课后作业 1

课后作业

1. 平均数原理与抽屉原理

1. 设$a_1, a_2, \ldots, a_N$的算术平均数为x。证明：存在至少一个a_i使得$a_i \le x$，存在至少一个a_j使得$a_j \ge x$。
2. 将N个球放进M个抽屉里。证明：至少有一个抽屉中放有不少于$[N/M]$个球。

2. 区间质数计数

Eratosthenes 筛法利用如下过程求出$1 \ldots N$中的所有质数：遍历$2 \ldots N$中的所有整数，对于每一个整数执行如下操作：若该整数尚未被标记，则该整数为质数，并标记除该整数外所有不超过N的该整数的倍数。可以证明该算法的时间复杂度为$O(N \ln \ln N)$。

利用上述算法求解下列问题。

给定整数Q（$1 \le Q \le 10^5$）以及Q个询问，每个询问包含整数L, R（$1 \le L \le R \le 10^7$）。求$L \ldots R$中质数的数量。

输入样例：

```
3
4 4
3 9
1 100
```

输出样例：

```
0
3
25
```

3. 高精度计算 IV

编写一个程序，输入两个非负整数a, b（$0 \le a, b < 10^{100}$），输出$a \times b$。

输入样例：

```
666 233
```

输出样例：

```
155178
```

4. 取石子

有N（$1 \le N \le 10^5$）堆石子，每堆中各有a_i（$1 \le a_i \le 10^9$）个石子。每次操作可以选择一堆非空的石子，取走其中1个。输入N, M ($1 \le M \le 2 \cdot 10^9$)，以及各堆石子的数量，求至少需要进行的操作次数，使得任意相邻两堆石子数量之和不超过M。

输入样例：

```
5 1
1 2 1 2 1
```

输出样例：

```
4
```

5. 子序列

给定两个字符串S, T（$1 \le |S|, |T| \le 10^5$）。判断T是否是S的一个子序列。一个字符串的子序列指的是删除其中一些字符而保持其他字符相对位置所得到的字符串。

输入样例 1：

```
abc
ac
```

输出样例 1：

```
Yes
```

输入样例 2：

```
abc
cb
```

输出样例 2：

```
No
```

6. (USACO 2012 November – Bronze 1) Find the Cow!

[Brian Dean, 2012]

Bessie the cow has escaped and is hiding on a ridge covered with tall grass. Farmer John, in an attempt to recapture Bessie, has decided to crawl through the grass on his hands and knees so he can approach undetected. Unfortunately, he is having trouble spotting Bessie from this vantage point. The grass in front of Farmer John looks like a string of N (1 <= N <= 50,000) parentheses; for example:

```
)(((())(())
```

Farmer John knows that Bessie's hind legs look just like an adjacent pair of left parentheses ((, and that her front legs look exactly like a pair of adjacent right parentheses)). Bessie's location can therefore be described by a pair of indices x < y such that ((is found at position x, and)) is found at position y. Please compute the number of different such possible locations at which Bessie might be standing.

PROBLEM NAME: cowfind

INPUT FORMAT:

* Line 1: A string of parentheses of length N (1 <= N <= 50,000).

SAMPLE INPUT (file cowfind.in):

```
)(((())(())
```

OUTPUT FORMAT:

* Line 1: The number of possible positions at which Bessie can be standing -- that is, the number of distinct pairs of indices x < y at which there is the pattern ((at index x and the pattern)) at index y.

SAMPLE OUTPUT (file cowfind.out):

```
4
```

OUTPUT DETAILS:

There are 4 possible locations for Bessie, indicated below:

```
1. )(((())(())
    ^^   ^^
2. )(((())(())
     ^^  ^^
3. )(((())(())
     ^^    ^^
4. )(((())(())
    ^^     ^^
```

7. (USACO 2015 December – Silver 2) High Card Wins

Bessie the cow is a huge fan of card games, which is quite surprising, given her lack of opposable thumbs. Unfortunately, none of the other cows in the herd are good opponents. They are so bad, in fact, that they always play in a completely predictable fashion! Nonetheless, it can still be a challenge for Bessie to figure out how to win.

Bessie and her friend Elsie are currently playing a simple card game where they take a deck of $2N$ cards, conveniently numbered $1 \ldots 2N$, and divide them into N cards for Bessie and N cards for Elsie. The two then play N rounds, where in each round Bessie and Elsie both play a single card, and the player with the highest card earns a point.

Given that Bessie can predict the order in which Elsie will play her cards, please determine the maximum number of points Bessie can win.

INPUT FORMAT (file highcard.in):

The first line of input contains the value of N ($1 \leq N \leq 50{,}000$).

The next N lines contain the cards that Elsie will play in each of the successive rounds of the game. Note that it is easy to determine Bessie's cards from this information.

OUTPUT FORMAT (file highcard.out):

Output a single line giving the maximum number of points Bessie can score.

SAMPLE INPUT:

```
3
1
6
4
```

SAMPLE OUTPUT:

```
2
```

Here, Bessie must have cards 2, 3, and 5 in her hand, and she can use these to win at most 2 points by saving the 5 until the end to beat Elsie's 4.

Problem credits: Austin Bannister and Brian Dean