

```
/**
 * A LeadedPencil has a default length: When the constructor is called with a
 * non-positive length, the LeadedPencil is initialized to have the default
 * length.
 *
 * @convention length > 0
 * @correspondence s = length <br /> c = color
 */
public class LeadedPencil implements Pencil {

    /**
     * Default value used to initialize length if constructor is not called with
     * a positive value.
     */
    private static final int DEFAULT_LENGTH = 10;

    /**
     * Color of the lead in this pencil. The lead can always be replaced with
     * different color.
     */
    private Colors color;

    /**
     * The length of the pencil. This length is always positive.
     */
    private int length;

    /**
     * Initializes the LeadedPencil with the given length, unless the length is
     * not positive, in which case the default length is used.
     *
     * @param color
     *         the initial lead color
     * @param length
     *         the initial length of the pencil
     */
    public LeadedPencil(Colors color, int length) {
        this.color = color;
        this.length = DEFAULT_LENGTH;
        if (length > 0) {
            this.length = length;
        }
    }

    /**
     * Returns a human-readable representation of the Pencil.
     */
    public String toString() {
        return (this.length + ": " + this.color);
    }

    /**
     * @param newColor
     *         is a valid color
     */
    public void setColor(Colors newColor) {
        this.color = newColor;
    }

    /**
     * @param remove
     *         amount by which pencil will be shortened
     */
    public void sharpen(int remove) {
```

```
    assert (remove >= 0);  
    if (remove < this.length) {  
        this.length -= remove;  
    }  
}
```