

实验九 实现 Walker's Alias Method

思路

预处理方法

1. 首先计算每个元素的 $p_i = w_i / W$ 值，并将序列按照 p_i 值升序排列
2. 维护两个优先队列，一个升序排列 (p_small)，一个降序排列 (p_large)。 p_small 中储存所有小于平均水平 ($averWeight$) 的元素， p_large 中储存所有大于平均水平的元素。
3. 不断弹出两个优先队列中的元素，将 p_large 的元素“倒出” $1/n - p_i$ 给 p_small 的元素。在 p_small 里设立一个指针指向 p_large 。更新完后判断从 p_large 中出来的元素，如果大于平均值，则加入 p_large ，如果它小于平均值，则加入 p_small 中。
4. 循环结束条件为两个优先队列中任何一个为空。

查询

使用 $\text{random}(0, n-1)$ 随机生成一个 $[0, n-1]$ 的数，表示 $1/n$ 的概率选中一个位置 ($\text{random}(a, b)$ 为 $(\text{rand}() \% (b - a + 1)) + a$)；使用 $(\text{double})(\text{rand}()) / (\text{double})\text{RAND_MAX} / \text{CNT}$ 随机生成一个 $[0, 1/n]$ 的数，用于判断在该位置选择该元素还是其指针指向的元素。

说明

关于 Walker's Alias Method 的误差

当我们在理想情况下，通过概率计算出的权重是可以均分的。但是在复杂的情况下，使用 double 计算时就会存在精度问题，尤其当小数位数特别多的时候。直接导致的结果使 p_small 堆还存在元素，但是 p_large 堆已分配空的情况（反之亦然）。

解决方法，无论是 p_small 堆还是 p_large 堆，在循环分配后，若还存在未分配的元素，那么我们强行将其占比置 1。在实验测试阶段也发现最后留在队列中元素的值已经非常接近平均值了，在可接受范围内。

结果

运行时间不超过 0.5s.

```
"C:\大三上\学习\算法分析与设计\上机\9-Walker's Alias Method\9.exe"
read 'data9.txt' finish!
sort finish!
averWeight = 0.0001
write finish!

The selected number is : 7139

Process returned 0 (0x0)   execution time : 0.447 s
Press any key to continue.
```