

CS250 Assignment 2 Lottery

Date assigned: Friday, February 3, 2017
Date due: Monday, February 13, 2017
Points: 30

A file **lottery.txt** contains up to 10,000 lottery tickets where each ticket can contain up to 6 lottery numbers and no fewer than 4 lottery numbers. Each lottery ticket costs \$1.00. Fifty percent of the lottery proceeds will go to a good cause and the remaining 50% (winner's pool of money) will go to the ticket winners as follows:

1. Tickets matching $n - 2$ numbers will evenly divide 20% of the winner's pool
2. Tickets matching $n - 1$ numbers will evenly divide 30% of the winner's pool
3. Tickets matching all n numbers will evenly divide 50% of the winner's pool

Note: Order is important when matching tickets, so for instance, if the winning numbers are 1 2 9 10 15 18 and a ticket is 1 9 2 10 12 19, then two numbers were matched 1 and 10.

Your program is to work as follows. The bold text is user input where each value will be an integer separated by one or more spaces. Assume the user will enter the winning numbers correctly. To test your program, you must use the result of the warm up assignment to generate random lottery.txt files. I will grade your program using a few different scenarios.

To access lottery.txt from another project, you must use the following:

```
const string LOTTERY_TICKETS = "..\\01CreateLotteryTickets\\lottery.txt";
```

Lottery Results

Enter Winning Lottery Numbers: **X X X X X X**

Total Tickets Processed = XXXXX
Winner's Pool Of Money = \$XX.XX

Numbers Matched	Total Tickets	A Ticket Wins
-----	-----	-----
0	XX	\$ XX.XX
1	XX	\$ XX.XX
2	XX	\$ XX.XX
3	XX	\$ XX.XX

...

To complete this assignment you must submit the following:

1. An electronic copy of your program on Grace

- a) Add a project called **02ProcessLottery** to your existing solution **PUNetID-Assignments**. It is vital that you name your project correctly!
- b) Type your program (fully documented/commented) into the project. You need to follow the coding standards from the CS250 Web page. These coding standards have been modified to include additional C++ language features introduced in CS250, so please be sure to read the new coding standards.
- c) Pay attention to the example output. Your program's output must look **exactly** like the sample output. The spacing and newlines in your output must match exactly.
- d) Make sure that your program builds without errors & warnings and runs correctly. If you get any errors or warnings, double check that you typed everything correctly. Be aware that C++ is case-sensitive. You will lose 10% if there are any warnings and 40% if your program does not build successfully.
- e) Once you are sure that the program works, it is time to submit your program. You do this by logging on to Grace and placing your complete solution folder in the correct drop folder based on the section of the course in which you are enrolled (**CS250-XX Drop**).
- f) The solution must be in the drop folder by the time class starts on the day the assignment is due. Anything submitted after that will be considered late.
- g) If you drop multiple solutions, you will lose 10% of the assignment points, so do not drop until you are entirely sure you are completely done working on the assignment.

2. A hard copy of your program

- a) The hard copy must be placed on the instructor's desk by the time class starts on the day that it is due.
- b) The hard copy must be printed in color, double-sided, and stapled in the upper left corner if your solution contains multiple pages.
- c) Your tab size must be set to 2 and you must not go past column 80 in your output.

**Remember, if you have any problems, come to me straight away
with your project on a flash drive or on
Grace. Good Luck!!!! ☐**