Project [1]: Formatted input/output

Due Thursday, 9/13/2018, 11:59 pm

Project Goals

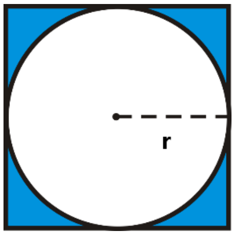
The goal of this project is to:

1. Get students familiar with the `printf` and `scanf` functions

Important Notes:

1. **Formatting:** Make sure that you follow the precise recommendations for the output content and formatting: for example, do not change the text in the first problem from ‶`Enter the value of the radius: `″ to ‶`Enter the radius: `″. Your assignment will be auto-graded and any changes in formatting will result in a loss in the grade.
2. **Comments:** Header comments are required on all files and recommended for the rest of the program. Points will be deducted if no header comments are included.

Problem 1



Write a program that will ask the user for the radius of a circle that has been inscribed in a square. Then your program will calculate and print out: the diameter of the circle, the circumference of the circle, the area of the circle, the perimeter of the square, the area of the square, and the difference between the area of the square and the area of the circle.

The program should function as follows (items underlined are to be entered by the user):

```
This program computes values related to an inscribed circle.
Enter the value of the radius: 7
The diameter of the circle is: 14
The circumference of the circle is: 43.960
The area of the circle is: 153.860
The perimeter of the square is: 56
The area of the square is: 196
The difference between the area of the square and the circle is: 42.140
```

Notes:

- You can assume that the radius will be a positive, whole number
- Please use `3.14f` as your value for pi (This is a good place to use `#define`)
- The circumference, area of the circle, and difference in areas should output only 3 digits after the decimal place

Save your program as `circle.c`

Problem 2

Write a program that takes as input four dates in the format `dd/mm/yyyy` and prints them out in a tabular format.

The program should function as follows (items underlined are to be entered by the user):

```
Enter date 1 (dd/mm/yyyy): 05/03/2017
Enter date 2 (dd/mm/yyyy): 21/03/2010
Enter date 3 (dd/mm/yyyy): 07/04/1776
Enter date 4 (dd/mm/yyyy): 11/12/0200

Year  Month Day
2017   3    5
2010   3    21
1776   4    7
 200  12    11
```

The year, month and day columns should be separated by tabs (\t) (this also refers to the words Year, Month and Day from the header of the table). The year column should occupy 4 spaces and the values should be right justified. The month column should occupy 2 spaces and the values should be right justified. The day column should occupy 2 spaces and the values should be left justified.

Note:

- You should use `\t` to print a tab instead of hitting the tab button on the keyboard

Save your program as `date.c`

Challenge for problem 2 (10 extra credit points):
Modify your previous program so that if the value of the year is less than 4 digits, add a 0 in front of the year to make the year have 4 digits. If the values of the month and day are less than 10 (meaning they have just one digit), the values of month/day will always have two digits displayed.

The program should function as follows (items underlined are to be entered by the user):

```
Enter date 1 (dd/mm/yyyy): 05/03/2017
Enter date 2 (dd/mm/yyyy): 21/03/2010
Enter date 3 (dd/mm/yyyy): 07/04/1776
Enter date 4 (dd/mm/yyyy): 11/12/0200

Year  Month Day
2017  03    05
2010  03    21
1776  04    07
0200  12    11
```

Save your challenge separately as `date_c.c`

Grading Rubric

Grading will be done for each problem as follows:

Correctly-named file	5%
Header comment	2%
Program compiles	5%
Correctly-reading data from terminal	18%
Correct result printed	20%

Submission details

To submit your project, you will have to use the submission script. You do this by either:

1. Working on an ECC machine
2. Working on the provided VMware
3. Secure Copying your files (See Mac Support for information)

To Submit your project:

- Have a directory called “project1”
- Save your *.c files in that directory
- To submit: **(don’t type the ‘>’ symbols)**

```
> cd project1
> submit
```

The submission script copies all files in the current directory to our directory. You may submit as many times as you like before the deadline, we only keep the last submission.

Academic Honesty

Academic dishonesty is against university as well as the system community standards. Academic dishonesty includes, but is not limited to, the following:

Plagiarism: defined as submitting the language, ideas, thoughts or work of another as one's own; or assisting in the act of plagiarism by allowing one's work to be used in this fashion.

Cheating: defined as (1) obtaining or providing unauthorized information during an examination through verbal, visual or unauthorized use of books, notes, text and other materials; (2) obtaining or providing information concerning all or part of an examination prior to that examination; (3) taking an examination for another student, or arranging for another person to take an exam in one's place; (4) altering or changing test answers after submittal for grading, grades after grades have been awarded, or other academic records once these are official.

Cheating, plagiarism or otherwise obtaining grades under false pretenses" constitute academic dishonesty according to the code of this university. Academic dishonesty will not be tolerated and penalties can include cancelling a student's enrolment without a grade, giving an F for the course, or for the assignment. For more details, see the University of Nevada, Reno General Catalog.