INF211

LABORATORY LEAFLET

FOR STUDENTS

LABORATORY-2
String Manipulation

Tasks	Explanations
1	Some mathematical conversions with inputs taken from the user
2	Calculation of some special numbers with inputs taken from the user
3	Reversing a string and removing unwanted printable characters
4	Valid brackets

Task 1a: Ask for a Fahrenheit degree from the user, then convert to Celsius and print the result as a float.

- inputs : input = { x: x ∈ R and 200.0 ≥ x ≥ -100.0}
- output : float (with epsilon = 1E-9)

The output must be as follows:

```
Enter Fahrenheit degree: 68

20.0
```

Task 1b: Ask for a Celsius degree from the user, then convert to Fahrenheit and print the result as a float.

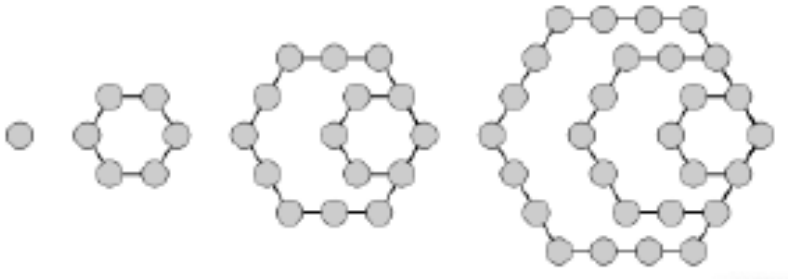
- inputs : input = { x: x ∈ R and 100.0 ≥ x ≥ -100.0}
- output : float (with epsilon = 1E-9)

The output must be as follows:

```
Enter Celsius degree: 20

68.0
```

Task 2a: The nth hexagonal number h_n is the number of distinct dots in a pattern of dots consisting of the outlines of regular hexagons with sides up to n dots, when the hexagons are overlaid so that they share one vertex. [1]



The formula for the nth hexagonal number is given as:

$$h_n = 2n^2 - n$$

Ask for a number from the user. Calculate and print the hexagonal number that corresponds to that number. As an example, first 7 hexagonal numbers are: 1, 6, 15, 28, 45, 66, 91.

- inputs : input = {x: x ∈ ℕ and 1E6 ≥ x ≥ 1}
- output : integer

The output must be as follows:

```
Enter a number: 1

1

>>>

Enter a number: 6

66
```

Task 2b: Each Lucas number is defined to be the sum of its two immediate previous terms and the first two Lucas numbers are L(0) = 2, and L(1) = 1 as shown below. [2]

$$L_n = \begin{cases} 2 & if\ n = 0 \\ 1 & if\ n = 1 \\ L_{n-1} + L_{n-2} & if\ n > 1 \end{cases}$$

Ask for a number from the user. Calculate and print the lucas number that corresponds to that number starting with index 0. As an example, first 7 Lucas numbers are: 2, 1, 3, 4, 7, 11, 18.

- inputs : input = {x: x ∈ ℕ and 1E6 ≥ x ≥ 0}
- output : integer

The output must be as follows:

```
Enter a number: 0
2
>>>
Enter a number: 6
18
```

Task 3a: Ask for an input string from the user and print the reverse of it.

- `inputs : input ∈ {x: printable characters except whitespaces and len(x) ∈ [1, 100]}.`
- `output : string`

The output must be as follows:

```
Enter a string: Inf211
112fnI
```

Task 3b: Ask for an input string from the user and print the string with only letters and numbers in the entered order.

- Unwanted printable characters that should be removed are: !"#$%&\'()*+,-./:;<=>?@[\\]^_`{|}~
- Note that the string can include " or ' characters, so take necessary precautions.
- inputs : input = {x: x ∈ printable characters except whitespaces and len(x) ∈ [1, 100]}.
- output : string

The output must be as follows:

```
Enter a string: :I!n#f21@1;,
Inf211
>>>
Enter a string: :I!n'"#f"2%$/1@1;,.
Inf211
```

Task 4: Ask for an input from the user containing only parentheses. Then print if the parentheses are actually in correct order (or it is valid).

- The parentheses are valid if open brackets closed by the same type of brackets and open brackets closed in the correct order.
- inputs: input = {x: x ∈ "{,},(,),[,]" and len(x) ∈ [1, 100]}

- output: Boolean

The output must be as follows:

```
>>>
Enter input: {}
True
>>>
Enter input: {}{}[]

True

>>>

Enter input: {([])}[]

True

>>>

Enter input: {(})

False

Explanation: After {( expression, first ( should be closed with ), then }.

>>>

Enter input: [()

False
```

[1] https://en.wikipedia.org/wiki/Hexagonal_number
[2] https://en.wikipedia.org/wiki/Lucas_number