IF100 – Spring 2019-2020
Take-Home Exam 1
Due March 11th, 2020, Wednesday, 23:55 (Sharp Deadline)

Introduction

The aim of this take-home exam is to practice on the basics of programming. You will write a Python program to get some inputs from the user, do some arithmetic operations and display the result as the output to the user.

Description

Let's assume that you are a spy of an agency called SU Intelligence Agency (SUIA), and you are planning to gather secret information from the computer of a very dangerous crime boss. The computer is in a room which is being protected by a guard and an electronic door lock. Also, the computer has a password which you need to crack before you can start gathering information. However, do not worry, you will have the specialized tools to crack both passwords. Your mission consists of 3 stages:

(i) *Cracking the password of the door lock*: We know that the password of the door lock includes digits only (from `0` to `9`), and the tool you have can try `1000` different passwords in a second.

(ii) *Cracking the password of the computer*: We have the information that the password of the computer has been created by using the characters in the string "`sabanciuniversity2020`". However, any character could be used several times. In other words, only `14` unique characters in the given string could have been used to create the password of the computer. Luckily, the tool you possess can try `600000` passwords in a second.

(iii) *Transfering the data*: We know that the speed of data transfer is constant and it is equal to `60 MBps (Megabytes per second)`.

Your aim here is to calculate how long it will take to complete your mission in the worst case, i.e. how long would it take to crack both of the passwords at most, and inform your partner about how long he has to distract the door guard. To calculate the duration of both of the password cracking stages, the program that you will code will need the length of the passwords as input. On the other hand, in order to calculate the duration of the data transfer stage, your program will need the size of the data to be transferred as another input.

Inputs

The program that you will develop needs to take a total of 3 inputs from the user:

1. Length of the door lock password,
2. Length of the computer password,
3. Size of the data to be transferred in GB.

You may assume that the user will always enter numeric values for all of these inputs. To be more precise, the first two inputs (door lock password and computer password) will be of type integer, and the third input (data size) can be a real number.

Here are some hints:

- `1 GB` is `1024 MB`.
- You need to calculate the number of all possible passwords while calculating the duration of both of password cracking stages.
- The number of all possible passwords is n^m, where `n` is the repertory size (i.e. total number of unique characters that may appear in the password), and `m` is the length of the password. For example, if a password is of length `3`, and if this password consists only of digits (i.e. repertory size of `10` since there are a total of `10` different digits - from `0` to `9`), then the number of all possible passwords will be 10^3 ($10x10x10=10^3$).

Output

You need to calculate and display the total time of the mission with the given inputs. The output of your program should be exactly in the following format:

You need to distract the guard's attention for h hour(s), m minutes and s second(s).

You should calculate three numbers (*h*, *m* and *s*) for the output of your program. If you find one of these results as `0`, you should also print that.

Please note that *h* and *m* values must be displayed as integers (without any floating point numbers); however *s* value must be displayed as a real value with *exactly* two floating point numbers.

You may check the "Sample Runs" section given below for some examples.

Sample Runs

Below, we provide some sample runs of the program that you will develop. The *italic* and **bold** phrases are inputs taken from the user. You have to display the required information in the same order and with the same words and characters as below.

Sample Run 1

```
Please enter the number of digits in padd-lock: 4
Please enter the password length of the PC: 6
Please enter the file size you want to transfer (GB): 60
You need to distract the guard's attention for 0 hour(s), 17 minute(s) and 26.55 second(s).
```

Sample Run 2

```
Please enter the number of digits in padd-lock: 5
Please enter the password length of the PC: 8
Please enter the file size you want to transfer (GB): 3.5
You need to distract the guard's attention for 0 hour(s), 43 minute(s) and 39.38 second(s).
```

Sample Run 3

```
Please enter the number of digits in padd-lock: 4
Please enter the password length of the PC: 5
Please enter the file size you want to transfer (GB): 120
You need to distract the guard's attention for 0 hour(s), 34 minute(s) and 18.90 second(s).
```

Sample Run 4

```
Please enter the number of digits in padd-lock: 6
Please enter the password length of the PC: 8
Please enter the file size you want to transfer (GB): 10.23
You need to distract the guard's attention for 1 hour(s), 0 minute(s) and 34.24 second(s).
```

How to get help?

You can use GradeChecker (http://sky.sabanciuniv.edu:8080/GradeChecker/) to check your expected grade. Just a reminder, you will see a character ¶ which refers to a newline in your expected output.

What and where to submit?

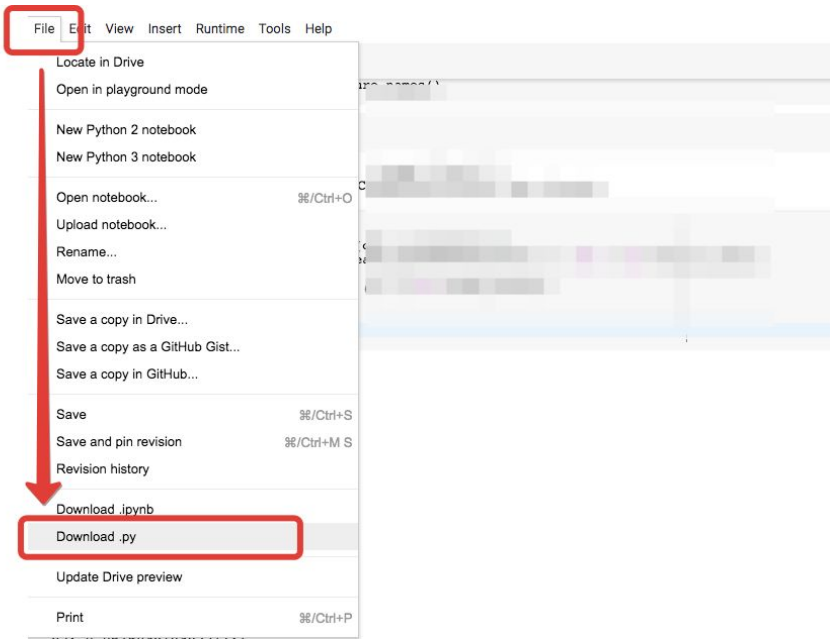
You should prepare (or at least test) your program using Python 3.x.x. We will use Python 3.x.x while testing your take-home exam.

It'd be a good idea to write your name and lastname in the program (as a comment line of course). Do not use any Turkish characters anywhere in your code (not even in comment parts). If your name and last name is "İnanç Arın", and if you want to write it as comment; then you must type it as follows:

 # Inanc Arin

Submission guidelines are below. Since the grading process will be automatic, students are expected to strictly follow these guidelines. If you do not follow these guidelines, your grade will be 0.

- Download your code as *py* file with "File" -> "*Download .py*" as below:



- Name your *py* file that contains your program as follows:

  "username_the1.py"

  For example: if your SUCourse username is **"duygukaltop"**, then the name of the *py* file should be: **duygukaltop_the1*.py*** (please only use lowercase letters).

- Please make sure that this file is the latest version of your take-home exam program.

- Submit your work **through SUCourse only**! You can use the GradeChecker only to see if your program can produce the correct outputs both in the correct order and in the correct format. It will not be considered as the official submission. You must submit your work to SUCourse.

General Take-Home Exam Rules

- Successful submission is one of the requirements of the take-home exam. If, for some reason, you cannot successfully submit your take-home exam and we cannot grade it, your grade will be 0.
- There is NO late submission. You need to submit your take-home exam before the deadline. Please be careful that SUCourse time and your computer time may have 1-2 minutes differences. You need to take this time difference into consideration.
- Do NOT submit your take-home exam via email or in hardcopy! SUCourse is the only way that you can submit your take-home exam.
- If your code does not work because of a syntax error, then we cannot grade it; and thus, your grade will be 0.
- Please do submit your **own** work only. It is really easy to find "similar" programs!
- Plagiarism will not be tolerated. Please check our plagiarism policy given in the syllabus of the course.

Good luck!

Elif Pınar Ön
Ethem Tunal Hamzaoğlu
IF100 Instructors