Northern Marianas College

CS225: C++ Programming

Final Assignment: Casino Program

Objective:

In this lab you will create a C++ programming language program that utilizes input, output, selection and loop programming structures, arrays, and file write.

The program is due at the beginning of the final exam and no late projects will be accepted. You must submit a written document before the final exam time and upload the source code file casino.cpp using NMC Online. The written document must include a program specification document which includes: analysis, specifications, a planned user interface, test data to test all possible conditions of selection structures, and flow charts for all user defined functions. Also include a source code listing, and a copy of the output and the generated text file for several runs of the test data. Submit a conclusion section, which describes what sections worked of your program and what did not work with the program.

This assignment is worth 15 points or 7.5% of the points for the course.

Program Requirements:

The Dynasty Casino in Tinian has decided to automate the chip counting using a computer program. They heard of your programming talents and have contracted with you to create a chip counting program that will assist in cashing out client's chips. They have agreed to fly you over for an all expense paid week to observe the casino in action.

You enjoy your time in Tinian while doing your Analysis Phase "research" and Design Phase and write your "Program Specs". You have agreed that you will not leave Tinian until all parties agree with your Program Specifications Report and Design. Once the design is approved you will enter the Implementation Phase and return home to do your coding. The Dynasty Casino has agreed to pay you one red chip per hour until you complete the program.

The program will be started at the beginning of a cashier shift and be run on all cashier computers. The cashier will begin by entering their 6 digit CashierID. The program will assist the cashier by calculating the value of chips and determine tax withholdings based on the new IRS regulations for U.S. Citizens. A record of all customer cash out transactions will be saved to the file Cashier.txt. The network administrator will collect these files via the LAN network at the end of each shift. You observe that the general sequence of events at the cashier window is as follows:

The customer submits to the cashier all chips they wish to cash out. Chips are counted and the total sum is calculated based on the following values:

Black = $5
Blue = $20
Red = $50
Green = $100
Gold = $500

The customer is asked to present receipts for purchased chips. Since gamblers generally make several purchases of chips in one night the program will prompt for multiple entries. If the customer can not provide receipts, then all chips will be considered taxable winnings if they are a US Citizen.

The winnings are calculated based on the previous information by subtracting the receipt total from the chip value total. If the customer has lost they are simply paid cash for all chips.

If the customer has won, the following information needs to be requested and calculations made to determine withholding tax based on the latest IRS regulations:

1. U.S. Residents will have withholding tax subtracted from the winning.

2. All U.S. Residents must enter a Social Security Number.
3. Residents of the CNMI will have withholding tax subtracted from the winnings in the amount of five percent tax on winnings.
4. All other American residents (including Guam) will have withholding tax subtracted from the winnings in the amount of twenty percent tax.
5. Payment is then made to the customer for chips received minus the withholding tax.

Display summary information and append the same information to a file describing each transaction. The file will contain all transactions for the cashier's shift. It would be useful to provide a time stamp for each transaction. The time stamp can be achieved by using the following four lines of code at an appropriate place in your program:

```
#include <ctime>  // Link the ctime library functions
time_t DateTime;  // Declare variable to store date & time
DateTime = time(NULL);  // Determine current time and date
cout <<  asctime(localtime(&DateTime)); // Output time
```

.

Assignment Written Report

Submit, on or before the due date the following for grading: Cover sheet with your name and project name; Program specifications report that must contain your analysis, assumptions, and design; The design should include a flow chart, proposed output display, and known test data. Include a source code listing for your program and a printout of the display for several runs of test data. Include a conclusion section which contains information described above in objective section.