**IT 1090C Computer Programming I   
IT 6090C Java Programming**

**Prof. Tom Wulf**

# Lab 05 Conditionals

**20 pts (3 gr or extra credit pts)**

# Learning Goals:

* **Conditional Structures in Java if, if else, cascaded if, nested if, switch**
* **Testing. Make sure your programs run by testing them. Correct them if they do not.**

This is an individual lab. Complete each of the programs here. (I may do the first with you in class.) Note that these are programs that we previously did as pseudo code. Copy or re-write the pseudo code as single line // java comments as an outline for your program. Create a separate IntelliJ java project for each program using the names I specified.  
  
Don’t forget to code the pseudo code for your program first within the java main() section as single line comments **// comment…..** Then code the java statements for the pseudo code. In each task, use a test suite that covers each of the logical paths the program can take. e.g. (for the first program) user enters an item less than $100, or user enters one > $100 Your screen shots should support that you did this testing! (Provide shots for each case, for each task.)  
  
Simulating Input: since we don’t know how to do java input yet, we will simulate it. Just set the variable to a value as if you had input it. Change your code and rerun it to test multiple values. In most cases, the testing just requires you to confirm that the code works when the input is correct and when it is not.

If your instructor directed you to use GitHub create a separate repo for each program using the name I indicated.

# Mini lecture:

**if (CONDITION)  
{**

**Code statements in this block execute if the CONDITION is true**

**}** // this is the end of the block like the endIf in our pseudo code

And here is the if..else:

**if (CONDITION)  
{**

**Code statements in this block execute if the CONDITION is true**

**}**   
**else**

**{  
 Code statements in this block execute if the CONDITION is false  
}** // this is the end of the block like the endIf in our pseudo code

**Cascaded if:**

**if(CONDITION)  
{**

**}  
else if (CONDITON)  
{**

**}  
// more else if (CONDITON) blocks go here  
else // no condition a default when all the other if tests fail  
{**

**}**

**As you test each program, you have to run it multiple times. Instead of screen shots, just copy the output window from IntelliJ into this doc for EACH test run. There will be several for each program. Please make sure that your output is readable, that’s what we use to grade your work.**

# Task 1 (5 pts):

**Project name: Lab\_05\_01\_ShipCostCalculator  
  
An application program where the user enters the price of an item and the program computes shipping costs. If the item price is $100 or more, then shipping is free otherwise it is 2% of the price. The program should output the shipping cost and the total price.**

**Test runs: (insert the output widow copies here for the test runs)  
- valid input less than 100  
  
- valid input greater than 100**

**Graphical user interface, text

Description automatically generated  
  
If you were directed to use GitHub, be sure to create the repo and commit and push it.**

Task 2 (5 pts):   
**Project name: Lab\_05\_02\_BirthMonth  
  
A program that asks the user to enter their birth month (integer 1 – 12 inclusive). If the user enters a value in range, the program echoes the input (“Your birth month is: N”) If the value is not in the range it outputs an error msg (“You entered an incorrect month value: N”). Here N should be the value they entered.**

**Test runs: (insert the output widow copies here for the test runs**

**Text

Description automatically generated**

**- valid input in range 1 - 12  
  
- invalid input out of range**

**If you were directed to use GitHub, be sure to create the repo and commit and push it.**

Task 3 (5 pts):   
**Project name: Lab\_05\_03\_PartyAffiliation  
  
(This task uses Strings:) A program that prompts the user for their party affiliation (Democrat, Republican, or Independent) and responds accordingly with a Donkey, Elephant, Person, or Other. (i.e. “You get a Democratic Donkey.”) Notes: create a menu so the user chooses D, R, or I and assume that any other choice will be Other. Tests: just the four options D, R, I, Other. Use as cascaded if structure not separate if statements!**

**\\**

**Test runs: (insert the output widow copies here for the 4 test runs)**

**Text

Description automatically generated**

**If you were directed to use GitHub, be sure to create the repo and commit and push it.**

Task 4 (5 pts):

**Project name: Lab\_05\_04\_TheaterKiosk**

**As people pass through an entry kiosk at the theater, they are prompted to enter their age. If they are 21 or older, they get a paper wrist band. Code a logic program that asks the user to enter their age and then if they are 21 or over displays a message that they get a wrist band. (Note that the program does nothing if they are not 21 or over…)  
  
Test runs: (insert the output widow copies here for the 2 test runs)**

**If you were directed to use GitHub, be sure to create the repo and commit and push it.**

**Graphical user interface, text

Description automatically generated**

Task 5 (3 pts Extra or Graduate Credit):

**Project name: Lab\_05\_05\_NumCompare**  
**A program that takes two numbers as inputs from the user and compares them. It indicates if they are equal or if they are not indicates the one that is less. This program should be bullet-proofed. If either input is not a number, the program should indicate that the user should try again and terminate.  
  
EMBED SCREEN SHOTS OF NETBEANS HERE SHOWING YOUR PROGRAM RUN:**

**If you were directed to use GitHub, be sure to create the repo and commit and push it.**

**Text

Description automatically generated**

# Submitting your work:

Save or rename this word docx file as **Lastname\_Firstname\_Lab\_05.docx** using your name.

Everyone submits this file.

If you are using GitHub:

submit an additional file with the links to your program repos.

If you are not using GitHub:

create a new compressed .zip archive folder. (Don’t give me any other type of archive!) called **Lastname\_Firstname\_Lab\_05.zip** using your name.

Place each of your IntelliJ project folders in this archive. (**Do not individually zip the projects!**) Place an extra copy of this word doc file with your screen output shots within the archive as well. (Don’t put this in the individual project folders, put it in the top level in the archive so I can access it easily.) Submit this archive as well

If you did the extra credit option, which I urge you to try, submit the entire assignment again exactly the same way using the extra credit link.