

Sequence: 数列

input ファイル “sequence.in”

output 標準出力

ソースファイル sequence.c/sequence.cpp

時間制限 1 秒 / データ

数列 $\{A_n\}$ は, A_1 から A_m が入力によって与えられ, $i \geq m + 1$ のとき, $A_i = A_{i-1} + A_{i-m}$ をみたす. この数列の第 p 項から第 q 項 $\{A_p, \dots, A_q\}$ に含まれる奇数の個数を求めよ.

Input. 入力ファイルは sequence.in である. 1 行目には m が書かれている. 2 行目には p が書かれている. 3 行目には q が書かれている. $i + 3$ 行目 ($1 \leq i \leq m$) には A_i に対応する値が書かれている.

4 行目から $3 + m$ 行目に書かれている値は非負整数であり, $2\,000\,000\,000 = 2 \times 10^9$ よりも小さい. また, $2 \leq m \leq 24, 1 \leq p < q \leq 2^{60}$ である. p, q の値が 32bit に収まらないことに注意せよ.

Output. 出力は標準出力に行うこと. 指定された範囲に含まれる奇数の数を表す整数を出力せよ.

採点基準 採点用データのうち, 配点の 30 % 分は $q \leq 1\,000\,000 = 10^6$ を満たす.

例 1

sequence.in	標準出力
4	3
2	
8	
1	
2	
3	
4	

例 2

sequence.in	標準出力
3	0
1	
100	
0	
0	
0	

Stamps: 判子 (はんこ)

input ファイル “stamps.in”

output 標準出力

ソースファイル stamps.c/stamps.cpp

時間制限 1 秒 / データ

IOI 国にある老舗判子メーカーの IOI 堂は今年で創業 101 年となる。これを記念し IOI 堂ではオーダーメイドで長いメッセージを入れた判子を作るサービスを始めることにした。というのも昨年、IOI 堂は手動で判子の特定の部分に文字を挿入したり、抜き出したり、入れ替えたりすることができる判子を開発し、より長い文字列の判子を作ることを可能としたのである。

今回作る判子は、アルファベットの I と O の 2 文字のみから構成される IOI 語を対象にする。また、既存の機械で予め適当な 1 文字以上の長さの判子を作り、その文字列を適宜編集して注文のメッセージを作ることにする。ただし、古くからの仕様により、この機械で作ることができるのは、I で始まり I で終わる、どの連続した 2 文字も同一ではない文字列 (例えば I や IOI や IOIO…OIOI) の判子である。

現在、IOI 堂では人手が足りていないので作業時間をできる限り短くしたい。機械作業は時間がかからないが、その後の 3 種類の編集作業、すなわち 1 つの文字を挿入する作業、1 つの文字を削除する作業、1 つの文字を入れ替える作業は、各々 1 回あたり 1 秒の時間が必要である。例えば IOIOIOI という文字列の判子を機械で作った後に、3 文字目を O に入れ替え、5 文字目と 6 文字目の間に O を 1 文字挿入し IOOOIOOI という文字列の判子を作ると 2 秒の時間が必要となる。

そこで、注文のメッセージが与えられた時、最短の作業時間と、最短時間で作業した時に予め機械で作らなければならない判子の長さの最小値を求めるプログラムを書いてほしい。

Input. 入力ファイル stamps.in の 1 行目には、1 つの注文されたメッセージの長さを表す整数 $N(1 \leq N \leq 1000000)$ が書かれている。2 行目には注文されたメッセージを表す N 文字の I または O からなる文字列 S が書かれている。

Output. 出力は標準出力に行うこと。出力の 1 行目には最短の作業時間を書け。2 行目には最短時間で作業した時に予め機械で作らなければならない判子の長さの最小値を書け。

採点基準 採点用データのうち、配点の 40% 分については $N \leq 5000$ を満たす。

例 1

stamps.in	標準出力
8	2
IOOOIOOI	7

例 2

stamps.in	標準出力
5	0
IOIOI	5

例 3

stamps.in	標準出力
5	2
IIIII	5

Pyramid: 貫きピラミッド

input ファイル “pyramid.in”

output 標準出力

ソースファイル pyramid.c/pyramid.cpp

時間制限 5 秒 / データ

古代 JOI 王国では、国王の墓として砂漠にピラミッドを作る風習があった。この国では、国王が死ぬと、占いによって決められる「ある場所」に、占いによって決められる「ある高さ」のピラミッドを作ることになっている。

JOI 王国の砂漠は東西幅が W 、南北幅が H の長方形の形をしている。砂漠は 1×1 の正方形の区画に分かれており、各区画は $0 \leq x < W$, $0 \leq y < H$ を満たす整数 x, y によって (x, y) と表現される。区画 $(0, 0)$ は北西隅の区画であり、区画 (x, y) は、区画 $(0, 0)$ から東に x 、南に y 進んだ地点に位置する区画である。

さて、ピラミッドは次のようにして作られる。まず、占いによってピラミッドの中心の区画 (X, Y) と高さ h が決められる。それに従って、次の規則で砂漠内の各区画に石を積むことによってピラミッドを建設する：

区画 (X, Y) を中心とする高さ h のピラミッドを作る場合、砂漠内の区画 (x, y) には $\max\{0, h - \max\{|X - x|, |Y - y|\}\}$ 個の石を積む。砂漠の外には石は一切積まない。

たとえば、砂漠の大きさが $W = 7, H = 6$ であり、区画 $(2, 1)$ を中心とする高さ 3 のピラミッドを作った場合、各区画に積まれている石の数は以下のようなになる。

1	2	2	2	1	0	0
1	2	3	2	1	0	0
1	2	2	2	1	0	0
1	1	1	1	1	0	0
0	0	0	0	0	0	0
0	0	0	0	0	0	0

しかし、JOI 王国はそれほど広くはないため、ピラミッドは過去のピラミッドと「重なって」作られることもある。すなわち、新たにピラミッドを作ろうとしてある区画に n 個の石を積むことになった場合に、その区画に既に n 個以上の石が積まれていた場合は、その区画には何も積まない。一方、その区画にまだ n 個未満しか石が積まれていなかった場合は、その区画に積まれている石の数を n 個まで増やす。

そのため、多数のピラミッドが作られた末の見た目は複雑となる。たとえば上図の状態、さらに区画(4,3)を中心とする高さ4のピラミッドを作ると、各区画に積まれている石の数は以下のようなになる。

1	2	2	2	1	1	1
1	2	3	2	2	2	2
1	2	2	3	3	3	2
1	1	2	3	4	3	2
0	1	2	3	3	3	2
0	1	2	2	2	2	2

さて、考古学者であるあなたは、一体どれほどの数の石がピラミッドの建設に用いられているかを知りたくなった。

全てのピラミッドの中心位置の区画と高さが与えられた時、それらを建設するのに必要な石の数を求めるプログラムを書け。

Input. 入力ファイル `pyramid.in` の1行目は3つの整数 W, H, N ($1 \leq W, H \leq 3000$, $1 \leq N \leq 10\,000$) が書かれている。 W, H はそれぞれ砂漠の横幅、縦幅を表す。また、 N はピラミッドの個数を表す。

2行目以降の $i+1$ 行目 ($1 \leq i \leq N$) には、3つの整数 x_i, y_i, h_i ($0 \leq x_i < W$, $0 \leq y_i < H$, $1 \leq h_i \leq 3000$) が書かれている。これらは、 i 番目のピラミッドの中心位置の区画が (x_i, y_i) であり、高さが h_i であることを表す。

Output. 出力は、標準出力に行うこと。全てのピラミッドを建設するのに必要な石の数を表す1つの整数を出力せよ。

採点基準 採点用データのうち、配点の10%分については、 $W, H \leq 1000$, $N \leq 5$ を満たす。また、配点の25%分については、 $W, H \leq 500$ かつ、全てのピラミッドの高さが100以下である。

例1

<code>pyramid.in</code>	標準出力
7 6 2	81
2 1 3	
4 3 4	

例2

<code>pyramid.in</code>	標準出力
3000 3000 1	17999999500
1500 1500 3000	