

Item	Possible Points	Earned Points	Notes
<p>Program design for Project 4</p> <ul style="list-style-type: none"> <li>• Program design is appropriate for the specifications</li> <li>• A correct class diagram is provided for all classes (not needed for the main program class and not required for this project)</li> <li>• Design documentation reflects actual logic of code</li> <li>• All methods are documented</li> <li>• No diagram is larger than one page (8 ½ by 11 inches with ½ inch margins on all sides)</li> <li>• If using flowcharts to diagram the logic: <ul style="list-style-type: none"> <li>○ Each flowchart begins and ends with a terminator symbol Note: the main method beginning terminator contains the word <code>main()</code>. The main method ending terminator contains the word <code>return</code>. Because you do not write the code that calls the main method, you will not have any flowcharts where the beginning terminator contains the word <code>START</code> and the ending terminator contains the word <code>END</code>.</li> <li>○ The appropriate symbol is used</li> <li>○ Only one task per process symbol (the rectangle); each variable declaration should be in its own symbol; show the entire formula for calculations</li> <li>○ Every symbol (except a terminator) has at least one flowline leading to it and one and only one flowline leading from it.</li> </ul> </li> <li>• If using structured pseudocode to diagram the logic: <ul style="list-style-type: none"> <li>○ The pseudocode is appropriately indented</li> <li>○ Each variable declaration is on its own line</li> <li>○ The entire formula is shown for calculations</li> <li>○ Selection and iteration blocks have a clear beginning and ending</li> </ul> </li> <li>• If using Warnier Diagrams to diagram the logic: <ul style="list-style-type: none"> <li>○ Braces are appropriately labeled</li> <li>○ Each variable declaration is on its own line</li> <li>○ The entire formula is shown for calculations</li> </ul> </li> </ul>	10	8	(-2) The logic diagram does not show the loop for the factorial calculation. for loops are diagramed like while loops.
<p>Following coding standards:</p> <ul style="list-style-type: none"> <li>• Code restricted to 80 columns</li> <li>• Follows naming conventions for classes, variables, methods, and constants</li> <li>• Appropriate comment block at top of program file (may use javadoc conventions)</li> <li>• Methods appropriately commented (may use javadoc conventions)</li> <li>• Variables have meaningful names</li> <li>• Braces align correctly</li> <li>• Control statements formatted correctly</li> </ul>	4	3	(-1) Braces and code not correctly aligned.

Item	Possible Points	Earned Points	Notes
<b>Code:</b> <ul style="list-style-type: none"> <li>• Reflects material covered through Chapter 4</li> <li>• The factorial calculation is done in a static method (not in <code>main</code>)</li> <li>• Uses a <code>for</code> loop to calculate the factorial</li> <li>• Good coding style</li> </ul>	<b>6</b>	4	(-2) Factorial is calculated in <code>main()</code> , not in a separate method.
<b>Penalties:</b> <ul style="list-style-type: none"> <li>• Incorrect calculations</li> <li>• No proof that the calculations are correct</li> <li>• Some test data not used</li> <li>• Output is not presented as shown (including spelling and spacing)</li> <li>• Reflects material outside what has been covered through Chapter 4</li> </ul>	<b>-20</b>	<b>-2</b>	(-2) No proof.
<b>Total</b>	<b>20</b>	<b>13.0</b>	