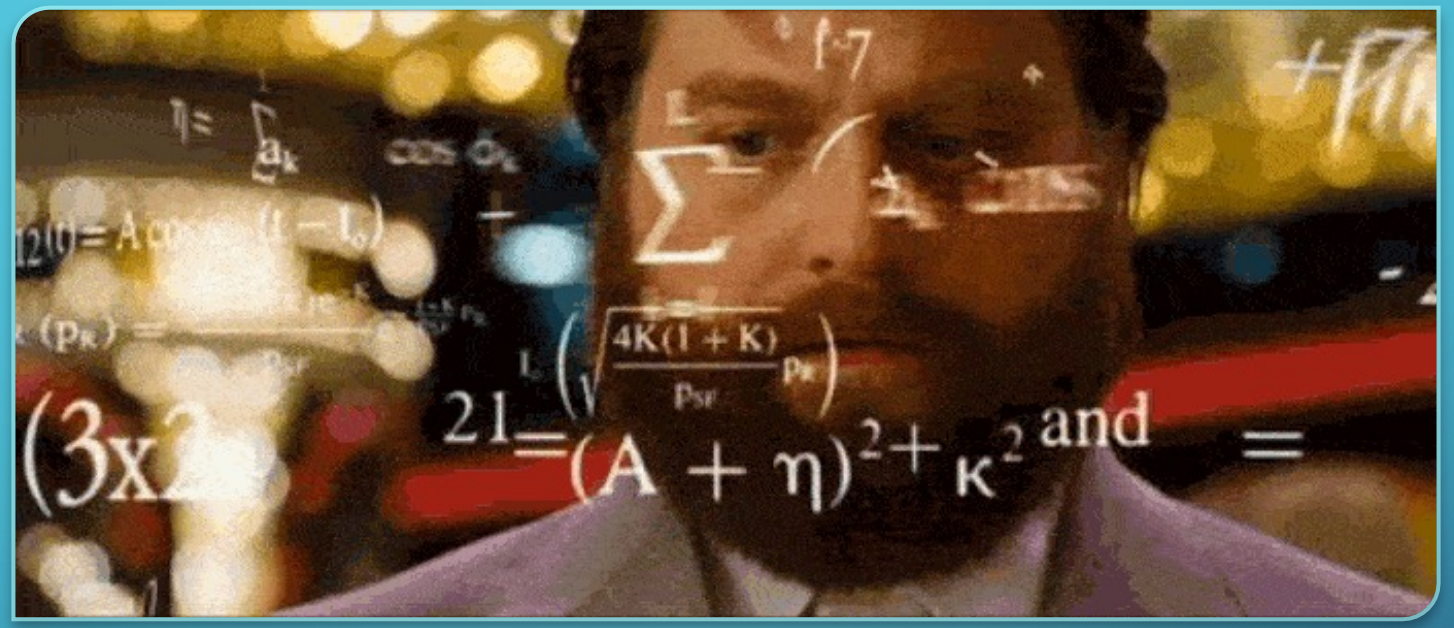

DECOMPOSING THOUGHTS & CLI (COMMAND LINE INPUT) PROGRAMS

THIS WEEK

- **Pseudocode** – Helps you think about a problem and alternative routes to approach it
- **Modulo**, the % operator – returns the remainder of the divided numbers. Can be used to do something every x iteration (every 3rd, 4th, 10th, 100th time etc.) by checking if there is no remainder between the iteration and the given value; if x % 10 == 0:
- **sys.argv** – A variable which contains arguments from the command line
- **sys.stdin** – Reads a file from the command line with <
- The **.rstrip()** method – Removes any trailing characters to the string, default is spaces
- The **enumerate()** function – Counts the current iteration of a loop, saving it in a variable

from random import randrange, randint

- **randrange(start, stop, step)** – returns a randomly selected element from the specified range
- **randint(start, stop)** – returns an integer element from the specified range

THE % OPERATOR - MODULO

Returns the remainder of dividing two numbers - Great for doing something every x times

```
>10 % 6
>> 4
>11 % 6
>> 5
>12 % 6
>> 0
```

```
for num in range(61):
    if num % 6 == 0:
        print(num, 'is part of the multiplication table for 6')
```

X % Y - If y is greater than x, the expression returns x

ENUMERATE()

```
for iteration_variable, element_variable in enumerate(range(x)):
    print('This is the', iteration_variable, 'iteration. The current value is', element_variable)


for i, num in enumerate(range(1, 61)):
    print('This is the', i, 'iteration. The current value is', num)
```

Good for tracking iteration and index - letter_count.py

SYS.ARGV

```
import sys


for argument in sys.argv:
    print(argument)
```

Stores arguments from the command line in a list – you will learn about lists next week

sys.argv = [argument1, argument 2, argument3…]

SYS.STDIN

```
import sys


for line in sys.stdin:
    name = line.rstrip()
    print("Hej " + name + ", welcome to class!")
```

Can be used to read a file from the command line with <

Calls the input() method internally