Due during the first lecture, on October 1st, 2020.

– – – – – – – – – – – – – ⋆ – – – – – – – – – – – – –

1. python, machine representation of numbers, etc

   (a) What is **integer division** in programming languages?
   How does python2.x differ from python3.x in this regard?
   If you ask python to calculate 1/3 and 1.0/3, would the answers differ?

   (b) Can the number 0.2 be represented exactly, in floating point binary representation?
   What about the number 0.25?
   Hint: The tutorial on the python documentation page
   `https://docs.python.org/3/tutorial/`
   has a section (section 15) on floating point arithmetic. It is very instructive to work through.

   (c) Consider the code

```
eps = 1.0
while 1.0 != 1.0 + eps:
    print 'eps value now:', eps
    eps = eps/2.0
print 'final eps:', eps
```

   Why does this while loop ever terminate?
   What is the value of eps at which the loop terminates?

   (d) What is the difference between **float** and **double** datatypes?
   (Also known as single-precision versus double-precision.)
   Does python know the difference?

   (e) Explain (or write the code) how to use python's **list comprehension** feature to generate a list containing the numbers

$$1, 4, 9, 16, \ldots, 400$$

2. Numerical integration.

   (a) The best-known **Newton-Cotes** formulae for numerical integration are the trapezoidal rule, Simpson's rule and Simpson's 3/8 rule. Explain the intuition behind any one of them (you can choose), including an appropriate figure.

   (Of course, you should make sure you know the intuition behind all of them.)

   (b) Numerical integration algorithms are usually used in **composite** form. E.g., the composite trapezoidal rule involves breaking up the integration region (say $[a, b]$) into N equal pieces, each of width $h = (b - a)/N$. The trapezoidal rule is then applied to each interval, and the results added up. Usually, if we say "trapezoidal rule", we refer to the composite trapezoidal rule. Similarly for the other rules.

   How does the error of the three rules (trapezoidal, Simpson's, and Simpson's 3/8) scale with N? (i.e., what is the leading dependence of the error on N, at large N?)

   (c) How do **Gaussian quadrature** rules differ from **Newton-Cotes** integration?