CS507 Computing Foundation for Computational Science HW3

Shell Programming

Min Long

Instruction

Due time: Due: 10/13/2022, 23:59
Submission command: `submit minlong CS507 HW3`

Instruction to electronic submission: http://cs.boisestate.edu/∼cs221/SubmissionProcedure.html

- The written assignment can be done in a pure text format (*.txt, for example, problem1.txt) on Onyx.
- The programming assignment should be presented with source codes.
- Each problem should have its own working directory, such as HW2/prob1, HW2/prob2 ... For example, the following table shows the structure of HW1 from the user "student1" and how to submit HW to us through Onyx

```
[student1@onyx:HW1]$ ls -l
drwxr-x---. 2 student1 Students  13 Sep 19  2021 prob1
drwxr-xr-x. 3 student1 Students  14 Sep 19  2021 prob2
[student1@onyx:HW1]$ cd prob1/
[minlong@onyx:prob1]$ ls
-rw-r-----. 1 student1 Students 943 Sep 19  2021 problem1.txt
$ cd ..
[student1@onyx:HW1]$ pwd
/home/student1/CS507/HW1
[student1@onyx:HW1]$ submit minlong CS507 HW1
```

Listing 1: A sample structure of homework and submission procedure.

- Your source codes (if any) must compile and run on Onyx.
- Documentation is important and proper comments are expected in your source code.
  - comments giving description of: purpose, parameters, and return value if applicable
  - other comments where clarification of source code is needed
  - proper and consistent indentation
  - proper structure and modularity

Don't ask us or your classmates directly for solutions (it happened); just try as much as possible. Be patient and enjoy coding!

Programming Problems

1. (5 pts) **Order check**. Compose a program that accepts three floats x, y, and z as command-line arguments and writes True if the values are strictly ascending or descending ($x < y < z$ or $x > y > z$), and False otherwise.

2. (5 pts) **Grade circle** Compose a program that accepts four floats as command-line arguments—x1, y1, x2, and y2—(the latitude and longitude, in degrees, of two points on the earth) and writes the great-circle distance between them. The great-circle distance d (in nautical miles) is given by the following equation:

$$d = 60 \arccos[\sin(x_1)\sin(x_2) + \cos(x_1)\cos(x_2)\cos(y_1 - y_2)]$$

   Use your program to compute the great-circle distance between Paris (48.87° N and -2.33° W) and San Francisco (37.8° N and 122.4° W).
   Hint:

   - You need to `import math` module into your code for mathematical functions like math.sin(), math.cos(), math.acos().
   - The equation uses degrees, whereas Python's trigonometric functions use radians. Use math.radians() and math.degrees() to convert between the two.
   - Read references on the math package at: https://docs.python.org/3/library/math.html

3. (10 pts) **Color conversion**. Several different formats are used to represent color. For example, the primary format for LCD displays, digital cameras, and web pages, known as the RGB format, specifies the level of red (R), green (G), and blue (B) on an integer scale from 0 to 255. The primary format for publishing books and magazines, known as the CMYK format, specifies the level of cyan (C), magenta (M), yellow (Y), and black (K) on a real scale from 0.0 to 1.0. Compose a program that converts RGB to CMYK. Accept three integers—r, g, and b—from the command line and write the equivalent CMYK values. If the RGB values are all 0, then the CMY values are all 0 and the K value is 1; otherwise, use these formulas:

$$\begin{aligned} w &= \max(r/255, g/255, b/255) \\ c &= (w - (r/255))/w \\ m &= (w - (g/255))/w \\ y &= (w - (b/255))/w \\ k &= 1 - w \end{aligned}$$

4. (10pts) **Trigonometric functions.** Compose 2 programs that compute the sine and cosine functions using their Taylor series expansions

$$\sin x = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+1)!} x^{2n+1} = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \ldots$$
$$\cos x = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n)!} x^{2n} = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \ldots$$

   Hint:

- Taylor's series of a function is an infinite sum of terms that are expressed in terms of the function's derivatives at a single point.
- In numerical calculation, we certainly can't compute **infinite** sum of terms but has to stop when the new term to be added becomes really small, say $term \leq 1e - 15$. You can use it for your boolean expression. You can see "EPSILON" in our example of Newton's method as a reference.

5. (10pts) **Guess game in Python**. Let's redo our previous problem using Python. Use `import random` in your code to generate a random number in a range [0, 50]. The code should be able to randomly set up a number as an answer and read input from the command-line argument. When the user's guess number doesn't match the answer, it will prompt the user to input another guess till they match.

   Hint:

   - You can check the definition of the following functions to generate a random integer: random.randrange(), random.randint() at https://docs.python.org/3/library/random.html

```
$ python numbergame.py
The magic number is between 0 and 50.
Make your guess:28
28 is too low
The magic number is between 29 and 50.
Make your guess:30
30 is too low
The magic number is between 31 and 50.
Make your guess:50
50 is too high
The magic number is between 31 and 50.
Make your guess:40
40 is too low
The magic number is between 41 and 50.
Make your guess:45
45 is too low
The magic number is between 46 and 50.
Make your guess:48
48 is too high
The magic number is between 46 and 47.
Make your guess:47
47 is too high
The magic number is between 46 and 47.
Make your guess:46
You got it in 8 guesses!
```

Listing 2: Sample output.

   Hint: You may use Python's built-in input() function as shown in class. Or you can take a look at the examples here: https://www.geeksforgeeks.org/python-input-function/

6. (10 pts) **Rydberg formula.** In 1888 Johannes Rydberg published his famous formula for the wavelengths λ for the emission lines of the hydrogen atom:

$$\frac{1}{\lambda}\left(\frac{1}{m^2} - \frac{1}{n^2}\right),$$

   where R is the Rydberg constant $R = 1.097 \times 10^{-2}\ \text{nm}^{-1}$ and m and n are positive integers. For a given value of m, the wavelengths λ given by this formula for all $n > m$ forms a series. The firms 3 such series, from m=1,2, and 3 are known as the Lyman, Balmer and Paschen series after their respective discoverers. Write a Python program that prints out the wavelengths of the first five lines in each of these three series.

```
$ python Rydberg.py
Series for m= 1
   121.5436037678517  nm
   102.55241567912488  nm
   97.23488301428137  nm
   94.95594044363415  nm
   93.76220862091418  nm
Series for m= 2
   656.3354603463993  nm
   486.1744150714068  nm
   434.084299170899  nm
   410.2096627164995  nm
   397.04243897498225  nm
Series for m= 3
   1875.2441724182836  nm
   1281.9051959890612  nm
   1093.8924339106654  nm
   1005.013673655424  nm
   954.6697605038536  nm
```

Listing 3: Sample output.