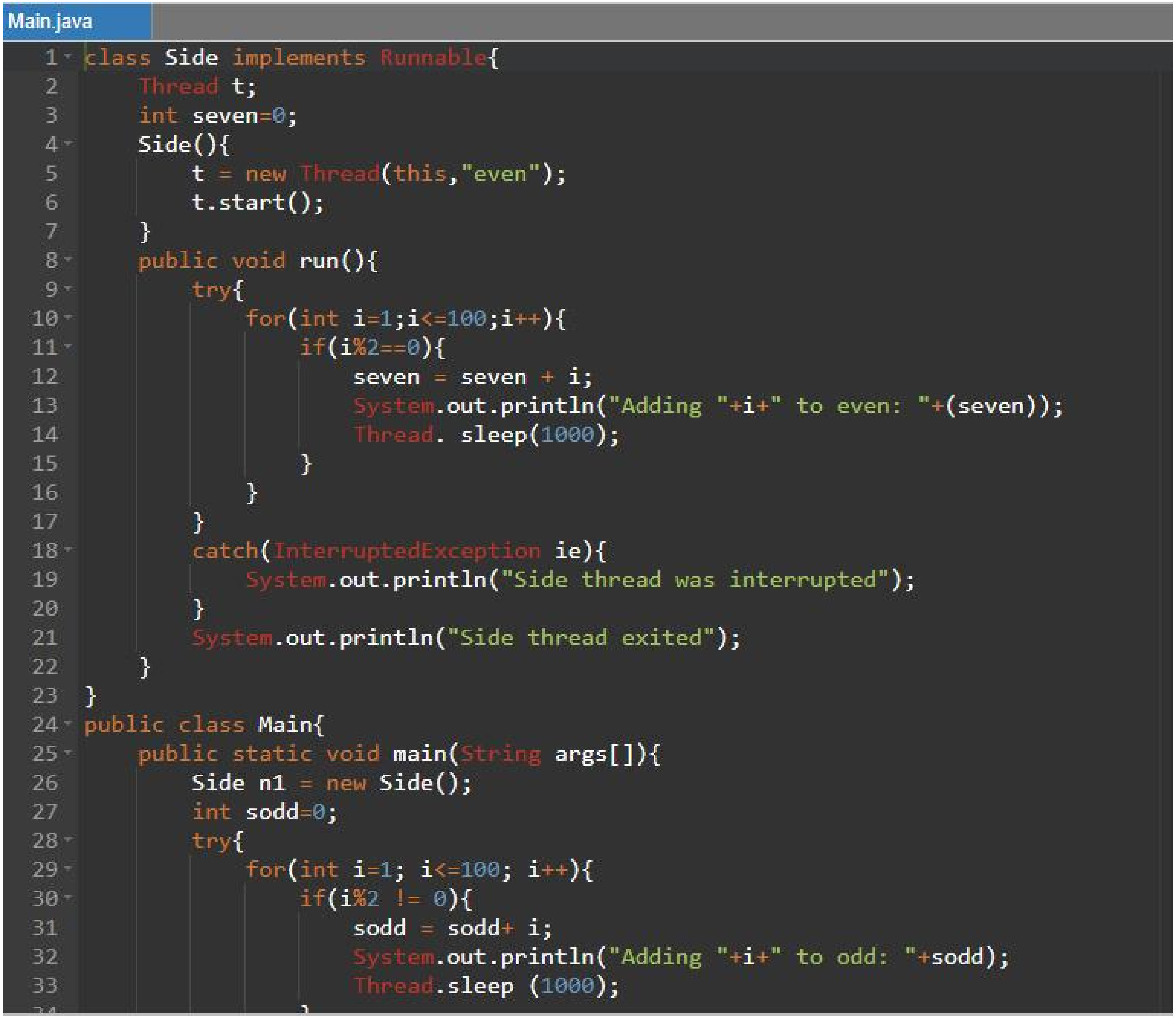
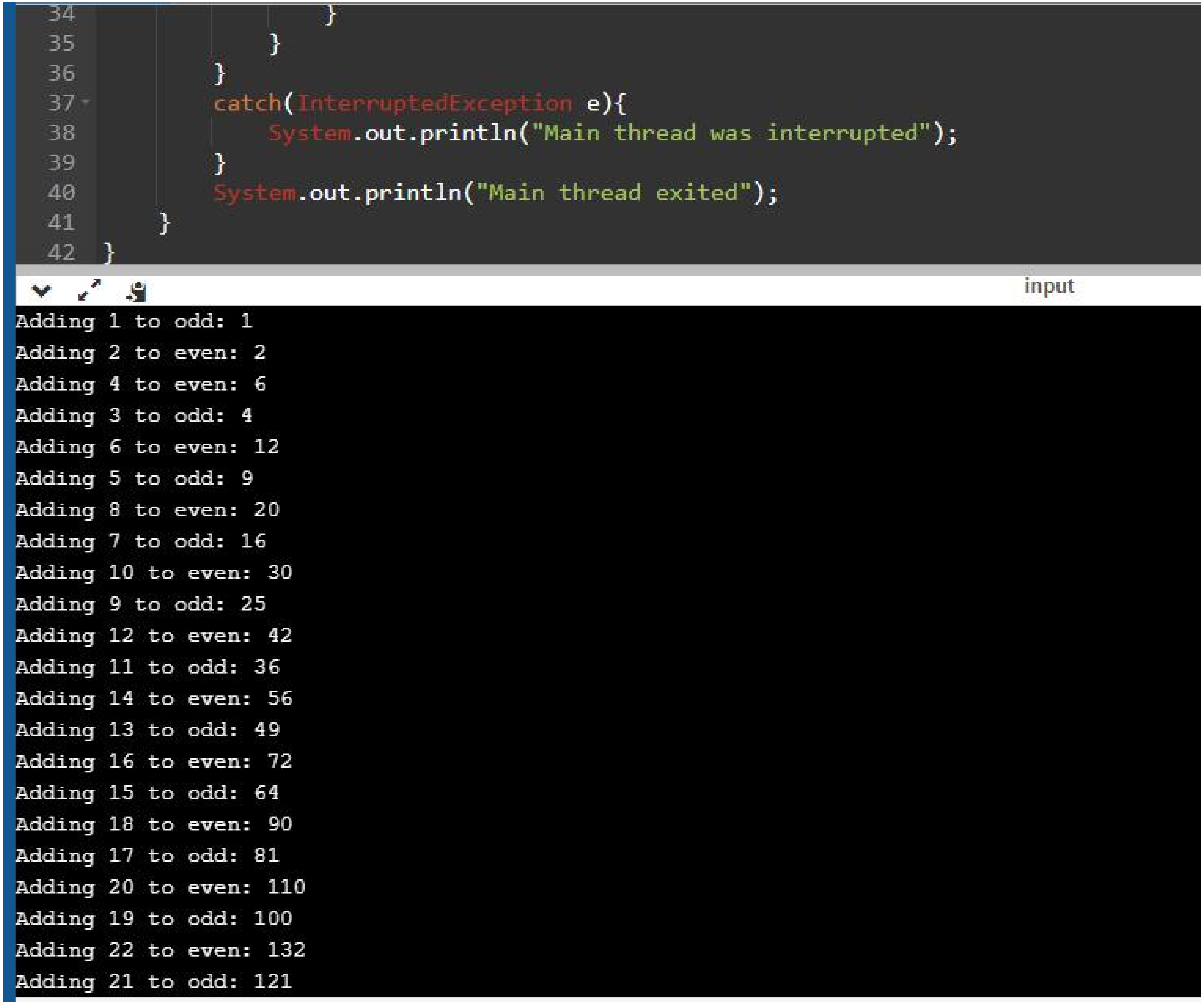
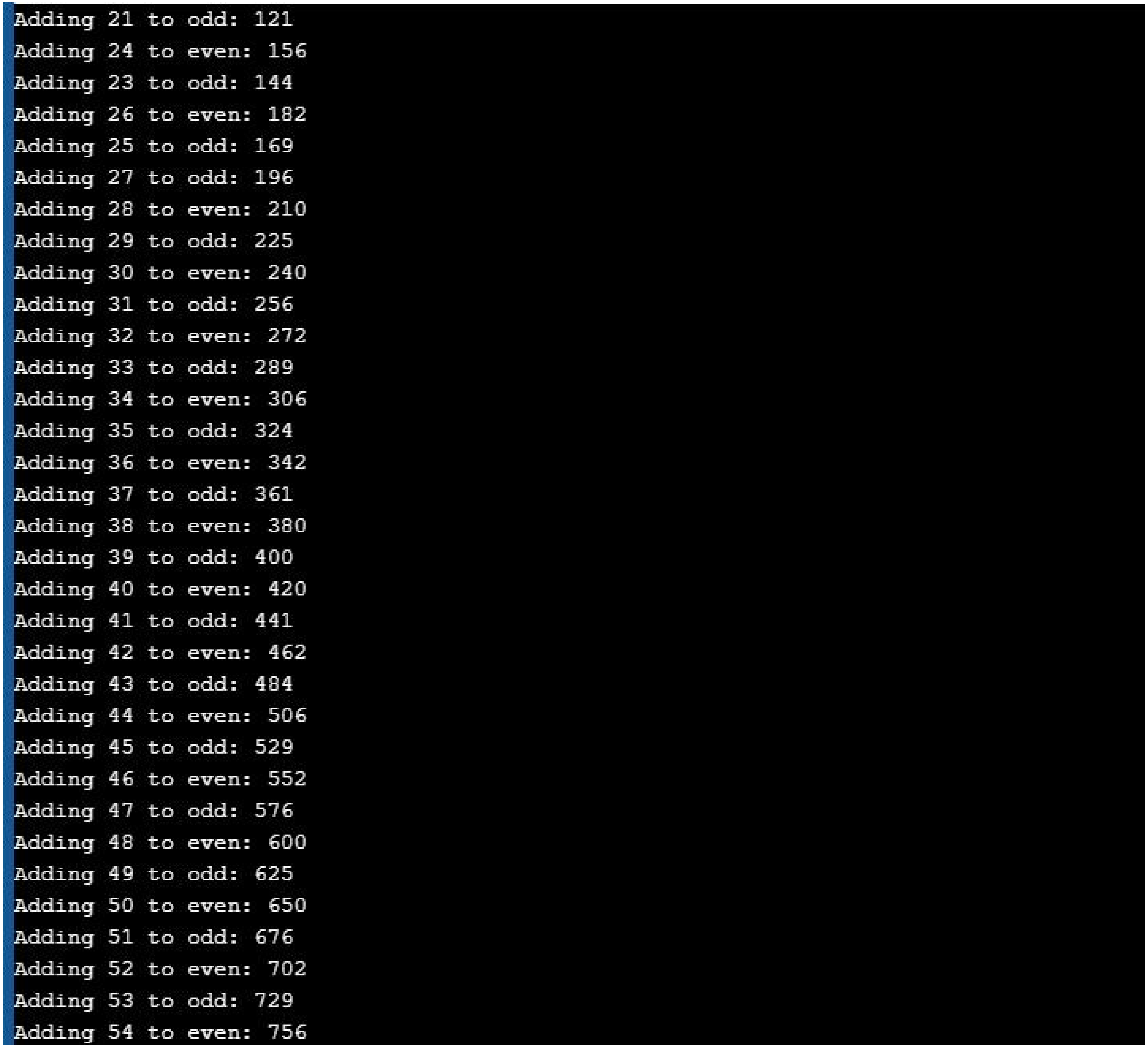


```java
class Side implements Runnable{
    Thread t;
    int seven=0;
    Side(){
        t = new Thread(this,"even");
        t.start();
    }
    public void run(){
        try{
            for(int i=1;i<=100;i++){
                if(i%2==0){
                    seven = seven + i;
                    System.out.println("Adding "+i+" to even: "+(seven));
                    Thread. sleep(1000);
                }
            }
        }
        catch(InterruptedException ie){
            System.out.println("Side thread was interrupted");
        }
        System.out.println("Side thread exited");
    }
}
public class Main{
    public static void main(String args[]){
        Side n1 = new Side();
        int sodd=0;
        try{
            for(int i=1; i<=100; i++){
                if(i%2 != 0){
                    sodd = sodd+ i;
                    System.out.println("Adding "+i+" to odd: "+sodd);
                    Thread.sleep (1000);
                }
```

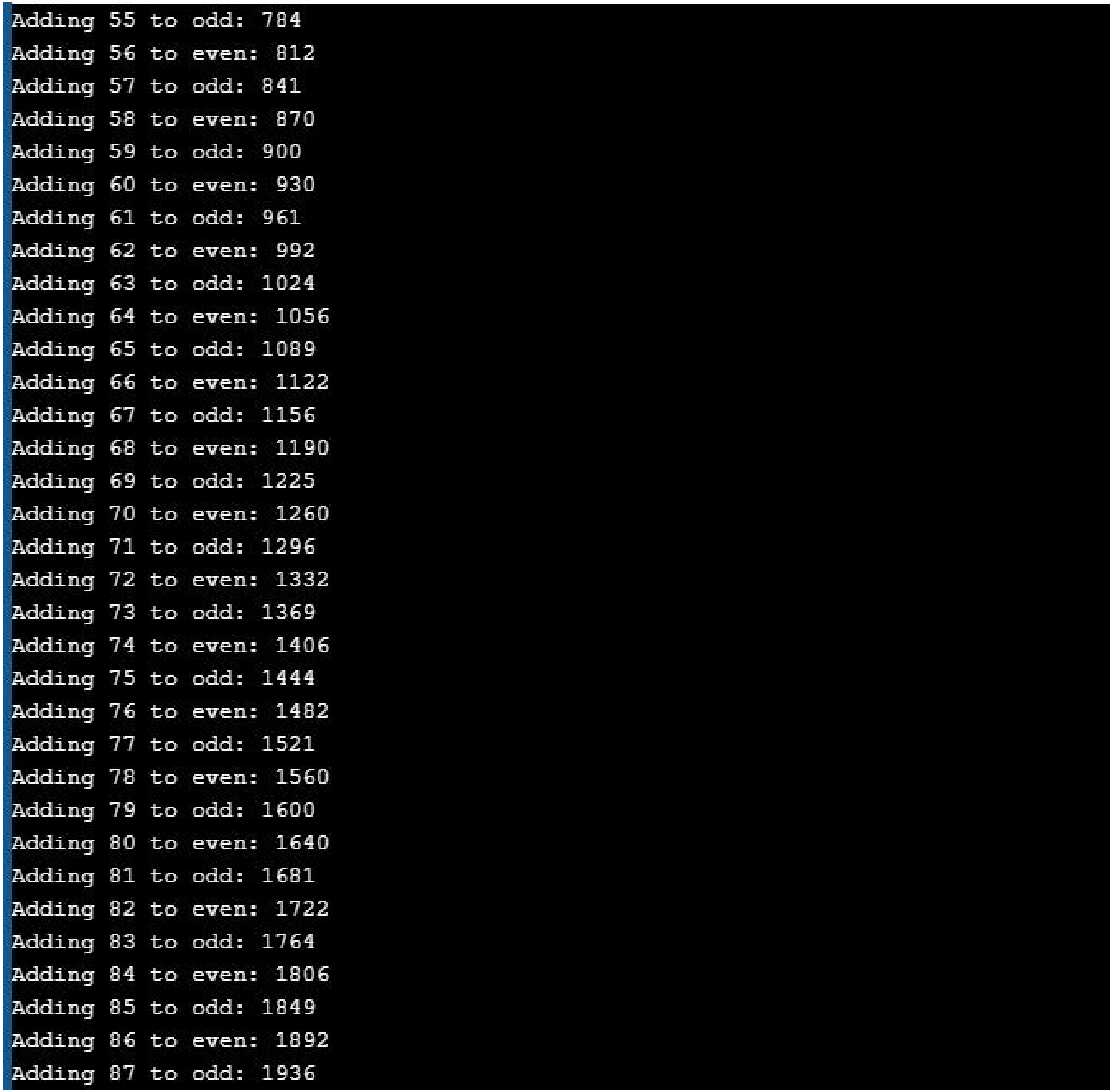


```java
                }
            }
        }
        catch(InterruptedException e){
            System.out.println("Main thread was interrupted");
        }
        System.out.println("Main thread exited");
    }
}
```

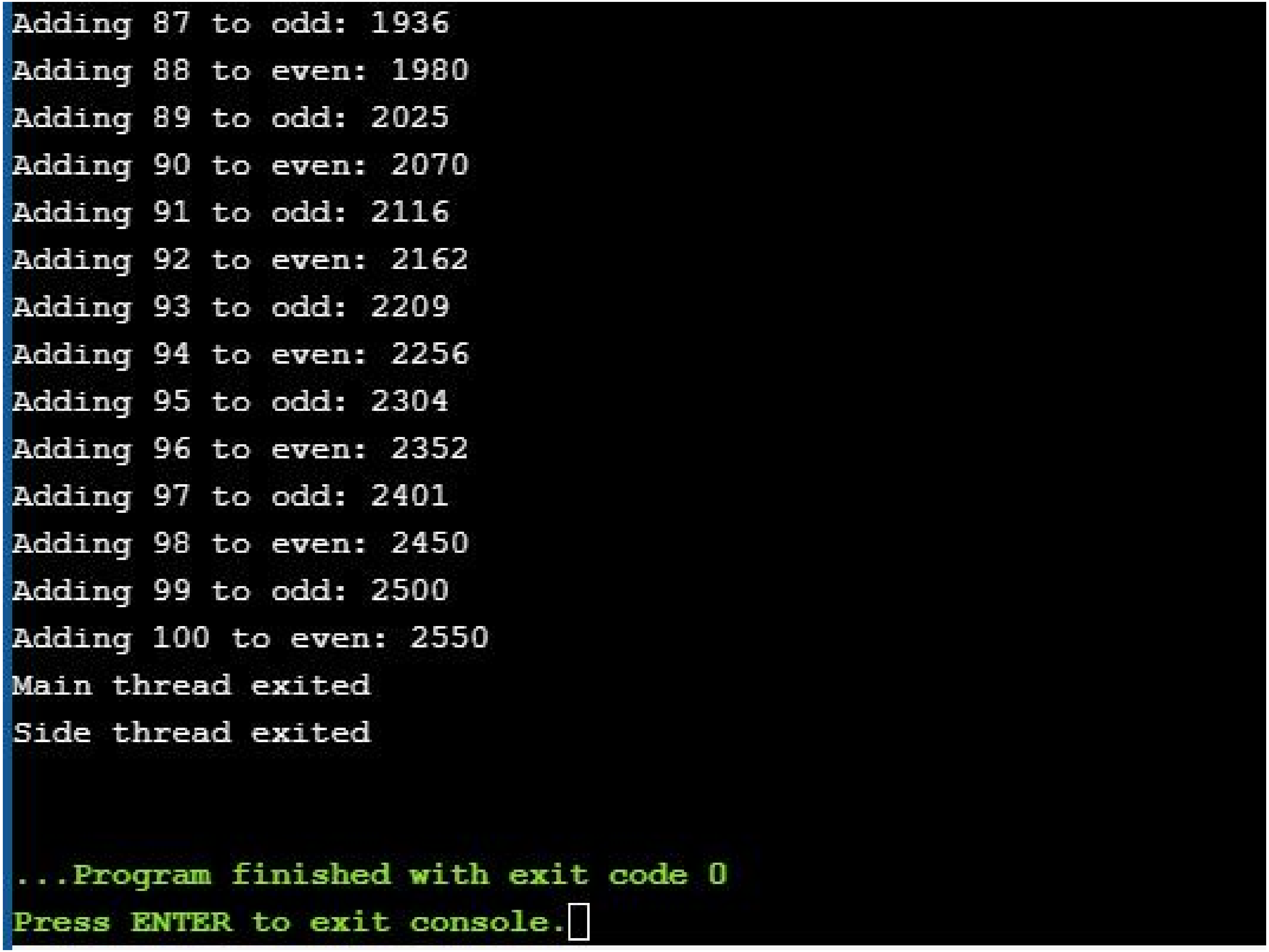
```
Adding 1 to odd: 1
Adding 2 to even: 2
Adding 4 to even: 6
Adding 3 to odd: 4
Adding 6 to even: 12
Adding 5 to odd: 9
Adding 8 to even: 20
Adding 7 to odd: 16
Adding 10 to even: 30
Adding 9 to odd: 25
Adding 12 to even: 42
Adding 11 to odd: 36
Adding 14 to even: 56
Adding 13 to odd: 49
Adding 16 to even: 72
Adding 15 to odd: 64
Adding 18 to even: 90
Adding 17 to odd: 81
Adding 20 to even: 110
Adding 19 to odd: 100
Adding 22 to even: 132
Adding 21 to odd: 121
```



```
Adding 21 to odd: 121
Adding 24 to even: 156
Adding 23 to odd: 144
Adding 26 to even: 182
Adding 25 to odd: 169
Adding 27 to odd: 196
Adding 28 to even: 210
Adding 29 to odd: 225
Adding 30 to even: 240
Adding 31 to odd: 256
Adding 32 to even: 272
Adding 33 to odd: 289
Adding 34 to even: 306
Adding 35 to odd: 324
Adding 36 to even: 342
Adding 37 to odd: 361
Adding 38 to even: 380
Adding 39 to odd: 400
Adding 40 to even: 420
Adding 41 to odd: 441
Adding 42 to even: 462
Adding 43 to odd: 484
Adding 44 to even: 506
Adding 45 to odd: 529
Adding 46 to even: 552
Adding 47 to odd: 576
Adding 48 to even: 600
Adding 49 to odd: 625
Adding 50 to even: 650
Adding 51 to odd: 676
Adding 52 to even: 702
Adding 53 to odd: 729
Adding 54 to even: 756
```

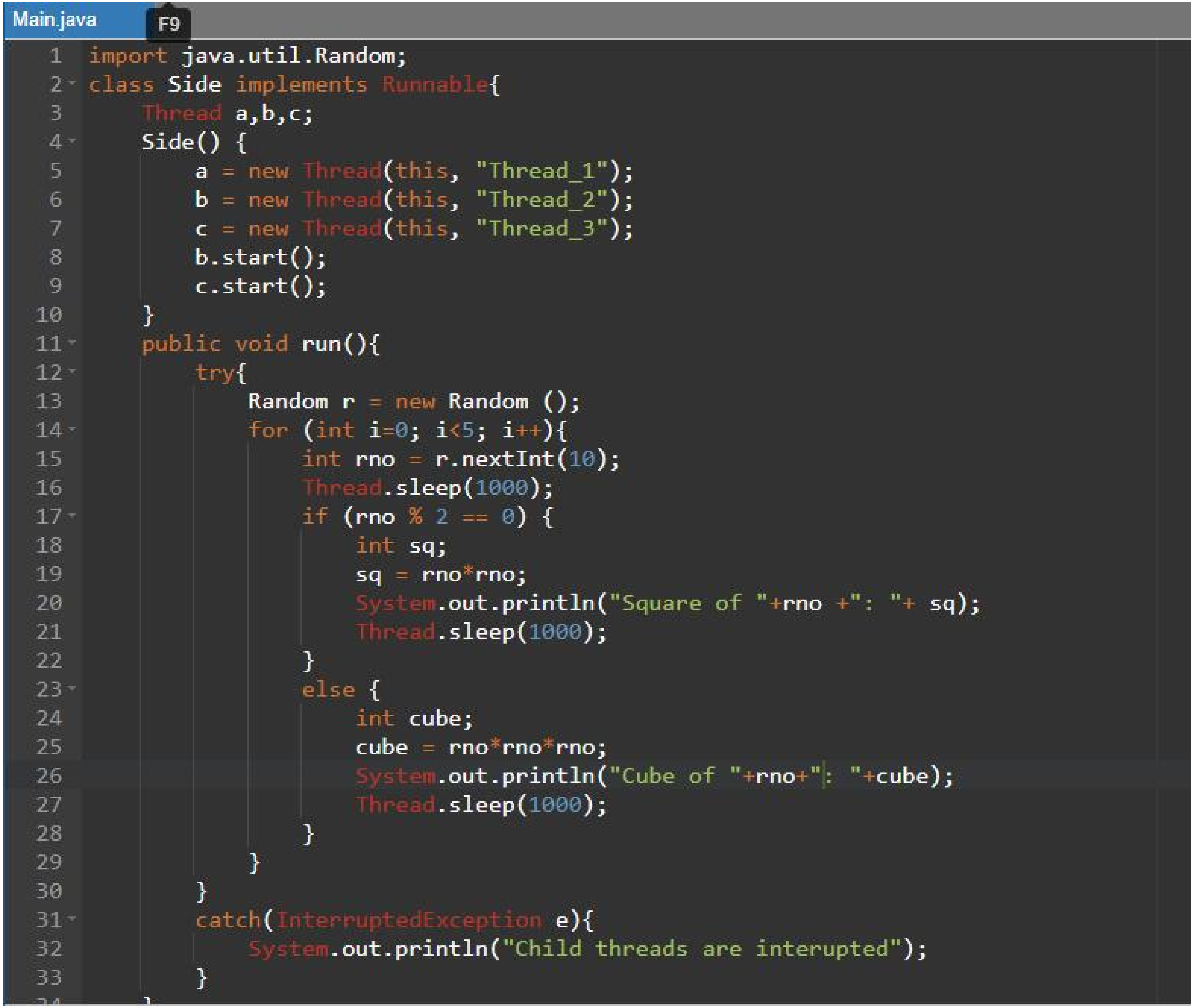


```
Adding 55 to odd: 784
Adding 56 to even: 812
Adding 57 to odd: 841
Adding 58 to even: 870
Adding 59 to odd: 900
Adding 60 to even: 930
Adding 61 to odd: 961
Adding 62 to even: 992
Adding 63 to odd: 1024
Adding 64 to even: 1056
Adding 65 to odd: 1089
Adding 66 to even: 1122
Adding 67 to odd: 1156
Adding 68 to even: 1190
Adding 69 to odd: 1225
Adding 70 to even: 1260
Adding 71 to odd: 1296
Adding 72 to even: 1332
Adding 73 to odd: 1369
Adding 74 to even: 1406
Adding 75 to odd: 1444
Adding 76 to even: 1482
Adding 77 to odd: 1521
Adding 78 to even: 1560
Adding 79 to odd: 1600
Adding 80 to even: 1640
Adding 81 to odd: 1681
Adding 82 to even: 1722
Adding 83 to odd: 1764
Adding 84 to even: 1806
Adding 85 to odd: 1849
Adding 86 to even: 1892
Adding 87 to odd: 1936
```

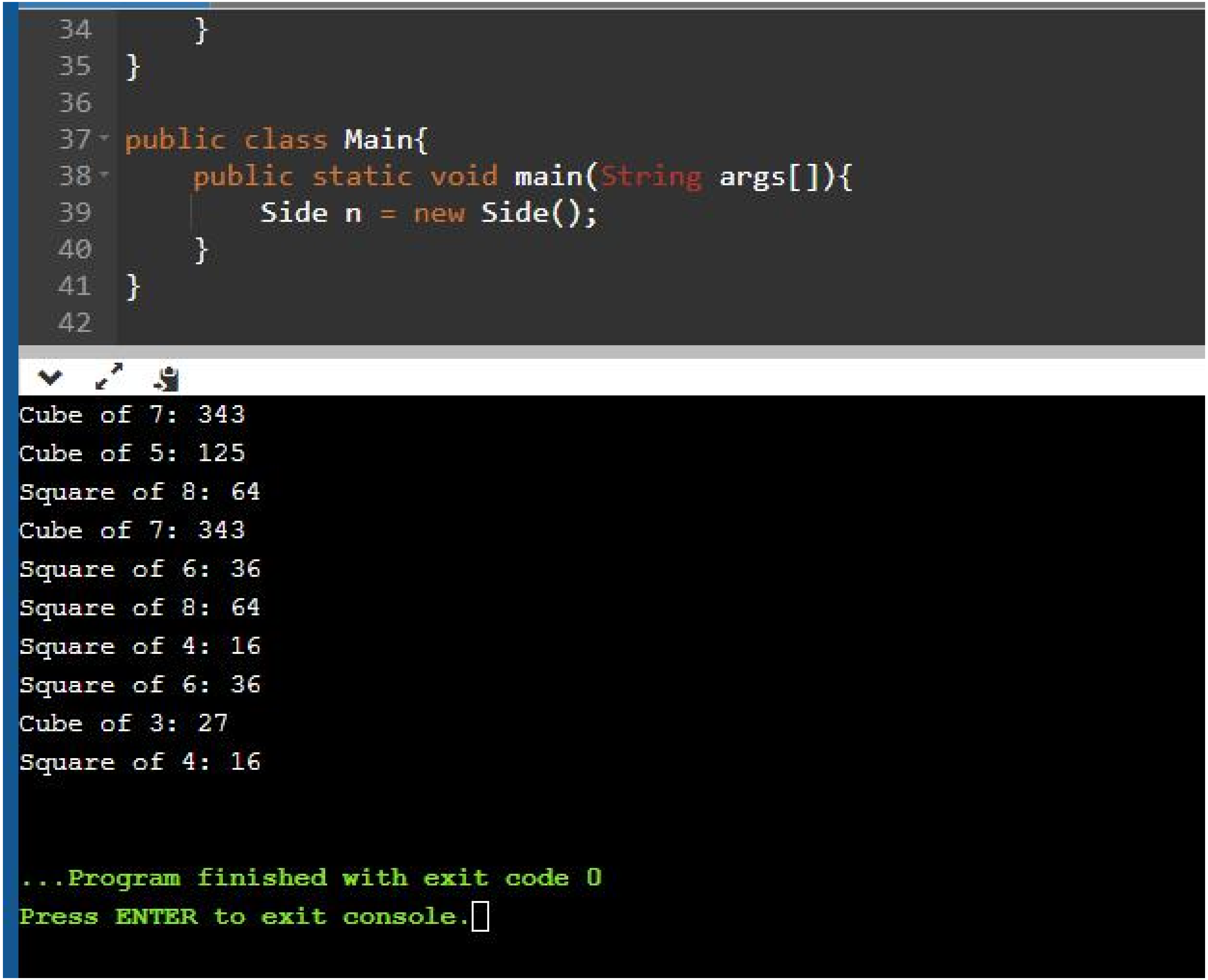


```
Adding 87 to odd: 1936
Adding 88 to even: 1980
Adding 89 to odd: 2025
Adding 90 to even: 2070
Adding 91 to odd: 2116
Adding 92 to even: 2162
Adding 93 to odd: 2209
Adding 94 to even: 2256
Adding 95 to odd: 2304
Adding 96 to even: 2352
Adding 97 to odd: 2401
Adding 98 to even: 2450
Adding 99 to odd: 2500
Adding 100 to even: 2550
Main thread exited
Side thread exited


...Program finished with exit code 0
Press ENTER to exit console.
```



```java
import java.util.Random;
class Side implements Runnable{
    Thread a,b,c;
    Side() {
        a = new Thread(this, "Thread_1");
        b = new Thread(this, "Thread_2");
        c = new Thread(this, "Thread_3");
        b.start();
        c.start();
    }
    public void run(){
        try{
            Random r = new Random ();
            for (int i=0; i<5; i++){
                int rno = r.nextInt(10);
                Thread.sleep(1000);
                if (rno % 2 == 0) {
                    int sq;
                    sq = rno*rno;
                    System.out.println("Square of "+rno +": "+ sq);
                    Thread.sleep(1000);
                }
                else {
                    int cube;
                    cube = rno*rno*rno;
                    System.out.println("Cube of "+rno+": "+cube);
                    Thread.sleep(1000);
                }
            }
        }
        catch(InterruptedException e){
            System.out.println("Child threads are interupted");
        }
    }
```



```java
    }
}

public class Main{
    public static void main(String args[]){
        Side n = new Side();
    }
}

```

```
Cube of 7: 343
Cube of 5: 125
Square of 8: 64
Cube of 7: 343
Square of 6: 36
Square of 8: 64
Square of 4: 16
Square of 6: 36
Cube of 3: 27
Square of 4: 16


...Program finished with exit code 0
Press ENTER to exit console.
```