CSC015 Assignment 2 (100pts)

What to Submit:
Each problem as a separate file. Each file is a complete program. Do not submit test runs, just the code.

Each problem is 25 points.

Note that I will deduct from the max score for errors or bad style.

The purpose of this assignment is to enforce the concepts of variables, expressions, assignment, types (int, float, double) and type conversion, input and output, including formatted print(); the steps in solving a problem, organizing the program (thinking algorithmically), practicing good programing style. Remember, programs should be easy to understand, maintain and modify if needed.

We will start the work on the assignment during lab and you will continue and complete it on your own.

For each problem , to arrive to develop the solution follow the steps:

1. Make sure you understand the problem
2. Identify the input and output
3. Develop an algorithm, i.e. outline the steps that will lead from the input to output
4. Code: observe Python syntax and rules, observe rules for good style
5. Test your code and debug as needed

You do not have to submit to me the above process in each case. Submit only the program files (one for each problem). Remember to think in terms of INPUT, PROCESS and OUTPUT. Your programs must include a function definition of main(), and a function call to main().

Naming conventions: the file names should be hw2_ProblemNumber_YourName.py, for example hw2_1_GerdaKamberova.py.

Problem 1. The net monthly profit made by a Bitcoin miner is the total value of the blocks mined for the month minus electricity usage and hardware depreciation. The value of a mined block is $52,900. Electricity costs $ 0.12 per KWH. Hardware depreciates at a constant value of $100 flat fee (independent of number hours). Write a program to input the number of blocks mined for the month and the number of units of electricity used, and output the net monthly profit for the miner.

Below is a sample output

```
Program calculates profit of a bitcoin miner.
How many blocks? 1
How many hours? 1

Bitcoin miner monthly profit $52799.88
```

And here is the skeleton of the program I expect to see, fill in the missing parts (including the needed Python statements and comment at the top).

```
# ****************************************************************
# put the program comment here
# ****************************************************************


def main():
    BLOCK_PRICE = 52900
    COST_UNIT_ECECTRIC = 0.12
    HWR_DEPRICIATION = 100

    print("Program calculates profit of a bitcoin miner.")

    #input data


    #process, compute profit


    #output
    print('\nBitcoin miner monthly profit ${:.2f}'.format(profit))

    return

main()
```

When you test the program, enter values for which you can easily check the result as well as "extreme cases". Below is some tests that I ran when I did my code. Notice the test I did with 0 hours or 0 blocks for example, these are easy to check. Look carefully at the result and convince yourself the outputs are correct.

```
========== RESTART: D:\CSC15Working\hw\hw1
Program calculates profit of a bitcoin miner.
How many blocks? 0
How many hours? 1

Bitcoin miner monthly profit $-100.12
>>>
========== RESTART: D:\CSC15Working\hw\hw1
Program calculates profit of a bitcoin miner.
How many blocks? 0
How many hours? 1

Bitcoin miner monthly profit $-100.12
>>>
========== RESTART: D:\CSC15Working\hw\hw1
Program calculates profit of a bitcoin miner.
How many blocks? 1
How many hours? 0

Bitcoin miner monthly profit $52800.00
>>>
========== RESTART: D:\CSC15Working\hw\hw1
Program calculates profit of a bitcoin miner.
How many blocks? 1
How many hours? 1

Bitcoin miner monthly profit $52799.88
```

Problem 2: The harmonic mean of two real numbers X and Y is given by

$$H = \frac{2}{\frac{1}{X}+\frac{1}{Y}}.$$

The geometric mean of the two numbers is

$$G = \sqrt{XY}$$

The arithmetic mean is something you're already familiar with:

$$M = \frac{X+Y}{2}$$

Write a program to input two numbers from the keyboard and print their harmonic, geometric and arithmetic means.

Note that since for the harmonic mean you must compute as intermediate steps $1/X$ and $1/Y$, the two numbers the user enters must not be equal to 0 (since division by 0 is not allowed). Because of the geometric mean where you must compute $\sqrt{XY}$ since you cannot take a square root of a negative number, both numbers must have the same sign (both are positive or both are negative).

Below is a sample output of 3 sample runs. Note that I tested with different types of inputs (both positive and both negative inputs; using values for which I can easily check the result, as well as and more complex inputs).

```
Calculate arithmetic, geometric and harmonic means of two numbers.
Enter two numbers which are not 0 nad have the same sign.
number 1: 1.0
number 2: 1.0
        Arithmetic mean is 1.0
        Geometric mean is 1.0
        Harmonic mean is 1.0
>>>
========== RESTART: D:\CSC15Working\hw\hw1means.py ========
Calculate arithmetic, geometric and harmonic means of two numbers.
Enter two numbers which are not 0 nad have the same sign.
number 1: -2.0
number 2: -2.0
        Arithmetic mean is -2.0
        Geometric mean is 2.0
        Harmonic mean is -2.0
>>>
========== RESTART: D:\CSC15Working\hw\hw1means.py ========
Calculate arithmetic, geometric and harmonic means of two numbers.
Enter two numbers which are not 0 nad have the same sign.
number 1: 125.0
number 2: 0.01
        Arithmetic mean is 62.505
        Geometric mean is 1.118033988749895
        Harmonic mean is 0.01999840012798976
>>> |
```

To compute square root you need the *math module.* Put the statement

```
import math
```

at the top of the program, inside the definition of main(). When you need to compute square root of an expression, say `expr`, make the call to the method `.sqrt( )` from the math module:

```
math.sqrt(expr)
```

This function computes the square root of the expression `expr` and return the value of it. Do not forget to assign this result to a variable so you do not loose it. Structure correctly your arithmetic expressions, + has lower precedence than division / .

Problem 3. My office phone number **is (516) 463-5775.**

- 516 is the area code
- 463 is the prefix
- 5775 is the line number

All phone numbers in US are structured like this. My phone number written as a 10-digit integer is 5164635775.

Write a program that asks the user to enter a name and a 10-digit integer number from the keyboard and prints the name and the phone number in the usual format with area code in ( ) and a - between the prefix and the line number.

Below is a sample output from two runs:

```
========== RESTART: D:/CSC15Workin
Program to print personal data.
Enter name: Gerda Kamberova
Enter 10-digit integer number: 5164635775
Gerda Kamberova  (516)463-5775
>>>
========== RESTART: D:/CSC15Workin
Program to print personal data.
Enter name: Humpty Dumpty
Enter 10-digit integer number: 2121005001
Humpty Dumpty  (212)100-5001
```

Since we have not studied strings in depth yet, to extract the various part of the phone number (area code, prefix and line) use the integer division // and mod % operators as appropriate. For Assume that only phone numbers for which the area code, prefix and line do not start with a 0 would be entered (although 0 may appear at other positions).

Problem 4: The money that a person will receive in the future is worth less than the same amount of money the person has on hand today. The present value of a payment $amount$ to be received n years from now is given by the formula

$$PV = \frac{amount}{(1+rat\ )^n}$$

where amount is the dollar amount to be received as payment, and rate is the interest rate.

A person wins a raffle. The prize is amount in $, but the person will receive the prize after n years form the day he or she won.

Write a program to asks the user to enter from the keyboard

- Name, $name$
- Prize in $ (real number) $amount$
- Interest rate $rate$ in %, real number
- Number years , *n,* integer

Your program computes and prints the present value of the payment the person would receive the amount after n years have passed.

Sample output:

```
Program calculates present value of monetary prize.
Winner: Jane Doe
Enter prize amount in $: 100.00
interest rate in %: 1
Enter number years: 2
   The present value of the prize for Jane Doe after  2  years is $98.03
```