

Homework 1.

1.1. Write a script (or function) that takes as an input two numbers `width` and `height` and prints a rectangle with specified size.

Examples:

- `width=5` and `height=4` should result in terminal output:

```
#####
#   #
#   #
#   #
#####
```

- `width=7` and `height=2` should result in terminal output:

```
#####
#####
```

- `width=1` and `height=1` should result in terminal output:

```
#
```

1.2. Sum all perfect squares (numbers that are equal to the square of other number – 1, 4, ..., 9, 16, 25, 36, 49, ..., 9801, 10000) in range between 0 and 10000.

1.3. Write a script for setting an alarm, which ask users whether they are employed (yes / no) and whether they are currently on vacation (yes / no). User should answer typing either `Y` for yes or `N` for no. If user specify incorrect answer (anything that is not `Y` or `N`) program should warn user about incorrect answer and ask again.

The script output `True` if user is employed and not on vacation (because these are the circumstances under which you need to set an alarm). It should output `False` otherwise. Examples:

Examples:

- setting an alarm

```
> Are you employed? (Y/N):
> y
> Incorrect answer.
> Are you employed? (Y/N):
> Y
> Are you on vacation (Y/N):
> N
> True
```

- not setting an alarm

```
> Are you employed? (Y/N):
> N
> Are you on vacation (Y/N):
> N
> False
```

Homework 2.

2.1. Write a script approximating `cos(x)` as a first five terms of Taylor series expansion (see [trigonometric functions section in Wikipedia](#)). User should provide an input value of `x` and script should output calculated value for each term as well as current sum. Output should be formatted according to scheme:

```
> Calculating cos(1.0471975511965976) as Taylor expansion...
>
> Value for k=0 is +1.0000 (current sum is 1.0000)
> Value for k=1 is -0.5483 (current sum is 0.4517)
> Value for k=2 is +0.0501 (current sum is 0.5018)
> Value for k=3 is -0.0018 (current sum is 0.5000)
> Value for k=4 is +0.0000 (current sum is 0.5000)
```

Use either the `.format()` method or f-strings.

You might need `factorial` function from `math` module

2.2. Write two scripts `coefficients.py` and `quadratic.py`. In `coefficients.py` define three variables `a`, `b` and `c` representing coefficients of the quadratic equation $ax^2 + bx + c$. In `quadratic.py` import coefficients, solve the equation and output properly formatted the solution. For example in `coefficients.py` define:

```
a = 1
b = -10
c = 25
```

Then running `quadratic.py` should produce:

```
> One solution found x=5.000
```

Output for two solutions should look like this:

```
> Two solutions found x=-2.414 and x=0.414
```

Output for no solutions (when delta is less than zero):

```
> No solutions found
```

Imports will work only when these two files will be placed within the same directory

2.3. Write a script that takes an input string from the user and transforms it according to three rules:

1. delete all vowels,
2. leave all consonants and place `.` (dot) after them
3. transform all uppercase letters for lowercase letters

Examples:

- input `Programming` should be transformed into `p.r.g.r.m.m.n.g.`
- input `ABACC` should be transformed into `b.c.c.`
- input `aaa` should be transformed into an empty string

Consider only inputs consisting of lowercase and uppercase ascii letters (a-z) and (A-Z)