ASSIGNMENT 2

Read the instructions below carefully. The instructions must be followed. This assignment is worth 4% of your grade. The assignment is due on Monday 5th of October 8AM. No late assignment will be accepted.
This is an individual assignment. Please review the Plagiarism and Academic Integrity policy presented in the first class.
The goal of this assignment is to learn and practice the concepts covered thus far: function design, function calls, branching (i.e. if statements), strings. You can make multiple submissions, but only the last submission before the deadline will be graded. What needs to be submitted is explained next.
The assignment has two parts. Each part explains what needs to be submitted. Put all those required documents into a folder called a2_xxxxxx where you changed xxxxxx to your student number, zip that folder (do not use rar compression format) and submit it by the deadline via Brightspace. Ihe folder should contain the following 5 files:
– for Part 1: a2_part1_xxxxxx.py, a2_part1_xxxxxx.txt
– for Part 2: a2_part2_xxxxxx.py a2_part2_xxxxxx.txt and
– declaration-YOUR-FULL-NAME.txt

I suggest that you complete part 2 first.

About declaration-YOUR-FULL-NAME.txt file:
It needs to be a plain text file that must contain references to any code you used that you did not write yourself, including any code you got from a friend/person or any type of public media or other source. The only exclusion from that rule is the code that we did in class or as part of the lab work. So here is what needs to be written in that file. In every question where you used code from somebody else, you must write:
1. question number
2. copy-pasted parts of the code that were written by somebody else. That includes the code you found/were-given that you then slightly modified.
3. whose code it is: name of a person or place on internet/book where you found it.
While you may not get points for that part of the question, you will not be in position of being accused of plagiarism.
Not including declaration-YOUR-FULL-NAME.txt will be taken as you are declaring that all the code in the assignment was written by you. Any student caught in plagiarism will receive zero for the whole assignment and will be reported to the dean. Finally showing/giving any part of your assignment code to a friend also constitute plagiarism and the same penalties will apply.

Both of your programs/parts must run without syntax errors. Particularly, when grading your assignment, TAs will first open your file a2_part1_xxxxxx.py with IDLE and press Run Module. If pressing Run Module causes any syntax error, the grade for Part 1 becomes zero. The same applies to Part 2, when they open and run file a2_part2_xxxxxx.py.
Furthermore, for each of the functions (in Part 1 and Part 2), one or more tests have been provided to validate your functions.
For example, you should test function high_school_quiz from Part 1 by making a call in Python shell with high_school_quiz(3,2,15).
To obtain a partial mark your function may not necessarily give the correct answer on these tests. But if your function gives any kind of python error when run on the tests provided below, that question will be marked with zero points.
Section 3 contains tests for Part 1. Tests for Part 2 are provided after each question in Part 2.

To determine your grade, your functions will be tested both with examples provided in Part 2 and section 4 and with some other examples. Thus, you too should test your functions with more examples than those provided in Part 2 and Section 3.
Each function must be documented with **docstrings**.
Global variables are not allowed. If you do not know what that means, for now, interpret this to mean that inside of your functions you can only use variables that are created in that function. For example, the following is not allowed, since variable x is not a parameter of function a_times(a) nor is it a variable created in function a_times(a). It is a global variable created outside of all functions.

```
def a_times(a):
        result=x*a
        return result

x = float( input("Give me a number: ") )
print(a_times(10))
```

1 Part 1: Math quiz-generator / equation-solver - 40 points

You would like to help your community center and teach elementary and high school students' math. To reach as many kids as possible, you decided to write some software that can automate some tasks for you. As a starting point, you decided to make an automatic quiz generator and grader for elementary school pupils and a quadratic equation solver for high school pupils that should handle all possible cases including those when the given quadratic equation has complex numbers for solutions.
For the elementary school pupils, your understanding is that they already know well additions and multiplications but that they struggle with subtractions and exponentiations.

For this part, a file called a1_part1_xxxxxx.py is provided with the starter code in. Start by replacing xxxxxx in file name with your student number. Then open the file. Your solution (code) for this part must go into that file in clearly indicated spaces. You are not allowed to delete or comment-out any parts of the provided code except for the keywords pass.

For this part, you need to submit two files: a2_part1_xxxxxx.py and a2_part1_xxxxxx.txt. a2_part1_xxxxxx.py needs to contain your program for Part 1 as explained above and a2_part1_xxxxxx.txt needs to contain the proof that you tested your two core functions from this part, namely elementary_school_quiz and high_school_quiz.

1.1 The Core Functions

Your solution in a2_part1_xxxxxx.py must have two functions called: elementary_school_quiz and high_school_quiz.
You should first design and test these functions before moving onto the main part of the program. Here are specifications for the two functions:

elementary_school_quiz: This function takes two parameters, namely an integer *flag*, whose values can only be 0 or 1, and integer *n* that only takes values 1, or 2. If *flag* is 0, the function helps practice subtractions. But if it is 1, it helps practice exponentiations. The function then generates n math problems that a pupil must answer in turn. For each question, it generates two random positive, single-digit numbers (check out python's random module to see if there's a useful function in there) and asks the pupil for the answer to the math problem with those two numbers (either subtract the second number from the first, or raise the first number to the power of the second number). elementary_school_quiz then

prompts the pupil for the answer and checks if it is correct. At the end of n questions, elementary_school_quiz returns the number of questions answered correctly.
high_school_quiz: This function has three parameters representing three real numbers for the coefficients of the quadratic equation ax2+bx+c = 0. The function displays/prints the equation first and then prints its solutions. The function must display correct and meaningful solutions given any three real numbers for coefficients a, b and c. See examples in Section 3 to understand what that means. Please consider the examples to be a part of the function/program specifications that must be followed.
Note that to solve this problem, you do not need to use Python's complex numbers.

1.2 The User Interaction i.e. the main part of the program

Now that you have the two functions that perform the core functionality, you want to make it more user friendly for the pupils (after all, alas, the pupil may not know how to write code and call functions in Python shell). In the main part of your program, write your code in specified places. You code must follow the behavior indicated in the example runs in Section 3.

For example, for the elementary school pupil, called Mia, you will first ask her whether she would like to practice subtraction or exponentiation. Then ask her how many practice questions she'd like (if she says 0, then your code should not ask her to solve any math questions). Using her responses, call the elementary_school_quiz function with the appropriate values. When it returns the number of correct answers, display a message to the pupil:

- If all her answers are correct, display on screen: Congratulations Mia! You'll probably get an A tomorrow. Goodbye Mia!
- If half of her answers are correct, display on screen: You did ok Mia, but I know you can do better. Goodbye Mia!
- If all her answers are incorrect, display on screen: I think you need some more practice Mia. Goodbye Mia!

For high school pupils, in the main part of your program in the specified places, you need to write some code that asks the pupil for the coefficients a, b and c. Then in order to display the solutions, you need to make the call to high_school_quiz.
After that your program should ask the pupils if they would like another quadratic equation solved. If the pupil says anything but yes. the program terminates by printing a good bye message as in the examples. Otherwise (and any form of typing yes should be acceptable, including with lots of white space before and after, and with capital letters and lower case letter etc.), as long as pupil answers yes, he/she should be asked for the coefficients again and the resulting new quadratic equation should be solved. Since you have not seen *while* loops yet, a code was provided for that in a2_part1_xxxxxx.py.
The rest of the specifications for your program in Part 1 can be inferred from examples in section 3.

2 Part 2: A Library of Functions

For this part of the assignment, you are required to write and test 3 functions. You need to save all functions in a2_part2_xxxxxx.py where you replace xxxxxx by your student number. You need to test your functions (like you did in Assignment 1) and copy/paste your tests in a2_part2_xxxxxx.txt. Thus, for this part you need to submit two files: a2_part2_xxxxxx.py and a2_part2_xxxxxx.txt.

2.1 min_enclosing_rectangle(radius, x, y) - 5 points

Computing a smallest (axis-aligned) rectangle that encloses a set of objects in the plane is a very common computational problem arising in graphics and thus game development too. Write a function, called min_enclosing_rectangle, that has 3 input parameters. The first is a number representing a radius of a circle and the next two are two numbers representing the x- and y-coordinates of its center. Consider the smallest axis-aligned rectangle that contains that circle.
The function should return the x- and y-coordinates of the bottom-left corner of that rectangle. If side is a negative number, min_enclosing_rectangle should return None.

```
>>> min_enclosing_rectangle(1,1,1)
(0, 0)
>>> min_enclosing_rectangle(4.5, 10, 2)
(5.5, -2.5)
>>> min_enclosing_rectangle(-1, 10, 2)
>>> min_enclosing_rectangle(500, 1000, 2000)
(500, 1500)
```

2.2 vote_percentage (results) - 5 points

Write a function called vote_percentage that takes a string as input. The function has one input parameter, called results. Your function should count the number of substrings 'yes' in the string results and the number of substrings 'no' in the string results, and it should return the percentage of 'yes' (among all 'yes' and 'no'). (You may assume that string results has at least one yes or no and that the only words present are yes, no and/or abstained).
Hint: you may use count method from Python's *str* module/library.

```
>>> vote_percentage('yes yes yes yes yes abstained yes yes yes yes')
1.0
>>> vote_percentage('yes,yes, no,  yes, no, yes, abstained, yes,  yes, no')
0.6666666666666666
>>> vote_percentage('abstained no abstained yes no yes no yes yes yes no')
0.5555555555555556
>>> vote_percentage('no yes no no no yes yes yes no')
0.4444444444444444
```

2.3 vote(results) - 5 points

If there is a vote at a meeting, there are several possible outcomes based on the number of yes and no votes (abstains are not counted). If all the votes are yes, then the proposal passes "unanimously", if at least 2/3 of the votes are yes, then the proposal passes with "super majority", if at least 1/2 of the votes are yes, then the proposal passes by "simple majority", and otherwise it fails. Write a function called vote that asks a user to enter all yes-s and no-s and abstained-s and then press enter. The function then prints the outcome of the vote. Your solution must involve making a call to function vote_percentage.

```
>>> vote()
Enter the yes, no, abstained votes one by one and then press enter:
yes yes yes yes yes abstained yes yes yes yes
proposal passes unanimously
>>> vote()
Enter the yes, no, abstained votes one by one and then press enter:
yes,yes, no,  yes, no, yes, abstained, yes,  yes. no
proposal passes with super majority
>>> vote()
```

```
Enter the yes, no, abstained votes one by one and then press enter:
abstained no abstained yes no yes no yes yes yes no
proposal passes with simple majority
>>> vote()
Enter the yes no abstain votes one by one and then press enter:
no yes no no no, yes yes yes no
proposal fails
```

3 Testing your code in Part 1

Here is how you should test your two functions from Part 1 in Python shell.

```
>>> elementary_school_quiz(0,2)
Question 1:
What is the result of 1-1? 0
Question 1:
What is the result of 3-8? 1
1
>>> elementary_school_quiz(1,1)
Question 1:
What is the result of 1^8? 1
1
>>> elementary_school_quiz(1,1)
Question 1:
What is the result of 7^4? 20
0
>>> elementary_school_quiz(1,2)
Question 1:
What is the result of 9^7? 4782969
Question 1:
What is the result of 5^8? 390625
2
>>>
>>> high_school_quiz(1,3,1)
The quadratic equation 1·x^2 + 3·x + 1 = 0
has the following real roots:
-0.3819660112501051 and -2.618033988749895
>>>
>>>
>>> high_school_quiz(3,2,15)
4
The quadratic equation 3·x^2 + 2·x + 15 = 0
has the following two complex roots:
-0.3333333333333333 + i 2.2110831935702664
and
-0.3333333333333333 - i 2.2110831935702664
>>>
>>>
>>> high_school_quiz(3,2,-15)
The quadratic equation 3·x^2 + 2·x + -15 = 0
has the following real roots:
1.9274433277084226 and -2.5941099943750894
>>>
```

```
>>>
>>> high_school_quiz(1,2,1)
The quadratic equation 1·x^2 + 2·x + 1 = 0
has only one solution, a real root:
-1.0
>>>
>>>
>>> high_school_quiz(0,2,4)
The linear equation 2·x + 4 = 0
has the following root/solution: -2.0
>>>
>>>
>>> high_school_quiz(0,0,0)
The quadratic equation 0·x + 0 = 0
is satisfied for all numbers x
>>>
>>>
>>> high_school_quiz(0,0,10)
The quadratic equation 0·x + 10 = 0
is satisfied for no number x
```

Here is what pressing Run on your program (Part 1) should look like:

```
****************************************
* *
* __Welcome to my math quiz-generator__ *
* *
****************************************
What is your name? Mia
Hi Mia. Are you in? Enter
1 for elementary school
2 for high school or
3 or other character(s) for none of the above?
1
********************************************************************************
* *
* __Mia, welcome to my quiz-generator for elementary school students.__ *
* *
********************************************************************************
Mia what would you like to practice? Enter
0 for subtraction
1 for exponentiation
0
How many practice questions would you like to do? Enter 0, 1, or 2: 2
Mia, here is your 2 questions:
Question 1:
What is the result of 8-8? 0
Question 1:
What is the result of 9-3? 6
Congratulations Mia! You'll probably get an A tomorrow.
Good bye Mia!
```

Another example run:

```
****************************************
* *
* __Welcome to my math quiz-generator__ *
* *
****************************************
5
```

What is your name? Mia Duart
Hi Mia Duart. Are you in? Enter
1 for elementary school
2 for high school or
3 or other character(s) for none of the above?
Ah
Mia Duart you are not a target audience for this software.
Good bye Mia Duart!
Another example run:
>>>
****
* *
* __Welcome to my math quiz-generator__ *
* *
****
What is your name? Arya Stark
Hi Arya Stark. Are you in? Enter
1 for elementary school
2 for high school or
3 or other character(s) for none of the above?
2
****
* *
* __quadratic equation, a·x^2 + b·x + c= 0, solver for Arya Stark__ *
* *
****
Arya Stark, would you like a quadratic equation solved? YeS
Good choice!
Enter a number the coefficient a: 1
Enter a number the coefficient b: 2
Enter a number the coefficient c: 3
The quadratic equation 1.0·x^2 + 2.0·x + 3.0 = 0
has the following two complex roots:
-1.0 + i 1.4142135623730951
and
-1.0 - i 1.4142135623730951
Arya Stark, would you like a quadratic equation solved? yeS
Good choice!
Enter a number the coefficient a: -1
Enter a number the coefficient b: 10
Enter a number the coefficient c: 36
The quadratic equation -1.0·x^2 + 10.0·x + 36.0 = 0
has the following real roots:
-2.810249675906654 and 12.810249675906654
Arya Stark, would you like a quadratic equation solved? YES
Good choice!
Enter a number the coefficient a: 0
Enter a number the coefficient b: 2.5
Enter a number the coefficient c: 3.246
The linear equation 2.5·x + 3.246 = 0
has the following root/solution: -1.2984
Arya Stark, would you like a quadratic equation solved? yes
Good choice!
Enter a number the coefficient a: 0
Enter a number the coefficient b: 0
Enter a number the coefficient c: 2.5

The quadratic equation 0.0·x + 2.5 = 0
is satisfied for no number x
Arya Stark, would you like a quadratic equation solved? yes
Good choice!
Enter a number the coefficient a: 0
Enter a number the coefficient b: 0
Enter a number the coefficient c: 0
The quadratic equation 0.0·x + 0.0 = 0
is satisfied for all numbers x
Arya Stark, would you like a quadratic equation solved? NO
Good bye Arya Stark!
6
Another example run:
****
* *
* __Welcome to my math quiz-generator__ *
* *
****
What is your name? Ji-Ah
Hi Ji-Ah. Are you in? Enter
1 for elementary school
2 for high school or
3 or other character(s) for none of the above?
1
****
* *
* __Ji-Ah, welcome to my quiz-generator for elementary school students.__ *
* *
****
Ji-Ah what would you like to practice? Enter
0 for subtraction
1 for exponentiation
5
Invalid chose. Only 0 or 1 is accepted.
Good bye Ji-Ah!
Another example run:
****
* *
* __Welcome to my math quiz-generator__ *
* *
****
What is your name? Hippolyta
Hi Hippolyta. Are you in? Enter
1 for elementary school
2 for high school or
3 or other character(s) for none of the above?
1
****
* *
* __Hippolyta, welcome to my quiz-generator for elementary school students.__ *
* *
****
Hippolyta what would you like to practice? Enter
0 for subtraction
1 for exponentiation
0

How many practice questions would you like to do? Enter 0, 1, or 2: 0
Zero questions. OK. Good bye
Good bye Hippolyta!
}
Another example run:
```
****************************************
* *
* __Welcome to my math quiz-generator__ *
* *
****************************************
```
What is your name? Leti
Hi Leti. Are you in? Enter
1 for elementary school
2 for high school or
3 or other character(s) for none of the above?
1
7
```
*********************************************************************************
* *
* __Leti, welcome to my quiz-generator for elementary school students.__ *
* *
*********************************************************************************
```
Leti what would you like to practice? Enter
0 for subtraction
1 for exponentiation
1
How many practice questions would you like to do? Enter 0, 1, or 2: 5
Only 0,1, or 2 are valid choices for the number of questions.
Good bye Leti!