

1. (a) (15 points) Write a class called `IceCream` that represents an order of ice cream. An `IceCream` starts with no scoops of ice cream, but scoops of various flavors can be added in later.

The `IceCream` class should have the following methods with the described arguments. You should not include any additional arguments for these methods:

Method	Description
<code>__init__(self)</code>	Creates an empty <code>IceCream</code> .
<code>add_scoop(self, flavor, scoops)</code>	Adds <code>scoops</code> scoops of the given flavor to this <code>IceCream</code> .
<code>get_scoops(self, flavor)</code>	Returns the number of scoops in this <code>IceCream</code> that are of the given flavor. Returns 0 if the flavor is not in this <code>IceCream</code> .
<code>__eq__(self, other)</code>	Returns <code>True</code> if this <code>IceCream</code> has the same set of flavors as the other. Note that the number of scoops doesn't count. Returns <code>False</code> if they are not the same.

Write your response in the box on the next page.

- (b) (5 points) In the box below, write a short program that constructs an `IceCream` objects and then adds 3 scoops of vanilla ice cream and 4 scoop of chocolate. It should then print out a message of the format

```
"Vanilla: <vanilla>, Chocolate: <chocolate>"
```

Where the values in angle brackets are found by calling the `get_scoops` method.

You do not need to write a main method for this problem, you may write the lines of code directly in the space below.

