**IT 1090C Computer Programming I**

**Lab 05 Conditionals**

**SPRING 2020**

**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 Netbeans project for each program using the names I specified. Create a single java main class for each of the programs using the filenames 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! (You don’t have to provide shots for each case though, just at least one 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 (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 Netbeans into the 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.**

1. **Task 1 (5 pts):**

**Project name: ShipCostCalculator  
Main file name: shipCost.java  
  
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**ant -f C:\\Users\\weydaej\\Documents\\NetBeansProjects\\ShipCostCalculator -Dnb.internal.action.name=run.single -Djavac.includes=shipCost.java -Drun.class=shipCost run-single

init:

Deleting: C:\Users\weydaej\Documents\NetBeansProjects\ShipCostCalculator\build\built-jar.properties

deps-jar:

Updating property file: C:\Users\weydaej\Documents\NetBeansProjects\ShipCostCalculator\build\built-jar.properties

Compiling 1 source file to C:\Users\weydaej\Documents\NetBeansProjects\ShipCostCalculator\build\classes

compile-single:

run-single:

Input the price of the item: 50.0

Shipping is $1.0 which gives you a total of $51.0 for this product.

BUILD SUCCESSFUL (total time: 0 seconds)

**- valid input greater than 100**ant -f C:\\Users\\weydaej\\Documents\\NetBeansProjects\\ShipCostCalculator -Dnb.internal.action.name=run.single -Djavac.includes=shipCost.java -Drun.class=shipCost run-single

init:

Deleting: C:\Users\weydaej\Documents\NetBeansProjects\ShipCostCalculator\build\built-jar.properties

deps-jar:

Updating property file: C:\Users\weydaej\Documents\NetBeansProjects\ShipCostCalculator\build\built-jar.properties

Compiling 1 source file to C:\Users\weydaej\Documents\NetBeansProjects\ShipCostCalculator\build\classes

compile-single:

run-single:

Input the price of the item: 150.0

Shipping is free for this product so your total is $150.0

BUILD SUCCESSFUL (total time: 0 seconds)

**Task 2 (5 pts):   
Project name: BirthMonth  
Main file name: birthMonth.java  
  
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  
  
- valid input in range 1 - 12**ant -f C:\\Users\\weydaej\\Documents\\NetBeansProjects\\BirthMonth -Dnb.internal.action.name=run.single -Djavac.includes=birthMonth.java -Drun.class=birthMonth run-single

init:

Deleting: C:\Users\weydaej\Documents\NetBeansProjects\BirthMonth\build\built-jar.properties

deps-jar:

Updating property file: C:\Users\weydaej\Documents\NetBeansProjects\BirthMonth\build\built-jar.properties

Compiling 1 source file to C:\Users\weydaej\Documents\NetBeansProjects\BirthMonth\build\classes

compile-single:

run-single:

Enter your birthmonth: 11

Your birth month is: 11

BUILD SUCCESSFUL (total time: 0 seconds)

**- invalid input out of range**

ant -f C:\\Users\\weydaej\\Documents\\NetBeansProjects\\BirthMonth -Dnb.internal.action.name=run.single -Djavac.includes=birthMonth.java -Drun.class=birthMonth run-single

init:

Deleting: C:\Users\weydaej\Documents\NetBeansProjects\BirthMonth\build\built-jar.properties

deps-jar:

Updating property file: C:\Users\weydaej\Documents\NetBeansProjects\BirthMonth\build\built-jar.properties

Compiling 1 source file to C:\Users\weydaej\Documents\NetBeansProjects\BirthMonth\build\classes

compile-single:

run-single:

Enter your birthmonth: 110

You entered an incorrect month value: 110

BUILD SUCCESSFUL (total time: 0 seconds)

1. **Task 3 (5 pts):   
   Project name: PartyAffiliation  
   Main file name: partyAffil.java  
     
   (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)**

* **D:**

ant -f C:\\Users\\weydaej\\Documents\\NetBeansProjects\\PartyAffiliation -Dnb.internal.action.name=run.single -Djavac.includes=partyAffil.java -Drun.class=partyAffil run-single

init:

Deleting: C:\Users\weydaej\Documents\NetBeansProjects\PartyAffiliation\build\built-jar.properties

deps-jar:

Updating property file: C:\Users\weydaej\Documents\NetBeansProjects\PartyAffiliation\build\built-jar.properties

Compiling 1 source file to C:\Users\weydaej\Documents\NetBeansProjects\PartyAffiliation\build\classes

compile-single:

run-single:

Please enter your party affiliation [Democrat (D), Republican (R), Independent (I)]: D

You get a Democratic Donkey.

BUILD SUCCESSFUL (total time: 0 seconds)

* **R:**

ant -f C:\\Users\\weydaej\\Documents\\NetBeansProjects\\PartyAffiliation -Dnb.internal.action.name=run.single -Djavac.includes=partyAffil.java -Drun.class=partyAffil run-single

init:

Deleting: C:\Users\weydaej\Documents\NetBeansProjects\PartyAffiliation\build\built-jar.properties

deps-jar:

Updating property file: C:\Users\weydaej\Documents\NetBeansProjects\PartyAffiliation\build\built-jar.properties

Compiling 1 source file to C:\Users\weydaej\Documents\NetBeansProjects\PartyAffiliation\build\classes

compile-single:

run-single:

Please enter your party affiliation [Democrat (D), Republican (R), Independent (I)]: R

You get a Republican Elephant.

BUILD SUCCESSFUL (total time: 0 seconds)

* **I:**

ant -f C:\\Users\\weydaej\\Documents\\NetBeansProjects\\PartyAffiliation -Dnb.internal.action.name=run.single -Djavac.includes=partyAffil.java -Drun.class=partyAffil run-single

init:

Deleting: C:\Users\weydaej\Documents\NetBeansProjects\PartyAffiliation\build\built-jar.properties

deps-jar:

Updating property file: C:\Users\weydaej\Documents\NetBeansProjects\PartyAffiliation\build\built-jar.properties

Compiling 1 source file to C:\Users\weydaej\Documents\NetBeansProjects\PartyAffiliation\build\classes

compile-single:

run-single:

Please enter your party affiliation [Democrat (D), Republican (R), Independent (I)]: I

You get an Independent Man.

BUILD SUCCESSFUL (total time: 0 seconds)

* **Other:**

ant -f C:\\Users\\weydaej\\Documents\\NetBeansProjects\\PartyAffiliation -Dnb.internal.action.name=run.single -Djavac.includes=partyAffil.java -Drun.class=partyAffil run-single

init:

Deleting: C:\Users\weydaej\Documents\NetBeansProjects\PartyAffiliation\build\built-jar.properties

deps-jar:

Updating property file: C:\Users\weydaej\Documents\NetBeansProjects\PartyAffiliation\build\built-jar.properties

Compiling 1 source file to C:\Users\weydaej\Documents\NetBeansProjects\PartyAffiliation\build\classes

compile-single:

run-single:

Please enter your party affiliation [Democrat (D), Republican (R), Independent (I)]: F

You did not enter a valid option.

BUILD SUCCESSFUL (total time: 0 seconds)

**4. Task 4 (5 pts): 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…)**

* **21+**

ant -f C:\\Users\\weydaej\\Documents\\NetBeansProjects\\TheaterAge -Dnb.internal.action.name=run.single -Djavac.includes=theaterAge.java -Drun.class=theaterAge run-single

init:

Deleting: C:\Users\weydaej\Documents\NetBeansProjects\TheaterAge\build\built-jar.properties

deps-jar:

Updating property file: C:\Users\weydaej\Documents\NetBeansProjects\TheaterAge\build\built-jar.properties

Compiling 1 source file to C:\Users\weydaej\Documents\NetBeansProjects\TheaterAge\build\classes

compile-single:

run-single:

Enter your age: 21

You get a paper wristband!

BUILD SUCCESSFUL (total time: 0 seconds)

* **Under 21**

ant -f C:\\Users\\weydaej\\Documents\\NetBeansProjects\\TheaterAge -Dnb.internal.action.name=run.single -Djavac.includes=theaterAge.java -Drun.class=theaterAge run-single

init:

Deleting: C:\Users\weydaej\Documents\NetBeansProjects\TheaterAge\build\built-jar.properties

deps-jar:

Updating property file: C:\Users\weydaej\Documents\NetBeansProjects\TheaterAge\build\built-jar.properties

Compiling 1 source file to C:\Users\weydaej\Documents\NetBeansProjects\TheaterAge\build\classes

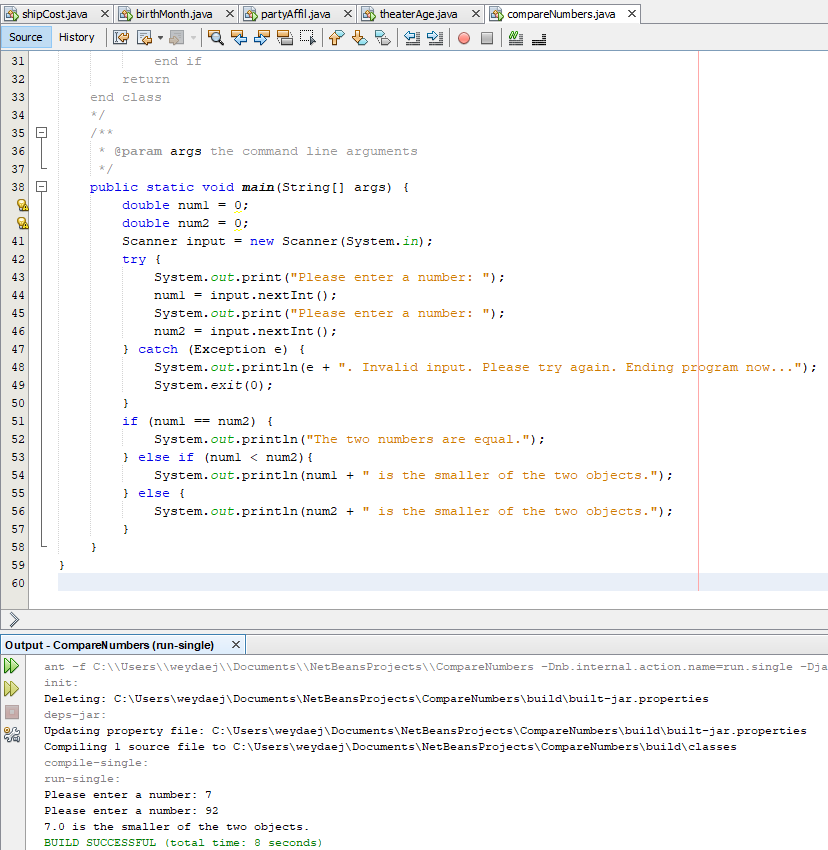
compile-single:

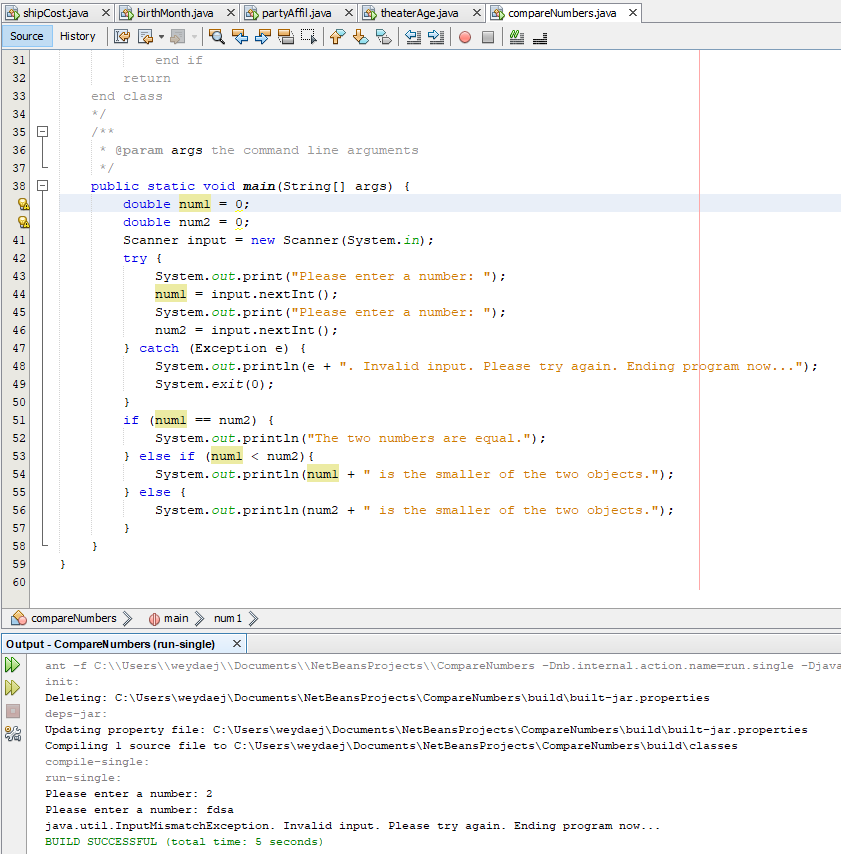
run-single:

Enter your age: 17

BUILD SUCCESSFUL (total time: 0 seconds)

1. **Task 5 (3 pts Extra or Graduate Credit):   
   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:**

**Submitting your work:**

Save or rename this word docx file as **Lastname\_Firstname\_Lab\_05.docx** using your name  
  
Create a new compressed .zip archive folder. (Don’t give me any other type of archive, it will be returned to you ungraded!) called **Lastname\_Firstname\_Lab\_05.zip** using your name.

Place each of your Netbeans project folders in this archive. (**Do not individually zip the projects!**) Place this word doc file with your screen output shots in 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.)   
  
(When I open you archive I should see each of your project folders in it and this file.)

Use the Canvas Assignment mechanism to submit a copy of your complete archive **as well as a separate copy of your Word docx**. If you submit the word docx separately, I can put my feedback comments in it directly for you.  
  
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.