

重编码

问题描述

有一篇文章，文章包含 n 种单词，单词的编号从 1 至 n ，第 i 种单词的出现次数为 w_i 。

现在，我们要用一个 2 进制串（即只包含 0 或 1 的串） s_i 来替换第 i 种单词，使其满足如下要求：对于任意的 $1 \leq i, j \leq n, i \neq j$ ，都有 s_i 不是 s_j 的前缀。（这个要求是为了避免二义性）

你的任务是对每个单词选择合适的 s_i ，使得替换后的文章总长度（定义为所有单词**出现次数与替换它的二进制串的长度乘积**的总和）最小。求这个最小长度。

字符串 S_1 （不妨假设长度为 n ）被称为字符串 S_2 的前缀，当且仅当： S_2 的长度不小于 n ，且 S_1 与 S_2 前 n 个字符组成的字符串完全相同。

输入格式

第一行一个整数 n ，表示单词种数。

第 2 行到第 $n + 1$ 行，第 $i + 1$ 行包含一个正整数 w_i ，表示第 i 种单词的出现次数。

输出格式

输出一行一个整数，表示整篇文章重编码后的最短长度。

样例输入

```
4
1
1
2
2
```

样例输出

```
12
```

样例解释

一种最优方案是令 $s_1=000$ ， $s_2=001$ ， $s_3=01$ ， $s_4=1$ 。这样文章总长即为 $1*3+1*3+2*2+1*2=12$ 。

另一种最优方案是令 $s_1=00$ ， $s_2=01$ ， $s_3=10$ ， $s_4=11$ 。这样文章总长也为 12。

数据范围

对于第 1 个测试点，保证 $n = 3$ 。

对于第 2 个测试点，保证 $n = 5$ 。

对于第 3 个测试点，保证 $n = 16$ ，且所有 w_i 都相等。

对于第 4 个测试点，保证 $n = 1000$ 。

对于第 5 个测试点，保证所有 w_i 都相等。

对于所有的 7 个测试点，保证 $2 \leq n \leq 10^6$ ， $w_i \leq 10^{11}$ 。

时间限制：2 sec

空间限制：256 MB

提示

[我们希望越长的串出现次数越少，那么贪心地考虑，让出现次数少的串更长。]

[于是我们先区分出出现次数最少的 2 个串，在它们的开头分别添加 0 和 1。]

[接着，由于它们已经被区分（想一想，为什么？），所以我们可以把它们看作是**一个**单词，且其出现次数为它们的和，然后继续上面的“添数”和“合并”操作。]

[这样，我们不停地“合并单词”，直到只剩 1 个单词，即可结束。]

[可以证明这是最优的。]

[朴素的实现是 $O(n^2)$ 的，可以用二叉堆或 `__std::priority_queue__` 将其优化至 $O(n \log n)$ 。]

另外，为了帮助大家完成题目，我们提供了只包含了输入输出功能的程序模板，也提供了含有算法的大部分实现细节的程序。

你可以根据自己的实际情况，在这些程序的基础上进行作答，或不参考这些程序，这将与你的得分无关。

这些程序可以从【[这里](#) (attachment/f7bd/f7bd45026f05b353c848ce1a670b97747966c177.zip)】下载。

Source

改编自：【NOI2015】荷马史诗 (<http://uoj.ac/problem/130> (<http://uoj.ac/problem/130>))

UI powered by Twitter Bootstrap (<http://getbootstrap.com/>).

Tsinghua Online Judge is designed and coded by Li Ruizhe.

For all suggestions and bug reports, contact [oj\[at\]liruizhe\[dot\]org](mailto:oj[at]liruizhe[dot]org).