Analysis:

The program is developed according to the EVA design pattern. Input and output are done in the console. To calculate with other values (parameters), the values must be changed in `ue01.py`. Only integers or floats are allowed. Other data types will result in an error. Decomposition: Implementation as functions. Output: Output is done in the console. Output is a float.

Testing the program:

Test Case	epr	gpr	zbnp	num1	num2	Result
task2(110, 110, 50), task3(2, 4)	110	110	50	2	4	12.5, min Value:2, is divisible by 2: True, is divisible by 4: False, is divisible by 8: False
task2(120, 110, 50), task3(4, 5)	120	110	50	4	5	Invalid input: epr and gpr must be less than 110!,min Value:4, is divisible by 2: True, is divisible by 4: True, is divisible by 8: False
task2(80, 40, 50), task3(7, 8)	80	40	50	7	8	8.6, min Value:7, is divisible by 2: False, is divisible by 4: False, is divisible by 8: False

Description of the program:

The program does not require any further libraries, no external files/functions/etc. The program is written and tested in Python 3.9.6. A Python interpreter must be installed. The program can be started with the command “python3 main.py” in the terminal. The Output is done in the console. No bugs are known. The results of the two defined functions are printed to the console inside the main function.

Task2:

Function uses the given formula to calculate the result. If either epr or gpr is greater than 110, the input is invalid as the maximum achievable bonus points are 110. ## Task3: ========== Calculates the minimum number given to the function via the built-in function min(). Then I declare a dictionary with keys of 2, 4 and 8 and assign each of them to False.

If the minimal number is divisible by either 2,4 and 8, the corresponding value of the key is set to True. The dictionary is returned.