



桶排序 (Bucket Sort)

【问题描述】

输入 n 个 0 到 100 之间的整数，由小到大排序输出。

【数据范围】

$$1 \leq n \leq 10000$$

【输入样例】

```
6
6 5 3 4 1 2
```

【输出样例】

```
1 2 3 4 5 6
```

【基本思想】

若待排序的值在一个明显有限范围内（整型）时，可设计有限个有序桶，待排序的值装入对应的桶，桶号就是待排序的值。

顺序输出各桶的值，将得到有序的序列。

【算法分析】

已知一组无序正整数数据 $a[1], a[2], \dots, a[n]$ ，需将其按升序排列。

首先定义一个数组 $b[m]$ ，且 $m \geq \max\{a[1], a[2], \dots, a[n]\}$ 。

接着循环 n 次，每次 $b[a[i]]++$ 。

【桶排序的性能】

优点：快，效率达到 $O(n)$ 。

缺点：数据范围必须为正整数并且比较小。

【参考程序】

```
#include<iostream>
```



```
using namespace std;
const int M = 100;
int b[M + 1];
int main()
{
    int n, x;
    cin >> n;
    for (int i = 1; i <= n; i++)
    {
        cin >> x;
        //将等于 x 的值全部装入第 x 桶中
        b[x]++;
    }
    //输出排序结果
    for (int i = 0; i <= M; i++)
    {
        //相同的整数，要重复输出
        for (int j = 1; j <= b[i]; j++)
        {
            cout << i << " ";
        }
    }
    cout << endl;
    return 0;
}
```