CS 159 – Spring 2012 – Lab #5

What will you submit?

1. The first task of this lab is a number of problems from and related to chapter 5 intended to improve your understanding of the material you will utilize in the lab assignment. **Be sure to show your lab instructor that you have successfully completed these tasks BEFORE you leave lab today.**
   - For your group to be eligible to earn points for the second task the first **must be completed to the satisfaction of your lab instructor.**
2. A single C-file (a file with a .c extension) will be submitted electronically via the **sage** server. An example submission was conducted during the first week of the semester in lab. If you have a concern regarding how to submit work, please **visit** course staff **prior** to the deadline for this, and all, assignments. **No late work will be accepted.**
   - You may keep the final pages of this document to complete the task outside of class if need be. Your lab instructor will collect **ONLY** your group communication responses before you leave lab today.
   - An electronic copy of this lab will be made available on Blackboard at the end of the week.

Lab Quiz #5:

The lab quiz will be made available during the last 10 minutes of your lab today. The quiz will emphasize logical data, logical operators, relational/equality operators, logical expression evaluation, complements, and the if/else construct, and the material in this lab. **Lab quizzes are individual efforts and you may not use any resources while completing the quiz.** Questions are presented one at a time and cannot be revisited. Be sure to save your answers to each question and to finish your quiz to ensure it is submitted for grading. Most problems on lab quizzes will be multiple-choice, matching, or true-false.

- For more information on taking lab quizzes (assessments) on Blackboard see the following: http://goo.gl/EPhY6
- Your lab quiz results will be available for review at the end of the week or early next week.

Reminders:

As a group you must determine who will make the final submission for your group, when that submission will be made, and how the concerns regarding submission will be communicated with the other members. **Only one person per group will make submissions for the entire group**. Lab instructors cannot be expected to grade submissions from multiple members of the same group to determine which submission you desire to be evaluated.

This programming assignment does not have a single solution. Our expectation is that each group will develop their own unique solution. Your submission will be processed by our software similarity service and results will be used to detect unacceptable collaboration between groups. The development of your algorithm and the resulting code should only be discussed between group members and course staff.

Groups are expected to meet outside of class to continue collaboration on this assignment. **Failing to participate will result in a zero.** A member that does not contribute to the satisfaction of all partners should have his/her career account login omitted from the assignment header. It is not acceptable to designate a single individual to complete the assignment. Every individual group member should have a full understanding of all work submitted. Assignments are an opportunity to develop and demonstrate your understanding of course material.

Individuals failing to remain on task during the lab will be assigned a zero for the lab, lab quiz, and will be considered absent for the lab. Web browsing, personal e-mail, social networking, and/or instant/text messaging is not permitted during lab.

It is a good idea to submit your work for grading prior to leaving lab today. This will allow each group member to **verify their contact information in the assignment header (head_lb)** and will e-mail each group member a copy of the work completed. If a group member cannot be contacted and he/she is the sole possessor of the work created in lab then the remaining group members may have to start over in order to complete the assignment.

CS 159 – Collaborative Group Communication

Name	Purdue ITaP Career Account ID	Initial Role Assigned (Driver, Navigator, Manager)

Which group member will make the submission for this assignment?

When will the final electronic submission be made?

When AND where is the group meeting next?

Who is responsible for bring the C programming text and class notes next week?

Solve the following problems related to material from Chapter 5:

Statement	True / False
`3 && -3 && 10 && -10`	
`3 || -3 || 10 || -10`	
`3 || 6 && 0`	
`3 == 4 || 6 || 8`	
`3 == 3 && -1 && 1`	
`6 % 2 || 7 % 2`	
Page 290 – Problem 19A	
Page 290 – Problem 19B	
Page 290 – Problem 19C	
Page 290 – Problem 19D	
Page 290 – Problem 19E	
The complement of `x > 3` is `x <= 3`	
The following two logical expressions are equivalent `!(x < 10)` and `x >= 10`	
A dangling `else` logical error can be avoided through the use of `{` and `}` with all nested `if/else` constructs.	
The compiler will issue a warning when an assignment operator rather than the equality operator is used in the logical expression of an `if` condition.	
The parentheses around the logical expression of an `if` statement are optional.	
A terminal semicolon after an `if` condition is optional.	
The false (`else`) statement is not required with every use of `if`.	

Write the complement to && each logical expression below:

`x > 0 && x < 10`   `y % 2 != 0 && x % 2 != 0`

What is the output generated by the code segment below?

```
int x = 4;
int y = 9;
int z = 0;

if(x++ % 2 && ++y % 2)
{
   z++;
}
else
{
   z--;
}

printf("x = %d y = %d z = %d\n", x, y, z);
```

Lab #5 – Programming Assignment
Due: 30 minutes prior to the start of your next lab meeting.
5 Points Possible

Collaborative Roles for the Lab Session

Collaborative Teaming. For this lab you will be working in your assigned collaborative teams. If you are unable to complete your assignment during the lab then it is expected that your team meet outside of class to finish and submit the problem assigned. **Be sure to exchange contact information during lab today!**

Role:	Description: **Every member will rotate roles approximately every 20-30 minutes.**
Driver	The driver is in charge of the computer which includes entering code, saving, testing, and submitting. This individual should be soliciting the other two members for advice.
Navigator	The navigator's primary objective is to look over the shoulder of the driver for syntactical errors, logical errors, and to encourage that course standards issues are addressed while the solution is being coded. The most common mistakes include failing to pair quotes, misplaced semi-colons, and improper placement of parentheses. **Now that user-defined functions** are a required part of lab assignments the navigator could be helpful to track all user-defined functions and their parameters.
Manager	The manager may not be as close to the driver as the navigator but still plays an important role. At this point in the semester the manager can be used to communicate the verification of the written problems with the lab instructor, ensure that a submission is made with the correct career account user IDs, and to complete the communication form to be submitted to your lab instructor before you leave lab today. The manager should track time and encourage the navigator and driver in their roles.

Problem: A student graduating from college shortly with a nice job offer in hand would like to put together a program that will help you plan how much you might borrow and the management of that loan.

For options #1 and #2 you will take into account the credit score of the potential loan applicant.

- Greater than or equal to 730 is a good score.
- Greater than 600 but less than 730 is an average score.
- And less than or equal to 600 is considered a poor score.

Option #1: You would like to calculate the monthly payment for a loan given the loan amount, the value of the home, the annual interest rate, and the term of the loan. If the applicant has a good credit score (defined above) then they are allowed to take out any loan. An applicant with an average score will need a loan to value ratio of 80% or less or they will have the conditions of the loan modified to represent the additional risk that the lender is assuming. The same is true with an applicant having a poor credit score with a requested loan that is less than or equal to 80% of the home value. Any applicant with a poor credit score that requests a loan to value ratio greater than 80% will be denied as this is more risk than the lender is willing to accept.

- Modified loans will be 80% of the requested loan amount at the given term and annual interest rate with the remaining 20% of the loan at a rate with an additional 2% AND is 10 years less to build up equity faster.

Credit Score	Loan Amount	Home Value	Annual Interest Rate	Term	Monthly Payment
750	$100,000.00	$100,000.00	6.50%	30	$632.07
700	$100,000.00	$100,000.00	6.50%	30	$679.22
575	$100,000.00	$100,000.00	6.50%	30	Denied Loan
700	$75,000.00	$100,000.00	6.50%	30	$474.05
575	$75,000.00	$100,000.00	6.50%	30	$659.14

Option #2: Those with a good credit score would like to know how much they can borrow given the terms of the loan and a desired monthly payment. Those without a good credit score are not permitted to make this calculation.

```
Select one of the following:              Select one of the following:

 1. Calculate monthly payment.             1. Calculate monthly payment.
 2. Determine maximum purchase.            2. Determine maximum purchase.
 3. Calculate remaining balance.           3. Calculate remaining balance.

Enter your option: 2                      Enter your option: 2

Enter your credit score: 735              Enter your credit score: 725
Enter the term of the loan in years: 30
Enter the annual interest rate of the loan: 6.5 Based on your input you are not eligible for a
Enter desired monthly payment: 850           loan.

Maximum Loan: $134479.20
```

Option #3: All applicants, assuming a traditional loan (rather than the 80-20 modified loan described in option #1) determine the remaining balance of the loan given the number of payments made.

```
Select one of the following:              Select one of the following:

 1. Calculate monthly payment.             1. Calculate monthly payment.
 2. Determine maximum purchase.            2. Determine maximum purchase.
 3. Calculate remaining balance.           3. Calculate remaining balance.

Enter your option: 3                      Enter your option: 3

Enter the term of the loan in years: 30   Enter the term of the loan in years: 30
Enter the annual interest rate of the loan: 6.5 Enter the annual interest rate of the loan: 6.5
Enter number of payments made: 10         Enter number of payments made: 350
Enter original loan amount: 100000        Enter original loan amount: 100000

Remaining Balance: $99073.63              Remaining Balance: $6136.39
```

Would you like to test the program/solution with your own input?

- Type `lab05` from your UNIX prompt. Let us know if you find any logical errors! 😊

Additional (VERY IMPORTANT, CRITICAL, ESSENTIAL, NON-NEGOTIABLE) Assignment Requirements:

1. Add the `head_lb` lab assignment header file to the top of your program. Include the Purdue University e-mail addresses of each **contributing** group member. A description of your program will be included in the assignment header.
2. A program **MUST** compile to be considered for partial credit. The attempt to submit a program that does not compile will be rejected.
3. Your program **MUST** accept input and produce output in the same manner as seen in the sample executable program provided.
4. You **MUST** make good use of user-defined function as outlined in chapter 4 of your text. Placing all code in the `main` function will result in a zero for this assignment.
   - What is acceptable to retain in the `main` function? The `main` function should make a majority of the user-defined function calls in the program, be the place where most of the data needs (variables) are declared, and a limited amount of control structures (selection) is permissible to assist `main` in accomplishing the previous two tasks.
5. **No material found beyond chapter 5** of the text and notes is permissible for use in your solution.

Course Programming and Documentation Standards Reminders:

- Use the function header (`head_fx`) for each user-defined function in your program. You will list and comment each parameter to your user-defined functions in the function header file.
- Place **a single space** between all operators and operands.
- Comment all variables to the right of each declaration. Declare only one variable per line.
  - **Note that the program in the text uses a single line comment to indicate the start of the local declaration and executable statement sections of the** `main` **function.** At no point during the semester should these two sections ever overlap. You may consider adopting this habit of commenting the start of each section to help you avoid this mistake.
  - All variables must be given a local scope, global variables are not permitted.
- Select meaningful identifiers (names) for all variables, functions, and symbolic constants in your program.
- Your program must accept input and produce output in the same manner demonstrated in the example executions.
- Indent all code found within the `main` and all user-defined functions exactly two spaces.
  - Code found within selection constructs must be indented exactly two additional spaces. See the notes and text for the indenting requirements with the `switch` construct.
- Do not single (or double) space the entire program, use blank lines when appropriate.

Academic Integrity Reminder:

- Please review the policies of the course as they relate to academic integrity. The assignment your group submits should be your own original work. You should be consulting only your assigned partners and course staff regarding your specific algorithm for assistance.

When you submit... only the last attempt of a submission is kept for grading. All other submissions (**from the same user**) are over-written and cannot be recovered. You may make multiple submissions but only the last attempt is retained and graded.

- Verify in the e-mail sent to you, and all partners listed in the lab header, that you have submitted the correct file, to the correct assignment (`lab05`), and to the correct lab section. Forwarding course e-mails from Purdue to external e-mail services may result in the mail being undelivered or being identified as spam.
- Leave time prior to the due date to seek assistance or to resolve group concerns should you experience difficulties completing or submitting this assignment.
- All attempts to submit via a method other than through the `sage` server as set up during the first week of the semester will be denied consideration.

Assignment deadlines... are firm and the electronic submission will disable promptly as advertised. We can only grade what you submit as expected (including the correct assignment and lab section) prior to the assignment deadline.

All course standards are available on Blackboard.
Please review this document before submitting this assignment!