

### **Robert Sedgewick**

斯坦福大学博士，导师为Donald E. Knuth，从1985年开始一直担任普林斯顿大学计算机科学系教授，曾任该系主任，也是Adobe Systems公司董事会成员，曾在Xerox PARC、国防分析研究所（Institute for Defense Analyses）和法国国家信息与自动化研究所（INRIA）从事研究工作。他的研究方向包括解析组合学、数据结构和算法的分析与设计、程序可视化等。

### **Kevin Wayne**

康奈尔大学博士，普林斯顿大学计算机科学系高级讲师，研究方向包括算法的设计、分析和实现，特别是图和离散优化。

# 数字版权声明

图灵社区的电子书没有采用专有客户端，您可以在任意设备上，用自己喜欢的浏览器和PDF阅读器进行阅读。

但您购买的电子书仅供您个人使用，未经授权，不得进行传播。

我们愿意相信读者具有这样的良知和觉悟，与我们共同保护知识产权。

如果购买者有侵权行为，我们可能对该用户实施包括但不限于关闭该帐号等维权措施，并可能追究法律责任。

表 1.2.14 一种能够累加数据的抽象数据类型（可视版本）

<b>API</b>	<code>public class VisualAccumulator</code>	
	<code>VisualAccumulator(int trials, double max)</code>	
	<code>void addDataValue(double val)</code>	添加一个新的数据值
	<code>double mean()</code>	所有数据的平均值
	<code>String toString()</code>	对象的字符串表示

#### 典型的用例

```
public class TestVisualAccumulator
{
    public static void main(String[] args)
    {
        int T = Integer.parseInt(args[0]);
        VisualAccumulator a = new VisualAccumulator(T, 1.0);
        for (int t = 0; t < T; t++)
            a.addDataValue(StdRandom.random());
        StdOut.println(a);
    }
}
```

#### 数据类型的实现

```
public class VisualAccumulator
{
    private double total;
    private int N;
    public VisualAccumulator(int trials, double max)
    {
        StdDraw.setXscale(0, trials);
        StdDraw.setYscale(0, max);
        StdDraw.setPenRadius(.005);
    }
    public void addDataValue(double val)
    {
        N++;
        total += val;
        StdDraw.setPenColor(StdDraw.DARK_GRAY);
        StdDraw.point(N, val);
        StdDraw.setPenColor(StdDraw.RED);
        StdDraw.point(N, total/N);
    }
    public double mean()
    public String toString()
    // 和 Accumulator 相同
}
```