

### Python Assignment -4

1. Write a function `translate()` that will translate a. That is, double every consonant and place an occurrence of "o" in between. For example, `translate("this is fun")` should return the string "tothohisos isos fofunon".

2. Write a program that contains a function that has one parameter, `n`, representing an integer greater than 0. The function should return `n!` (`n` factorial). Then write a main function that calls this function with the values 1 through 20, one at a time, printing the returned results. This is what your output should look like:

1 1

2 2

3 6 etc

3. Write a function `find_longest_word()` that takes a list of words and returns the length of the longest one.

4. Define a simple "spelling correction" function `correct()` that takes a string and sees to it that

a) two or more occurrences of the space character is compressed into one, and

b) inserts an extra space after a period if the period is directly followed by a letter. e.g. `correct("This is very funny and cool.Indeed!")` should return "This is very funny and cool. Indeed!"

5. In English, the present participle is formed by adding the suffix `-ing` to the infinite form: `go -> going`. A simple set of heuristic rules can be given as follows:

a) If the verb ends in `e`, drop the `e` and add `ing` (if not exception: `be`,

see, flee, knee, etc.)

b) If the verb ends in ie, change ie to y and add ing

c) For words consisting of consonant-vowel-consonant, double the final letter before adding ing

d) By default just add ing

Your task in this exercise is to define a function `make_ing_form()` which, given a verb in infinitive form, returns its present participle form. Test your function with words such as lie, see, move and hug. However, you must not expect such simple rules to work for all cases.

6. Make the program of network error codes done yesterday with match-case.

7. Define a class `Student` with data members `rollno`, `name`, `mark1`, `mark2`, `mark3`, `total`, `avg`. Use appropriate method for entering the details and displaying the details. Also define a method for calculating the total mark and average. Create an object for the class and invoke all the methods.

8. Create a class corresponding to `BankAccount` with the data members `accno`, `custname`, `balamt`. Use two methods for entering the details and displaying the details. Define one more method for checking whether the `balamt` is greater than 20,000 and display a corresponding message.

Create an object for the class and invoke all the methods.