MATH 4332

Assignment# 10

Due : 12/01/2023, Friday, before 11:59pm Term `:Fall 2023`

Important Note: Do not use regular expression in this assignment.

---

1. Write a function called `closest()` that takes a list of integers and another integer `n` and returns the integer in the list that is the closest to `n`. If two integers are tied for the closest, then return the smaller of the two.

2. Write a function called `names()` that returns a list of all of the names in the following string:

   ```
   s = """Anna is 7 years old, and her sister Olivia is 2 years old.
   Evelyn and Paul, their parents, have 3 kids."""
   ```

   Also find `len(names())`.

3. The file `grades.txt` contains a line separated list of people with their grade in a class. Write a function called `grades()` that returns a list of just those students who received a `B` in the course. Also find `len(grades())`.

4. Consider the file `logdata.txt`. This file records the access a user makes when visiting a web page. Each line of the `logdata.txt` has the following items:

   - a host (e.g., '146.204.224.152')
   - a `user_name` (e.g., `'feest6811'` note: sometimes the username is missing! In this case, use `'-'` as the value for the username.)
   - the time a request was made (e.g., `'21/Jun/2019:15:45:24 -0700'`)
   - the post request type (e.g., `'POST /incentivize HTTP/1.1'` note: not everything is a `POST!`).

   Write a function called `logs()` that reads through the lines of the file and converts them into a list of dictionaries, where each dictionary looks like the following:

   ```
   example_dict = {"host":"146.204.224.152",
                   "user_name":"feest6811",
                   "time":"21/Jun/2019:15:45:24 -0700",
                   "request":"POST /incentivize HTTP/1.1"}
   ```

   Also find `len(logs())`.

5. Recall that if `s` is a string, then `s.find('a')` will find the location of the first `a` in `s`. The problem is that it does not find the location of every `a`. Write a function called `findall` that given a string and a single character, returns a list containing all of the locations of that character in the string. It should return an empty list if there are no occurrences of the character in the string.

6. The `digital root` of a number `n` is obtained as follows: Add up the digits `n` to get a new number. Add up the digits of that to get another new number. Keep doing this until you get a number that has only one digit. That number is the `digital root`. For example, if `n = 45893`, we add up the digits to get `4 + 5 + 8 + 9 + 3 = 29`. We then add up the digits of `29` to get `2 + 9 = 11`. We then add up the digits of `11` to get `1 + 1 = 2`. Since `2` has only one digit, `2` is our `digital root`. Write a function that returns the `digital root` of an integer `n`. [Note: there is a shortcut, where the `digital root` is equal to `n mod 9`, but do not use that here.]

7. Write a function called `closest` that takes a list of numbers `L` and a number `n` and returns the largest element in `L` that is not larger than `n`. For instance, if `L = [1,6,3,9,11]` and `n = 8`, then the function should return `6`, because `6` is the closest thing in `L` to `8` that is not larger than `8`. Don't worry about if all of the things in `L` are smaller than `n`.