Assignment 2

~Kals

General Tips

• "Any fool can write code that a computer can understand. Good programmers write code that humans can understand." --- Martin Fowler

• Variable names – Use meaningful variable names
• Indentation – Use spaces(not tabs). Show the structure of code flow and scope clearly.
• Comments – Use them! But, avoid trailing comments

• Refer these nice style guides:
  – http://www.cwu.edu/~gellenbe/javastyle/index.html
  – http://www.javaranch.com/style.jsp
Problem 1

- There are no java files to be turned in.
- You will get 6 points for each correct answer.
- Although only answers are expected, you may optionally provide explanation as to how you arrived at the answer. Your explanation can fetch you partial credit even if your answer is wrong.
- Refer operator precedence table in Week 2 – Slide 9

Problem 2

- Make sure to use correct class name – Triangle.java
- Start with the code – Triangle.java. Do not change the contents of public static void main method. Add code inside getPerimeter method only.
- 10 points will be deducted for compilation failure. So make sure your code compiles before turning it in.
- Inside the getPerimeter you need to do two things:
  1) Check that the given 3 values can form a triangle.
  2) If it can form a valid triangle then return the perimeter value otherwise return 0.
- Do not use any loops(e.g. for, while) or library classes (other than Scanner).
Problem 3

- Make sure to use correct class name – DayOfYear.java
- Start with the code – DayOfYear.java. Do not change the contents of public static void main method. Add code inside getDayOfYear method only.
- Do NOT use loops, arrays or advance java libraries (e.g. date class.)
- Inside getDayOfYear do these things:
  1) Check that if it’s a valid year – You can use “If” condition for this. Make sure to catch all kind of wrong input values. These are the conditions you must ensure to check –
     a) year value should be between 1600 to 2400. E.g., ‘12 31 2500’ should be caught as an invalid date entry.
     b) month value should be between 1 to 12. E.g. ‘13 31 1999’ should be caught as an invalid date entry.
     c) day value should be correct depending on which month it is. E.g. ‘02 29 2010’ should be caught as an invalid date entry.
  2) Check if given year is leap or not:
     If a year is divisible by 400 then it is a leap year. E.g. 1600, 2000, 2400 are leap years. If its not divisible by 4 and not divisible by 100 then it is a leap year. E.g. 1996 is a leap year. 2100 is not a leap year.
  3) Calculate day of the year:
     You can use nested if loops or better still you can use switch cases. Do not excessively repeat the code.
- Your code should be readable (good amount of comments and proper indentation). 5 points will be deducted for extreme cases of poor readability

Problem 3 (continued)

If an invalid date entry is caught, then use System.err.println() to display the error message and System.exit(1) to terminate program execution.

2) Check if given year is leap or not:
   If a year is divisible by 400 then it is a leap year. E.g. 1600, 2000, 2400 are leap years. If its not divisible by 400 then check if it is divisible by 4 and not divisible by 100 then it is a leap year. E.g. 1996 is a leap year. 2100 is not a leap year.

3) Calculate day of the year:
   You can use nested if loops or better still you can use switch cases. Do not excessively repeat the code.

- Your code should be readable (good amount of comments and proper indentation). 5 points will be deducted for extreme cases of poor readability
Problem 3 (continued)

• Sample output 1:
Enter a date as month(1-12), day(1-31), year (1600-2400): 12 30 1999
12/30/1999 is day number 364 of the year
Java: Dec 30, 1999 is day number 364 of the year

• Notice that the last line (highlighted in red) shows you the correct ‘day of year’ value with which you can manually compare and check your program’s answer (highlighted in blue). The text highlighted in green is the user input.

• Sample output 2:
Enter a date as month(1-12), day(1-31), year (1600-2400): 31 30 1999
31/30/1999 is an illegal date

Useful URLs

• Class website: http://www.cs.uoregon.edu/classes/10W/cis210/
• Kals’ notes: http://ix.cs.uoregon.edu/~kals/cis210/
• Homework Help Sessions (Blue and Cream colored ones): http://www.cs.uoregon.edu/classes/10W/cis210/labsched.html
• Class Slides & Examples: http://www.cs.uoregon.edu/classes/10W/cis210/notes/