

IST 256
Lab Week 1 – Wednesday, August 31, 2011

I. Computing Football Scores

1. Open NetBeans and create a new project

Go to the Program menu and find NetBeans. Start a new project in NetBeans and create a Java GUI Application called FootballScores with a new JFrame form called FootballScoresUI. (Don't forget to make FootballScoresUI the main class.)

2. Create the Form Design.

- **Add four labels to the form** by dragging the objects from the Palette pane
 - Change the text of the labels to say something like “Type number of touchdowns”, “Type number of extra points”, “Type number of field goals”, and “Resulting Score”.
- **Add four textfields to the form** by dragging the objects from the Palette pane.
 - Initialize the textfields to have 0 in them.
- Add a button to the form by dragging it from the Palette pane.
 - Change the text property of the button to be an instruction to the user, e.g. “Compute Scores”.
- **To add a program to be executed** when the user clicks the button, right click on the button on your form and select Event -> Action -> actionPerformed.

3. Write the program

The problem statement is:

Given that number of touchdowns, number of extra points and number of field goals in a football game, compute the total score of the game.

The score for touchdowns is the number of touchdowns times 6.

The score for extra points is the number of extra points times 1.

The score for field goals is the number of field goals times 3.

The total score is the sum of the touchdown score, the extra points score and the field goals score.

Write the program developed in class to solve this problem right after the comment with //TODO:

- 4. **Run this program** by the little green triangle (play) on the top bar.
 - When the program executes, it should bring up the user interface form, and you should type in three integers into the first three text fields. Then click on the button to get the sum.

5. Test your program with three sets of numbers:

- a. **Write down what integers you typed in and what the results are here.**

Touchdowns	Extra Point	Field Goals	Resulting Score

6. Questions about the program

Give an example of a declaration statement in the program that you wrote:

In this declaration, what is (one of) the variable name?

In this declaration, what is the type?

Give an example of an assignment statement in the program that you wrote:

In this statement, what is the variable being assigned to?

II. Writing a second GUI program – Try this one on your own, but don't hesitate to ask for help.

1. Create a new project

Follow the directions for starting a new project in NetBeans and create a Java GUI Application called ConvertMeters.

2. Create the Form Design.

- **Add two labels to the form** by dragging the objects from the Palette pane (upper right hand side).
 - One label should say “Type number in meters”
 - The second label should say “Result in feet”
- **Add two textfields to the form** beside the labels by dragging the objects from the Palette pane.
- **Add a button** to the form by dragging it from the Palette pane.
 - Change the text property of the button to be something like “Convert Number”
- **To add a program to be executed** when the user clicks the button, right click on the button on your form and select Event -> Action -> actionPerformed .

3. Write the program

Problem Statement:

Given a number in meters, convert the number to the equivalent number of feet.
The number of feet is 3.28084 times the number of meters.

Write a program to solve this problem, putting it right after the //TODO comment in the button event function.

4. Run the program

Test the program with 3 numbers. Show the numbers that you tried and the results here:

Number in Meters	Number in Feet

In order to get credit for this lab, parts I and II must be completed by Wednesday, Sept. 7.

To hand in the lab, copy the program that you wrote to convert meter to feet to a text file, only copying the part of the code that you wrote after the //TODO comment in the button event function.

Print the copied code onto a sheet of paper and attach it to the lab sheet when you hand it in.

III. Testing Arithmetic Operators – This program can also be completed by Wednesday, Sept 7, for extra lab credit. (Just hand in these sheets with the results of your arithmetic tests, not the code.)

1. Make the project and the form for the user interface.

For this application, start a new project in NetBeans and create a Java GUI Application called TestArith with a form TestArithUI. Remember the three main steps: create the new project, create the new JFrameForm and name it TestArithUI, and set the Properties under Run to have the form as the main class.

This application will be used to test the results of a number of arithmetic operators. The form will be created once with a test program. Then the test program will be continually modified with new operators to test them.

The form should look something like this with three labels, three textfields and a button:

First	__0_____	Compute Integers
Second	__0_____	(Button)
Result	__0_____	

Right click on the button and select Event -> Action -> actionPerformed.

2. Write the code for testing Integer arithmetic in the Button

For the Compute Integers button, write the following program

1. Declare three variables named first, second, and result as type int
2. Get input values for first and second from jTextField1 and jTextField2.
3. Add the two numbers and assign the sum to the variable result.
4. Display the result value in jTextField3.

Test your program with a pair of integers to make sure that it is adding correctly.

3. Test for Input Errors:

Continue the testing to see what happens if you don't put integers in the TextFields. Now run the program and replace the 0's in the TextFields without typing any numbers. Click on the button. What happens?

Now run the program again, and instead of typing a valid integer, type a number with a decimal point – for example, type 3.6 in the first TextField and type 2.2 in the second TextField. What happens?

What happens if you type the string Hello in the first TextField?

4. Test the Integer Operators +, -, *, /, % (addition, subtraction, multiplication, integer division and remainder) :

For the remaining tests, put the operator into the program line that computes with the first and second number and assigns it to result. Then run the program with two different sets of numbers (as described below) and write your results on this paper.

For each operator, in the first set of numbers, the first number should be bigger than the second, but in the second set of numbers, the first number should be smaller than the second. Negative numbers can be used, except for %. The first two sets of numbers are filled in for you – you can pick the remaining ones to test.

Integer Operator	First Integer	Second Integer	Result
+	9	5	
+	-5	9	
-			
-			
*			
*			
/			
/			
%			
%			