

```
import java.util.Random;

/**
 * A random number generator that returns a number within the specified bound
 * and of alternating parity. The number returned from the first call is even,
 * then odd, then even and so on.
 *
 * @convention numCalls >= 0
 * @correspondence numCalls = |history|
 * @author paolo
 */
public class UnfilteredRandom implements RandomWithParity {

    /**
     * Number of times the generateNumber method has been called.
     */
    private int numCalls;

    /**
     * Generates a stream of random numbers.
     */
    private Random generator;

    /**
     * Initially, no calls have been made.
     *
     * @ensures numCalls = 0
     */
    UnfilteredRandom() {
        this.numCalls = 0;
        this.generator = new Random();
    }

    /**
     * Generates a random number in the specified interval.
     *
     * @param upperBound an upper limit on the number to be returned
     * @requires upperBound >= 1
     * @alters numCalls
     * @ensures numCalls = #numCalls + 1
     * @return 0 <= generateNumber <= upperBound </br>
     * generateNumber is even <==> #numCalls is even
     */
    public int generateNumber(int upperBound) {
        int randomNumber;
        do {
            randomNumber = this.generator.nextInt(upperBound+1);
        } while (randomNumber % 2 != this.numCalls % 2);
        this.numCalls++;
        return randomNumber;
    }
}
```