

2. The `Bottle` class, which you will write, represents a bottle that contains liquid.

`Bottle` objects are created by calls to a constructor with a `double` parameter that represents the bottle's capacity, in milliliters (mL), which is the maximum amount of liquid in the bottle. Assume that this value will be greater than or equal to 0. When `Bottle` objects are constructed, each is filled to its capacity.

The `Bottle` class contains an `updateAmount` method, which updates the amount of liquid in the bottle. This method has a `double` parameter that represents the amount of liquid, in mL, to be removed from the bottle, with a precondition that the parameter is always greater than zero and less than or equal to the amount of liquid remaining in the bottle. If updating the amount of liquid in the bottle would cause it to be less than 25% of the capacity, the bottle is filled and reset to the capacity. The `updateAmount` method returns a `double` that represents the amount remaining in the bottle, in mL, after the amount has been updated.

The following table contains a sample code execution sequence and the corresponding results. The code execution sequence appears in a class other than `Bottle`.

Statement	Return Value (blank if no value)	Explanation
<code>double amt;</code>		
<code>Bottle water = new Bottle(1000.0);</code>		A <code>Bottle</code> object named <code>water</code> is constructed with a capacity of 1,000 mL of liquid.
<code>amt = water.updateAmount(400.0);</code>	600.0	400 mL are used, so 600 mL remain in the bottle.
<code>amt = water.updateAmount(100.0);</code>	500.0	100 mL are used, so 500 mL remain in the bottle.
<code>amt = water.updateAmount(300.0);</code>	1000.0	300 mL are used, so 200 mL remain in the bottle. Since this amount is less than 250 mL (25% of the capacity of 1,000 mL), the bottle is immediately filled and reset to the initial amount of 1,000 mL.
<code>Bottle shampoo = new Bottle(40.0);</code>		A <code>Bottle</code> object named <code>shampoo</code> is constructed with a capacity of 40 mL of liquid.
<code>amt = shampoo.updateAmount(30.0);</code>	10.0	30 mL are used, so 10 mL remain in the bottle.
<code>amt = shampoo.updateAmount(1.0);</code>	40.0	1 mL is used, so 9 mL remain in the bottle. Since this amount is less than 10 mL (25% of the capacity of 40 mL), the bottle is immediately filled and reset to the initial amount of 40 mL.

Write the complete `Bottle` class. Your implementation must meet all specifications and conform to the examples shown in the table.