One of the professors in the CS department has decided to retire from the university and start her own dairy farm. She has already purchased some prime grazing land, restored an old barn, and acquired some cows. And, being a CS professor, she has decided to write a Java program to help. She has started to write Dairy.java, shown below.

```java
/** The Dairy class keeps track of a dairy herd and the profitability of producing
 * either milk or butter. */
public class Dairy {

   /** Instance data member to keep track of the number of cows */

   /**
   * Constructs a new dairy with initialCows cows
   * @param initialCows the initial number of cows in the herd
   */
   public Dairy(int initialCows) {
   }

   /**
   * Adds one to the number of cows in the herd
   */
   public void addCow() {
   }

   /**
   * Returns the number of cows currently in the herd
   * @return the number of cows
   */
   public int getNumCows() {
   }

   /**
   * Returns the profit of selling the herd’s milk
   * (assuming each cow produces 10 gallons of milk)
   * @param pricePerGallon the price received for a gallon of milk
   * @return the total profit from selling milk
   */
   public double calcMilkProfit(double pricePerGallon) {
      return 10.0 * pricePerGallon * getNumCows();
   }

   /**
   * Returns the profit of selling butter made from the herd’s milk
   * (assuming each cow produces enough milk to make 4 pounds of butter)
   * @param pricePerPound the price received for a pound of butter
   * @return the total profit from selling butter
   */
   public double calcButterProfit(double pricePerPound) {
      return 4.0 * pricePerPound * getNumCows();
   }
}
```

Help our recently departed colleague finish this class, with the following guidelines:

- Add an instance field to keep track of the size of the professor’s herd.
- The constructor should initialize this field to the value of the parameter `initialCows`.
- `addCow()` and `getNumCows()` should do what the Javadoc comments describe.
Now that the Dairy class is complete, we can use it to keep track of the professor’s dairy in an application class. She has added the basic structure already.

```java
public class ProfessorsDairy {
    public static void main(String[] args) {
    }
}
```

Complete this application, with the following steps:
- Create a Dairy object called profDairy with 8 initial cows.
- Add two cows.
- Get the current number of cows and print it.
- Print the professor’s profit if she sells her cow’s milk for $1.10/gallon.
- Print the professor’s profit if she turns the milk into butter and sells it for $2.50/pound.

Questions
1. Is `addCow()` an example of an accessor method or a mutator method?
2. Is `getNumCows()` an example of an accessor method or a mutator method?
3. How can we verify that the Dairy class is working correctly?
4. What is the output of the application program?

5. Given the current prices, should the professor sell milk or butter?

6. The Dairy class is relatively simple. Name two additional features that you could add that would make it more useful.

7. What assumptions are we making about the parameters in the constructor and the methods in Dairy.java? (Hint: What happens if someone creates a Dairy object with -30 initial cows?)