NAME:

Open book and open notes. No calculators, no computer. Write all answers on these sheets. Do not ask questions!

<table>
<thead>
<tr>
<th>question</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>points</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>maximum</td>
<td>20</td>
<td>10</td>
<td>10</td>
<td>15</td>
<td>20</td>
<td>15</td>
<td>10</td>
<td>100</td>
</tr>
</tbody>
</table>

1. Consider the modular structure of a program to create posters in three stages: painting background colors, inserting pictures, and printing letters. For each of these three stages we have in and out functions. The in function takes instructions for each stage, while the out function renders the design on paper.

We can design the program in two ways:

(a) There are two modules: input and output. The input module collects all in functions for each stage. All out functions are contained in the module output.

(b) There is a module for each stage. Each module contains the specific in and out functions for the stage.

Which design would be best? Justify using the principles of good modular design.
2. What does “to copyleft software” mean? Who can do this? What are the consequences of this?

3. The memory consumption of an algorithm grows like $O(n^2)$ where $n$ is the dimension of the input. Suppose the algorithm requires 0.5 Gigabyte of memory to process inputs of dimension $n = 1000$. If we have only two Gigabytes of memory available, how large can $n$ be? Justify your answer.

4. Give Python code to define the following function:

   ```python
   def GetValue():
       """
       Asks the user for a floating-point number. As long as the user input
cannot be converted into a float, an error message is printed and the user
is invited to try again. The function returns the float given by the user.
       """
   ```
5. Write a program that prompts the user for the name of a Python script. The program counts all the print statements and appends all arguments of each print to a list. After reading the entire script, the program writes the number of print statements and the list of all arguments that occurred in a print.

The length of the list must equal the number of print statements, so arguments of one print separated by spaces must be joined. You may assume the script has at most one statement per line, but do not make any assumptions about spaces or tabs.
6. Draw a use case diagram to distinguish the actions on an object of the class car which can be performed by two different types of users: passenger and driver. Indicate also those object data attributes of a car, most relevant to the distinction between its users.

7. A circle is determined by three parameters: the two coordinates $x$ and $y$ of its center, and its radius $r$. Draw the design of a GUI to visualize circles for various values of the three parameters $x$, $y$, and $r$. Label each component of the GUI with the proper name of the widget. Do not give any Python code.