CSC1015F Assignment 1

Introduction

Assignment Instructions

The first two questions in this assignment give you practice in typing in and editing programs. The next two are slightly more substantial. Question three involves a program that consists of the right statements but in the wrong order. Last but not lease, Question 4 gives us some level of independence whereby we get to write our own program as per the given specification. ```

Question 1 [20 marks]

Retype the following program into a file called `secret.py` and test that it runs. The program must be identical, down to the last bit of punctuation, including the spaces at the beginning of some lines!

Note: If you copy and paste code, you will most probably have problems submitting it to the Automarker. Furthermore, leave the author of the code as "Hussein Suleman" as that will be a way of referencing/acknowledging the author. Remember that if we do not acknowledge the author, that constitutes plagiarism.

You may need to create directories as per the instructions in the orientation manual.

```
# program to guess a secret number
# Huleman Suleman
# 10 february 2011

secret_number = 42    # create secret number in program
guess = 0             # variable to store user's guess

# as long as we have not found the secret number
while guess != secret_number:
    # get a new guess from user
    guess = eval(input("? "))
    # check if guess is too low
    if guess < secret_number:
        print ("lo")
    # or too high
    elif guess > secret_number:
        print ("hi")

print ("Correct!")    # print message indicating success
```

This program is a classic from the early days of Computer Science. A user is expected to guess numbers until he or she converges to a secret internal number. At each incorrect guess, the system lets the user know if the number is too high or too low.

Sample Input/Output (*The input from the user is shown in bold***):**

```
? 22
lo
? 55
hi
? 42
Correct!
```

Question 2 [20 marks]

Edit the program from Question one so that the messages printed are more user-friendly. You need to copy the secret.py file from the first question to a file called `secret_2.py`.

Change each of the messages printed to the screen to be the same as the example output below.

User-friendliness of programs was a concept that gained popularity in the 1980s, where programs were made easier for human beings to identify with. This has since grown into the current field of Usability Engineering, which you will learn about while studying Computer Science.

Sample Input/Output (*The input from the user is shown in bold font****):***

```
What is the secret number? 14
That is way too low. Please try again.
What is the secret number? 337
That is much too high. Please try again.
What is the secret number? 48
That is much too high. Please try again.
What is the secret number? 40
That is way too low. Please try again.
What is the secret number? 0
That is way too low. Please try again.
What is the secret number? 42
Congratulations, you have guessed the secret number!
```

Question 3 [30 marks]

You will find the following program on the Amathuba page for this assignment. It is called '`time.py`'.

NOTE: leave the author of the code as "Stephan Jamieson" as that will be a way of referencing/acknowledging the author. Remember that if we do not acknowledge the author, that constitutes plagiarism.

```
# Program to convert an amount of minutes into an equivalent amount
# of days, hours and minutes.
#
# Name: Stephan Jamieson
#
minutes = int(input_str)
days = hours//24
```

```
print(".")

hours = minutes//60

print("The number of days is", days, end=', ')
print("and the number of minutes is", minutes, end='')

minutes = minutes%60

print("the number hours is", hours, end=', ')

input_str = input("Enter a quantity of minutes: ")
hours = hours%24
```

The program consists of correct statements that are in the wrong order. Here is an example of how the program is supposed to behave:

```
Enter a quantity of minutes: 3500
The number of days is 2, the number hours is 10, and the number of
minutes is 20.
```

Download the program and rearrange the statements so that it operates correctly.

HINT: check you understand what the integer operations '//' and '%' do, and think how, given an amount in minutes, you would calculate the equivalent amount of days, hours and minutes.

Question 4 [30 marks]

Albert Einstein[1] once said "*it followed from the special theory of relativity that mass and energy are both but different manifestations of the same thing — a somewhat unfamiliar conception for the average mind.*"

You do not need to know the Physics behind the equation below. However, if you are curious, you may read further here: https://www.britannica.com/science/E-mc2-equation. Our interest is to write a python program to calculate the energy, *E*, given the values of the mass, *m*, and the speed of light, *c* based on the following equation:

$$E = mc^2$$

[1] See https://www.forbes.com/sites/startswithabang/2018/01/23/the-three-meanings-of-emc2-einsteins-most-famous-equation/?sh=e19fd8a71c0b Last accessed on 16th February 2023.

Task:

Write a program called '`energy.py`' to input the values of the integer numbers *m* and *c*, then calculate and output the value of the energy quantity from the equation above.

*(**Note**: recall that, in Python, the statement d ** 3 is equivalent to d^3)*

***Sample I/O** (The input from the user is shown in bold font)*

```
Enter the value of m:
6
Enter the value of c:
6
The value of energy, E, is: 216
```

***Sample I/O** (The input from the user is shown in bold font)*

```
Enter the value of m:
4
Enter the value of c:
6
The value of energy, E, is: 144
```

Submission

Create and submit to the automatic marker a Zip file called `ABCXYZ123.zip` (where ABCXYZ123 is YOUR student number) containing `secret.py`, `secret_2.py`, `perimeter.py`, and `energy.py`.