Do You Know? Introduction to Programming with Python

The following are programming projects similar in content and difficulty to those covered in our **Introduction to Programming with Python** class. If you can implement all of the following programs with little difficulty, then the class **Introduction to Programming with Python** would largely serve as a review for you.

Note: this class uses the Python programming language. If you can solve these problems using a different programming language, then you may be better served by learning Python on your own, as the course covers basic programming concepts that you would likely find review.

1. In a **Caesar shift code**, each letter in a message is shifted n places later in the alphabet (with Z wrapping around to A). For example, if $n = 1$, the word `example` becomes `fybnqmf`. If $n = 2$, the word `python` becomes `ravjqp`.

   Write a function called `caesar_shift()` that takes two parameters. The first is a word to encode. The second is the number of places to shift. It returns the shifted text. You can assume the first parameter is a single word with no spaces or other punctuation. You may also assume it is all lowercase.

   Some sample calls:

   ```
   >>> caesarShift('example', 1)
   'fybnqmf'
   >>> caesarShift('example', -1)
   'dwzlokd'
   >>> caesarShift('python', 2)
   'ravjqp'
   >>> caesarShift('pecan', 4)
   'tiger'
   ```

2. Suppose the file `studentdata.txt` contains information on grades students earned on various assignments. Each line has the last name of a student (which you can assume is one word) and the numeric grade that student received. All grades are out of 100 points. Students can appear multiple times in the file.

   Here's a sample file:

   ```
   Arnold 90
   Brown 84
   Arnold 80
   Cocher 77
   Cocher 100
   ```

   Write a function that reads the data from the file into a dictionary. Then continue prompting the user for names of students. For each student, it should print the average of that student's grades. Stop prompting when the user enters the name of a student not in the dictionary.

   A sample run for the given file:

   ```
   Enter name: Arnold
   The average for Arnold is: 85.0
   Enter name: Brown
   ```

© 2019 AoPS Incorporated

```
The average for Brown is: 84.0
Enter name: Cocher
The average for Cocher is: 88.5
Enter name: Doherty
Goodbye!
```

3. The *Fibonacci numbers* are the numbers

$$1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, \ldots$$

where every number (after the first two 1's) is the sum of the previous two numbers (for example, $34 = 21 + 13$). The numbers are counted starting with the first 1, so that, for example, the 7^{th} Fibonacci number is 13. Write a program that asks the user to enter a positive integer n and then outputs the n^{th} Fibonacci number. (For example, if the user enters 10, the program would output 55.)

Don't look at the next page until you've attempted all the problems!

© 2019 AoPS Incorporated

Sample solutions to the programming projects from *Do You Need Introduction to Programming with Python* are below. These are only samples and there are many different possible approaches. All of the techniques used in the programs below will be covered during the class. (Also note that these programs are written in Python 3. If you are using Python 2, there may be slight differences in your solutions.)

1.
```
def caesar_shift(word, offset):
    """Shifts each lowercase character in the given word the given
    number of places, and returns the result."""
    letters = "abcdefghijklmnopqrstuvwxyz" # the lowercase alphabet
    output = ""
    for char in word:
        if char in letters:
            # add the offset and move the result back into the 0..25 range
            newIndex = (letters.index(char) + offset) % 26
            output = output + letters[newIndex]
        else: # char isn't a lowercase letter -> don't convert
            output = output + char
    return output
```

2.
```
def student_grades():
    """Reads student names and grades from a file and stores in dictionary
    It then returns the average grade for the requested student.
    This ensures that a student can receive multiple grades if he/she
    is already present in the gradesDict."""
    file = open("studentdata.txt", "r")
    gradesDict = {}  # dictionary with names as keys and lists of grades as values
    for line in file:
        ls = line.split()
        if ls[0] in gradesDict:
            # if name already in the dictionary, add the grade to the grade list
            gradesDict[ls[0]] = gradesDict[ls[0]] + [int(ls[1])]
        else:
            # create a new dictionary entry, and put the first grade in the list
            gradesDict[ls[0]] = [int(ls[1])]
    name = input("Enter name: ")
    while name in gradesDict:
        # compute and display the average for the student
        avg = sum(gradesDict[name])/len(gradesDict[name])
        print("The average for "+str(name)+" is "+str(avg))
        name = input("Enter name: ")
    print ("Goodbye!")
```

© 2019 AoPS Incorporated

3.
```
def fibonacci():
    """Asks user to enter an integer n
    Computes the nth Fibonacci number"""
    num = int(input("Please enter a positive integer: "))
    fib = [1,1] # first two Fibonacci numbers
    if num <= 2:
        answer = 1
    else:
        for i in range(num-2):
            # compute the next Fibonacci number
            fib.append(fib[-1]+fib[-2])
        answer = fib[-1]
    print("The "+str(num)+"th Fibonacci number is: "+str(answer))
```


© 2019 AoPS Incorporated