Heron’s Method 1

Thomas Schwarz, SJ

A simple program for quadratic equations

- Given a quadratic equation
  - $ax^2 + bx + c = 0$
- the solutions are
  - $\dfrac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

Quadratic Equation

- Plan:
  - Ask user to enter the three variables a, b, and c
    - But specify what is what through a print statement
  - Calculate the discriminant
    - $b^2 - 4ac$
  - And from there the two solutions
  - Print out the two solutions

Quadratic Equation

- Printing out the information and asking for the coefficients:

```
print("Solving a x^2 + b x + c")
a = float(input('Enter a: '))
b = float(input('Enter b: '))
c = float(input('Enter c: '))
```

- Now a, b, and c refer to the corresponding coefficient

Quadratic Equation

- Calculating the root of the discriminant
- Calculating the two solutions

```
discriminant_root = (b*b-4*a*c)**0.5
sol1 = (-b + discriminant_root)/(2*a)
sol2 = (-b - discriminant_root)/(2*a)
```

Quadratic Equation

- Alternatively, we can use the square-root function from the math module
  - This involves two steps
    - We need to ***import*** the math module
      - This is done by putting `import math` (traditionally at the top of the script)
    - Use the proper name, which is `math.sqrt( )`
      - `sqrt` is the name of the function
      - and we can find it in `math`

Quadratic Equation

- Program so far is

```
import math
print("Solving a x^2 + b x + c")
a = float(input('Enter a: '))
b = float(input('Enter b: '))
c = float(input('Enter c: '))

discriminant_root = math.sqrt(b*b-4*a*c)
sol1 = (-b + discriminant_root)/(2*a)
sol2 = (-b - discriminant_root)/(2*a)
```

Quadratic Equation

- Now we need to output the solutions
  - A little bit of text can be quite useful
    - In practice, you might be the only one to execute your script, but try understanding a bare-minimum program after a few weeks
  - And if others use your script, you want to be clear

Quadratic Equation

- We should print out an explanation

```
print('The solutions of the quadratic equation are')
print(sol1)
print('and')
print(sol2)
```

Heron's Method

- A useful way to calculate the square-root of a number
  - Also known as the Babylonian method
    - But named after an Alexandrian mathematician from ca 100 B.C.E.
  - Takes an approximation to the square root and returns a better one
  - Formula for square root of S:
    - If guess is x then a better guess is $x = \frac{1}{2}(x + \frac{S}{x})$

Heron's Method

- Example: $\sqrt{2}$
  - Initial guess is 1

$$1 \qquad 2 \qquad 3$$

  - Second guess is $\frac{1}{2}(1+\frac{1}{1}) = \frac{}{2}$
  - Third guess is

$$\frac{1}{2}(\frac{3}{2}+\frac{2}{\frac{3}{2}}) = \frac{1}{2}(\frac{3}{2}+\frac{4}{3}) = \frac{1}{2}(\frac{9+8}{6}) = \frac{17}{12}$$

Heron's Method

- To implement this:
  - We need to explain the purpose of the algorithm
  - We need to ask the user for the number S of which we want to calculate the square root
  - We then repeatedly apply the formula to the variable containing our initial guess
    - Which for simplicities sake we set to $\frac{S}{2}$

Heron’s Method

Heron’s Method

Heron’s Method

Heron’s Method

Heron’s Method

Heron’s Method

Heron’s Method