

April 01 2008

Project 1 (Mini-Project): Design Part

CS135 Lab 8

Chang Jia
chang@cse.unr.edu

Mini-Project

- Lab 8 – **April 8** - Due in lab, at beginning of lab.
- Lab 9 – **April 15** - Due in lab, at beginning of lab.
- NOTE: **Plan ahead!**
 - A lot of coding work!
 - **April 10** – Second Mid-Term!
- NOTE: Need use **curses library**.
- NOTE: Need at least **6 functions**, e.g. the number of functions used in the sample program is 15.
- NOTE: The use of **global variables** is **not allowed** for this project.

Lab 8

- Need to do **one complete design document** (**Necessary write-ups and complete design**) (No c++ code needed here, but wise to start coding ASAP!)
- Program descriptions should show **DETAILS** about Design!
 - Meet the requirement functions – **use functions** will save your efforts for re-writing same codes!
 - Require **one separate block** of description for each function!

Lab 9

- Need to do **one complete c++ program and necessary test cases** for Lab 9 (Start earlier!)
- C++ Code should follow the Design in Lab8!
 - Need to have **at least 6 Functions** according to lab7 design!
 - Require **make changes of design descriptions** if design is updated when coding!
- More details will be discussed next week in lab!

Grading Policy for Lab8

- Need to be typed and neatly presented
- Need to print the design out with title page
- Please check with **grading policy sheet**

Grading Policy

<http://www.cse.unr.edu/~chang/Labs/P1/DesignGradingOutline.pdf>

0 on lab 8 for:

- Handwritten submissions
- Missing functions from required minimum (6)

What to do for Lab8? – Design Document

- **Abstract**
- **Design Logic**
 - Overall design of the whole program
- **Details for each function**

Abstract

- **Abstract:** What are you designing?
Short description of the project.
- Design descriptions
 - A write-up briefly describing the requirements of the program **from the programmer's perspective**.
 - A short description of your **overall design of the project**.
 - Any **assumption** you made.

Design Logic

- Introduction:
 - Total number of function plan to use.
 - What are the functions you want to use.
 - How the program flows.
- Design with a chart: answer the following questions:
 - Where (**in what order**) you use these modules/functions.

Function Design Details

- Component/function name
- Function Prototype
- Relationship with other function (calls & called by)
- A list of formal parameter and return (with data type)
- A description of what the function does step by step

Function Design Details – Example

- **Name:** main
- **Prototype:** int main()
- **Called by:** the Operating System
- **Calls:** displayChoices, selectFile, printSummary
- **Parameters:** none
- **Returns:** An integer denoting success or failure to the operating system
- **Displays top level menu** by calling displayChoices, handles top level selection and then calls selectFile with the name of the appropriate file ("appetizer.txt", "entree.txt", "dessert.txt"), and appropriate corresponding text string ("Appetizer", "Main Entree", "Dessert") to display.

Check list

- No **handwritten/email submissions**; No **global variables**.
- No missing functions from required minimum (**at least 6**)
- Write-ups:
 - **Abstract**
 - **Design Logic**
 - Overall design of the whole program
 - **Details for each function**
- Note: Detailed **descriptions** for **each function** as separate block.
- Correct showing design logic.

Issues for Loops

- Nested for loops
 - Go download [Nested For Loop.cpp](#)
 - Run it, observe the result, and try to answer the questions
 - Make sure that you have no problem with all these questions