that that student filled in on their exam. Please note that a student may have not answered a question, therefore the character for that is a space. HINT: Use `inFile.get(ch)` to get spaces, including characters. You must use an array (of type `char`) to read in the "key" answers. Then as you read in each student, compare their answers to the "key" answers, and tally the number of correct answers. When done tallying, compute their exam grade by multiplying their total points by 5 (with 20 questions, each is 5% of the overall exam grade). Finally, determine the letter grade for the student (using a standard 10-point scale).

You must print out (to an output file): 1) the answers read in the "key" (in a neat / meaningful manner), 2) for each student: a) their ID, b) their answers, c) their grade (out of 100), and d) their corresponding letter grade. See the sample output file for a suggesting of f the format for your output (see sample output file #1).

**Task #2:** The second task is to process two input files containing political candidate election information. The first input file contains no more than 50 candidate names. The name is formatted as follows: last name (which may contain spaces), followed by a comma, followed by exactly one space, followed by first/middle name (see sample input file #2a). The second input file contains the number of votes received by each of the candidates in the first input file (see sample input file #2b).

Output for this task is a table of candidate information and election results, followed by a statement of who won the election. The table must contain at least 3 columns: votes received, percentage of total votes cast, and candidate name (printed with first name first, then last name – and comma must be removed). See sample output file #2. You may choose the number of decimal places printed.

**Input Files**

Each task must allow the user to specify the name of the input and output files. Please note that there are two input files for task #2.

**Sample Input File #1**

*order of data:*  
exam "key" answers  
multiple lines of:  
student ID  
student answers

```
TFTFTTFFTTTTTTTTTTTTT  
ABC123 FTFTFTFTFTFTFTFT T  
CDE456 FFFFFFFFFFFFFFFFFF  
FGH789 TTTTTTTTTTTTTTTTTT  
IJK012 FFFFFFFF TTTT  
LMN345 TTFFTTFFTTFFTTFF
```

**Sample Input File #2a** (candidate names)

```
Smith, Bob  
Taylor, Johnny Boy  
Boudreaux, Jane  
Thibodeaux, Jean Doris  
Hebert, Elizabeth  
Vail, Bonnie Nicole  
Davidson, Kyle Slater  
Van de Kamp, Kristi  
Newton, Sir Isaac  
King, Dr. Martin Luther
```

**Sample Input File #2b** (votes received)

```
4982 287 2388 9329 5000 3865 817  
1 999 6000
```

**Sample Output File #1**

Exam Key				
-----				
Question # 1:	T			
Question # 2:	F			
Question # 3:	T			
Question # 4:	F			
Question # 5:	T			
Question # 6:	T			
Question # 7:	F			
Question # 8:	F			
Question # 9:	T			
Question #10:	T			
Question #11:	T			
Question #12:	F			
Question #13:	F			
Question #14:	F			
Question #15:	F			
Question #16:	F			
Question #17:	T			
Question #18:	T			
Question #19:	T			
Question #20:	F			
Student Exam Results				
-----				
ABC123	FTFTFTFTFTFTFTFTFT T	30	F	
CDE456	FFFFFFFFFFFFFFFFFFFFF	50	F	
FGH789	TTTTTTTTTTTTTTTTTTTTT	50	F	
IJK012	FFFFFFFF TTTT	35	F	
LMN345	TFTFTFTFTFTFTFTFTFF	70	C	

**Sample Output File #2**

Votes	% of Total	Candidate
-----		
4982	14.80	Bob Smith
287	0.85	Johnny Boy Taylor
2388	7.09	Jane Boudreaux
9329	27.71	Jean Doris Thibodeaux
5000	14.85	Elizabeth Hebert
3865	11.48	Bonnie Nicole Vail
817	2.43	Kyle Slater Davidson
1	0.00	Kristi Van de Kamp
999	2.97	Sir Isaac Newton
6000	17.82	Dr. Martin Luther King
The winner is: Jean Doris Thibodeaux, with 27.71% of the total vote.		

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 'pa9.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 pa9run pa9.cc
```

To run (execute):

```
pa9run
```

7) **After it is debugged and running correctly, submit pa9.cc (the source file only) electronically by 11:00 PM, Monday, November 28, 2005 to receive full credit.**

```
submit -d pa9.cc
```

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

<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:

```
assn9
```

**REMINDER: NO Late Programs Accepted !!!**

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