Assignment #1

Math 1800: Mathematical Programming in Python

Instructions: Choose 3 of these problems. Some are hard, some easy. Some are boring, and some are interesting. As your answer, submit Python text files, with the extension `.py`. Each file should include your name and the problem number.

- *Problem 1.0:* Try to find the square root of π using the following idea:

```
Initialize x to 10.
Replace x by the average of x and pi/x.
Repeat this process several times.
Print the result.
For comparison, print the value of sqrt(pi).
```

- *Problem 1.1:* A brick whose mass is 3.5 kilograms is converted into energy according to Einstein's formula: $E = mc^2$, where the speed of light $c = 299,792,458$ meters/second. The unit of energy is the `Joule` $= 1\frac{kg\,m^2}{s^2}$. Using this data, how much energy in Joules is released by this process?

- *Problem 1.2:* The Body Mass Index is the ratio $BMI = \frac{w}{h^2}$ where w is a person's weight in kilograms, and h is the height in meters. Sandy's weight is 180 pounds, and height is 5'8". Dale's weight is 120 pounds, and height is 5'3". What are Sandy and Dale's BMIs? Note that 1 pound is approximately 0.45 kilograms, and 1 inch is approximately 0.025 meters, and your doctor would prefer your BMI to be between 20 and 25.

- *Problem 1.3:* In combinatorics, an important quantity is known as "n-choose-k", counting the number of ways that k objects can be chosen from a set of n. Sometimes this is represented by C(n,k) or $\binom{n}{k}$. A formula for this value is $C(n,k) = \frac{n!}{(n-k)!k!}$ where $n!$ is the factorial product $n! = 1 \cdot 2 \cdot 3 ... \cdot n$. There is a factorial function available in the Python math library. Use it to evaluate C(6,2), C(60,20), and C(600,200). You can get an exact answer if you use the right kind of division.

- *Problem 1.4:* A fractal image known as the Mandelbrot set can be defined as all the points c in the complex plane which stay within the circle of radius 2 under the repeated operation

$$z_0 = c$$
$$z_1 =\rightarrow z_0^2 + c$$
$$z_2 =\rightarrow z_1^2 + c...$$

  and so on. Try two different starting points, $c_1 = -0.80 + 0.1i$ and $c_2 = -0.75 + 0.4i$. For each starting point, repeat about 10 steps of the iteration, and print the absolute value of z at each step. For which starting value do the iterates stay small, and for which do the iterates begin to grow large?

- *Problem 1.5:* As mentioned in the Hill textbook, the Greek mathematician Heron had a formula for computing the area of an triangle, whose sides were of length a, b and c:

$$Area = \sqrt{s\,(s-a)\,(s-b)\,(s-c)}$$

  where $s = \frac{1}{2}(a+b+c)$. Use Heron's formula to determine the area of a triangle whose sides are 10, 12, and 6. What goes wrong if you apply Heron's formula to a triangle whose sides are 5, 12, and 6?

- *Problem 1.6:* A standard die used in games is a cube with the numbers 1 through 6 printed on the faces in a particular order. If two adjacent sides have the values a and b, reading from left to right, the value on the top of the die must be

$$c = (3\,(a^3 b - a b^3))\,mod\,7$$

Look at the pair of dice in the illustration.

In this image, we see one die for which $a = 5, b = 6, c = 4$, and one with $a = 5, b = 1, c = 3$. Note that, for the first die,

$$(3 * (5^3 * 6 - 5 * 6^3)) \text{ mod } 7 = -990 \text{ mod } 7 = 4$$

and the number on the top face of the first die is indeed 4. You can carry out the calculation for the second die to check how this works. Now write a Python program which evaluates the formula to determine the value of c if $a = 2, b = 6$. and again if $a = 3, b = 5$.

- *Problem 1.7:* A sheet of paper has a thickness $t = 0.1$ mm. The distance from the Earth to the Moon is $D = 384,400$ km. A millimeter is equal to one millionth of a kilometer. How many times would the paper have to folded so that it would be thick enough to reach the moon? You can solve this using logarithms.

$$D = 2^k * t/1000000$$

Solve for k, then round k up to an integer. Once we learn about the `while` statement, there is another way to do this by repeatedly doubling the size of t until we just pass the value D.

- *Problem 1.8:* An oblate spheroid is the technical term for a sphere that has been slightly squashed along the poles to make a sort of ellipsoid. The earth can be modeled this way, with a semi-major axis $a = 6,378,137.0$ m, and semi-minor axis of $c = 6,356,752.314245$ m. If the eccentricity e is defined by

$$e^2 = 1 - \frac{c^2}{a^2}$$

then the surface area of an oblate spheroid is

$$\text{area} = 2\pi a^2 (1 + \frac{1 - e^2}{e} \text{arctanh}(e))$$

According to this model, what is the estimated surface area of the Earth?

- *Problem 1.9:* The exponential function can be defined by the power series

$$\exp(x) = \sum_{k=0}^{\infty} \frac{x^k}{k!} = 1 + x + \frac{x^2}{2} + \frac{x^3}{6} + ...$$

Suppose we need the value of exp(4). We could make the following sequence of estimates:

$$\begin{aligned} \exp(4) &\approx 1 \\ &\approx 1 + 4 \\ &\approx 1 + 4 + \frac{4^2}{2} \end{aligned}$$

and so on. Start your Python calculation as follows:

```
est = 0
k = -1
```

Now add the k-th term by:

```
k = k + 1
est = est + x**k / factorial ( k )
print ( 'k = ', k, '  estimate = ', est )
```

By repeating this last set of commands, you can try to improve your estimate. The correct value is a bit more than 54. How many steps do you need to get these first two digits correct?