**eEvery Boilermaker Engineering Code – Entry Level Programming**

**Week 4 – Programming Exercises**

1. **(15 points, Falling Distance)** When an object is falling because of gravity, the following formula can be used to determine that the distance the object falls in a specific time period:

**d = 1/2gt2**

The variables in the formula are as followings: d is the distance in meters, g is 9.8, and t is the amount of time, in seconds, that the object has been falling.

Write a function name **falling\_distance** that accepts an object’s falling time (in seconds) as an argument. The function should return the distance, in meters, that the object has fallen during that time interval. Write a program that calls the function in a loop that passes the values 1 through 10 as arguments and displays the return value. (**format the precision of falling distances as 2.**)

**Use the following numbers to test (Only type in the red numbers in interactive mode):**

|  |  |
| --- | --- |
| **Sample Input** | **Expected Output** |
|  | **Time(s) Falling Distance(m)**  **-----------------------------**  **1 4.90**  **2 19.60**  **3 44.10**  **4 78.40**  **5 122.50**  **6 176.40**  **7 240.10**  **8 313.60**  **9 396.90**  **10 490.00** |

1. **(10 points Maximum of Two Values)** Write a function named max that accepts two integer values as arguments and returns the value that is the greater of the two. For example, if 7 and 12 are passed as arguments to the function, the function should return 12. Use the function in a program that prompts the user to enter two integer values. The program should display the value that is the greater of the two.

**Use the following numbers to test (Only type in the red numbers in interactive mode):**

|  |  |
| --- | --- |
| **Sample Input** | **Expected Output** |
| **Enter the first integer: 109**  **Enter the second integer: 2165** | **2165 is greater.** |
| **Enter the first integer: 7**  **Enter the second integer: 1.5** | **1.5 is not an integer.** |

1. **(20 points, Test Average and Grade)** Write a program that ask the user to enter five valid test scores. The program should display a letter grade for each score and the average test score. Write the following functions in the program:
   * **get\_valid\_score:** This function should accept a score as argument, then validate the score (if the score is less than 0 or greater than 100, then ask user re-enter a score, until user enter a valid one), finally the function returns the valid input.
   * **calc\_average.** This function should accept five test scores as arguments and return the average of the score. (f**ormat the precision of average score as 2.**)
   * **determine\_grade.** This function should accept a test score as an argument and return a letter grade for the score base on the following grading scale.

|  |  |
| --- | --- |
| Score | Letter Grade |
| 90-100 | A |
| 80-89 | B |
| 70-79 | C |
| 60-69 | D |
| Below 60 | F |

**Use the following numbers to test (Only type in the red numbers in interactive mode, the blue part in the input should be printed by your code):**

|  |  |
| --- | --- |
| **Sample Input** | **Expected Output** |
| **Enter a score: 89**  **Enter a score: 75**  **Enter a score: 91**  **Enter a score: 82**  **Enter a score: 27** | **The letter grade for 89 is B.**  **The letter grade for 75 is C.**  **The letter grade for 91 is A.**  **The letter grade for 82 is B.**  **The letter grade for 27 is F.**  **The average test score is 72.80.** |
| **Enter a score: -10**  **Invalid Input. Please try again**  **Enter a score:101**  **Invalid Input. Please try again**  **Enter a score:97**  **Enter a score: 100**  **Enter a score: 0**  **Enter a score: 109**  **Invalid Input. Please try again**  **Enter a score:82**  **Enter a score: 79** | **The letter grade for 97 is A.**  **The letter grade for 100 is A.**  **The letter grade for 0 is F.**  **The letter grade for 82 is B.**  **The letter grade for 79 is C.**  **The average test score is 71.60.** |

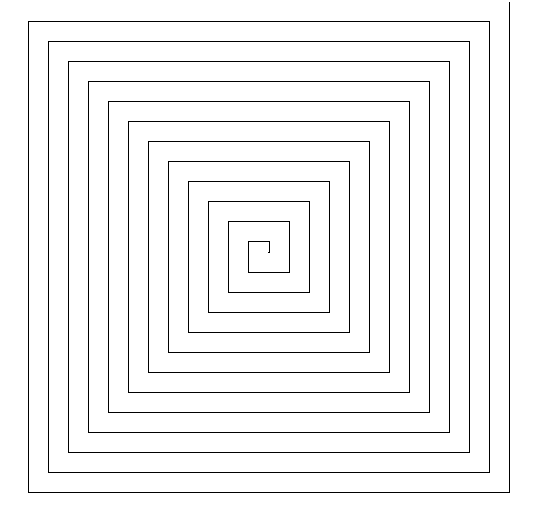
1. **(15 points, Prime Numbers)** A prime number is a number that is only evenly divisible by two numbers: itself and 1. For example, the number 5 is prime because it can only be evenly divided by 1 and 5. The number 6, however, is not prime because it can be divided evenly by 1, 2, 3, 6. Write a Boolean function named **is\_prime** which takes an integer as an argument and returns true if the argument is prime number, or false otherwise. Use the function in a program that prompt the user to enter a number then displays a message indicating whether the number is prime. Then use **is\_prime** function to displays all the prime numbers from 1 to 100. (Hint: the program should have a loop that calls the **is\_prime** function.)

**Please submit two screenshots:**

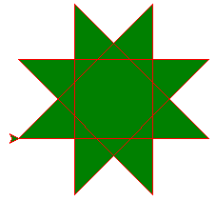
1. **Use the following numbers to test the part (a) of the question (Only type in the red numbers in interactive mode):**

|  |  |
| --- | --- |
| **Sample Input** | **Expected Output** |
| **Enter an integer: 1** | **1 is not a prime number.** |
| **Enter an integer: 2** | **2 is a prime number.** |
| **Enter an integer: 10** | **10 is not a prime number.** |
| **Enter an integer: 3001** | **3001 is a prime number.** |

1. **Please submit the screenshot with the list of all prime numbers from 1 to 100.**
2. **(10 points)** Use a loop with the turtle graphics library to draw the following Hypnotic Pattern. You can set the width between two rings as 10.



1. **(Bonus question: 15 points)** Use a loop with the turtle graphics library to draw the following Start Pattern with a user input number of corners:



|  |  |
| --- | --- |
| **Sample Input** | **Expected Output** |
| **Enter the number of corners: 8** |  |
| **Enter the number of corners: 24** |  |