1 Simple Stuff

Prerequisites: control flow (branching, iteration), IO, arithmetic, atomic types.

1.1 The Good Old Days ⋆

Input: An integer 4.

Output: The word `"Elephant"`.

1.2 Equation of a Line ⋆

Input: Two integers k and b, $k \neq 0$.

Output: Such value x, that it satisfies the equation $kx + b = 0$.

1.3 Wait, what? ⋆⋆

Input: Two integers a and b.

Output: The product of a and b.

Note: You may not use the multiplication operation.

1.4 Late'o'clock ⋆⋆

Input: An integer $0 \leq h < 24$. Hours on a clock.

Note: Convert the given time h to the 12-hour clock format.

Output: First the time h in 12-hour clock format, then `"am"` or `"pm"` depending on the time.

1.5 Quadratic Equation ⋆⋆

Input: Three integers a, b and c.

Output: Find all values of x, such that $ax^2 + bx + c = 0$.

Note: If there are no possible values of x output `"NaN"` (not a number). The values should not be repeated.

1.6 Qubic Equation ⋆⋆⋆

Input: Four integers a, b, c and d.

Output: Find all values of x, such that $ax^3 + bx^2 + cx + d = 0$.

Note: If there are no possible values of x output `"NaN"` (not a number). The values should not be repeated.

Hint: use Cardano's formula.

1.7 Euclid Approves ⋆

Input: Two integers a and b, sides of a right angled triangle.

Output: The hypotenuse c of the aforementioned triangle.

1.8 Euclid Disapproves ⋆⋆

Input: Two integers a and b, sides of a right angled triangle and an integer angle θ (given in degrees) between them.

Output: The third side of the triangle.

Hint: You may use `import math` to get some functions you might want.

1.9 Everyone but Euclid Approves ⋆⋆⋆

Input: An integer n the amount of following lines, $3 \leq n \leq 100$. Each following line i contains a number $-100 \leq a_i \leq 100$, a component of the vector $\hat{v} = \{a_1, a_2, \ldots, a_n\}$.

Output: The length of a vector $||\hat{v}||$.

1.10 Minmaxed ⋆

Input: Two integers, a and b.

Output: Two integers, first the largest of them two, next the smallest.

1.11 TreE ⋆⋆⋆

Input: An integer h, the height of the christmass tree.

Output: A christmas tree with total height $h + 1$, 1 being the trunk of said tree and h all the result of it.

```
e
a a
e e e
a a a a
a
```

1.12 Sigma for Sum ⋆⋆

Input: An integer a such that $1 \leq a \leq 10^{10^{10}}$.

Output: The sum all the integers $1 + 2 + \ldots + a$.

Hint: Loop isn't the only way to go.

1.13 Factor!al ⋆⋆⋆

Input: An integer a such that $1 \leq b \leq 10^5$.

Output: The product all the integers $1 \times 2 \times \cdots \times b$.

Hint: Lookup the arguments for `range` in the official Python3.x documentation.

1.14 Minmaxed 2: The Sequel ⋆⋆⋆

Input: Two integers, a and b.

Output: Two integers, first the largest of them two, next the smallest.

Note: You may only use `min()` or `max()`, not both. You may not use branching.

2 Turtle or Tortoise?

Prerequisites: `turtle` module, the entire previous section.

2.1 Fair Square ⋆

Input: An integer A such that $10 \leq A \leq 100$.

Output: Using `from turtle import Turtle`'s methods like `forward` and `right` draw a square of length A.

2.2 Fair Ngon ⋆⋆

Input: Two integers, A such that $10 \leq A \leq 100$ and N such that $2 \leq N \leq 20$.

Output: Using `Turtle` draw a regular polygon (an N-gon) with N sides and side length A. Ensure that the turtle finishes in the same position as it started in. The turtle shouldn't draw over itself at any point.

Hint: Loops are your friend.

2.3 Trigonometry BFF ⋆⋆⋆

Input: Two integers, a and b.

Output: Using `Turtle` draw a graph of the function $y = a * sin(\frac{\pi x}{10}) + b$. From 0 to 20 and a graph of the function $y = b$. Print the final position of the turtle.

Hint: You can get sin and π with `from math import pi, sin`, they are accurate enough for this purpose.

2.4 The Fair Ngon ⋆⋆⋆⋆⋆

Input: Two integers, A such that $10 \leq A \leq 100$ and N such that $2 \leq N \leq 20$.

Output: Using `Turtle` draw a regular polygon (an N-gon) with N sides and side length A. Ensure that the turtle finishes in the same position as it started in. You are only allowed to control the turtle with `penup`, `pendown`, `goto`.

Hint: Trigonometry might help.