

Day1模拟赛

一、题目概览

中文题目名称	A	B	C	D
英文题目名称	a	b	c	d
可执行文件名	a	b	c	d
输入文件名	a.in	b.in	c.in	d.in
输出文件名	a.out	b.out	c.out	d.out
时间限制	1s	1s	1s	2s
空间限制	512MB	512MB	512MB	512MB
测试点数目	10	子任务	子任务	子任务
测试点分值	10	子任务	子任务	子任务
题目类型	传统	传统	传统	传统
比较方式	全文比较	全文比较	全文比较	全文比较
是否有部分分	是	是	是	是

二、注意事项：

- 1.文件名（程序名和输入输出文件名）必须使用小写。
- 2.C/C++中函数main()的返回值类型必须是int，程序正常结束时的返回值必须是0。
- 3.评测环境为Windows，使用lemon进行评测。
- 4.选手不得使用SSH等命令。
- 5.选手不得使用内嵌汇编，#pragma等指令。

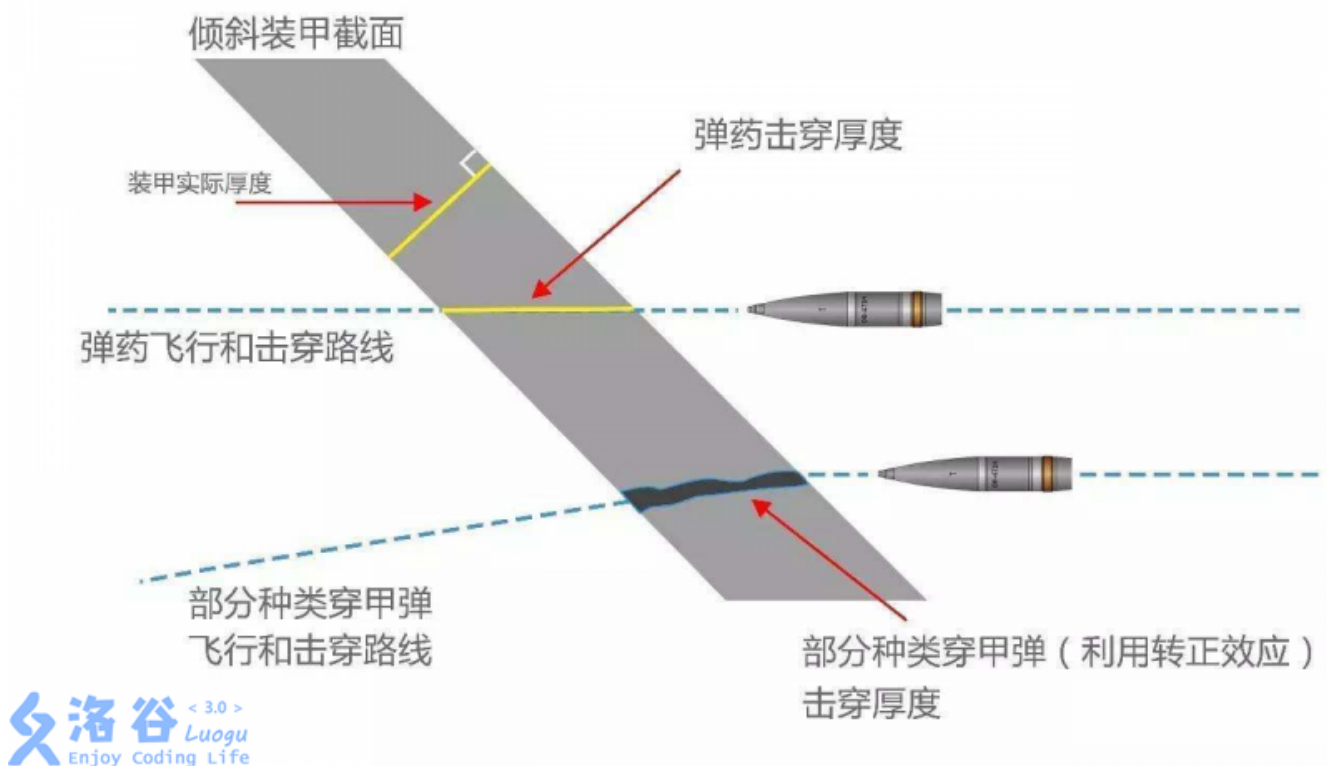
A

题目描述

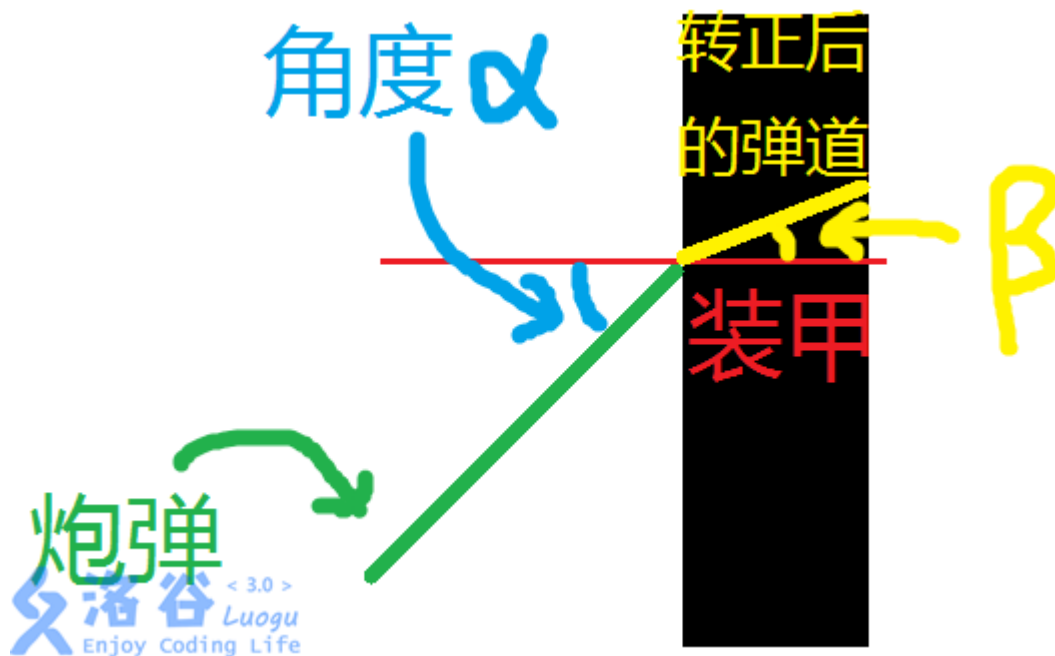
我们来模拟一款垃圾坦克游戏的击穿判定吧。

给定炮弹类型，穿透力，炮弹口径，敌方装甲的垂直厚度，和炮弹与装甲的角度，判断能不能击穿敌人的装甲！

1. 坦克炮弹有6种：AP，APCR，APFSDS，HEAT，HE，HESH。
2. 坦克炮弹打在装甲上是有角度的，角度和是否击穿的关系是很大的。
3. 坦克炮弹是有转正效应的，也就是说：



AP 弹的转正是5度，APCR 和 APFSDS 弹的转正是2度，HEAT，HE，HESH 没有转正 当一发炮弹打中敌人装甲的时候，假设角度是 α ，转正度数是 ω ，那么实际上我们计算的时候可以认为炮弹和敌人装甲的实际角度是 $\beta = \max(0, \alpha - \omega)$ 。



4.当炮弹口径比垂直装甲厚度的两倍还大但不比垂直装甲厚度的三倍大的时候，炮弹的转正度数 ω 会变成 2ω 。

当炮弹口径比垂直装甲厚度的三倍还大的时候，炮弹的转正度数 ω 会变成 3ω 。

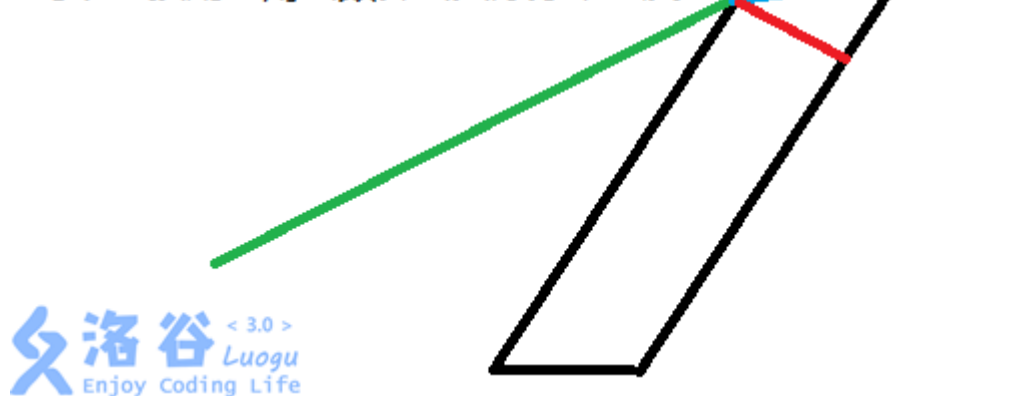
5.计算穿甲能力的时候，我们直接按照算出来的实际角度（考虑转正），算出来炮弹实际要击穿多厚的装甲，这个厚度叫做“等效装甲”，然后判断是否击穿即可。

绿色：炮弹弹道

蓝色：炮弹转正后实际穿甲厚度

红色：装甲垂直厚度

可以直接用三角函数算出实际穿甲厚度



6.当AP，APCR，APFSDS在实际角度（计算转正后）大于等于70度的时候会发生跳弹，HEAT在大于等于85度的时候会发生跳弹，跳弹则认为无法击穿，HE，HESH永不跳弹。



如图是发生了一次跳弹。

7.炮弹穿透力有上下25%的浮动，也就是说穿透力最高是给定的穿透力 $\times 125\%$ ，最低是给定的穿透力 $\times 75\%$ ，这个是一个随机数，本题只需判断：

1. 一定可以击穿
2. 概率可以击穿
3. 一定无法击穿

只要炮弹穿透力经过浮动后大于或等于等效装甲的厚度，那就认为发生了击穿。

8.为了简化题目，不考虑间隙装甲，反应装甲，人肉装甲以及负转正。

输入格式

第一行两个字符串，字符串不包含空格和 tab，表示两个坦克的名字。

第二行，一个字符串表示炮弹类型，然后四个实数表示：炮弹的穿透力，炮弹和装甲的角度（角度是给的入射角，即弹道和法线的夹角），炮弹的口径，垂直装甲的厚度。

输出格式

输出一行一个字符串表示：

如果一定可以击穿，"Penetration!"

如果可能可以击穿，"Server is important~"

如果一定不能击穿，"We didn't penetrate their armor!"

输出不包含`"``符号。

样例输入1

```
E50Ausf.M E50Ausf.M
APCR 270 16 105 185
```

样例输出1

```
Penetration!
```

样例输入2

```
JagdpanzerE100 obj.430u
AP 299 47.1 170 195
```

样例输出2

```
Server is important~
```

样例输入3

```
G.W.Tiger obj430
HE 53 66.1 210 16
```

样例输出3

```
Penetration!
```

样例输入4

```
strv103b BC25t
APCR 350 72.1 105 60
```

样例输出4

```
We didn't penetrate their armor!
```

数据范围

所有读入的所有数在 $[1, 420]$ 内。

数据保证浮点误差在 0.1 内不影响答案，即最多一位小数。

样例解释

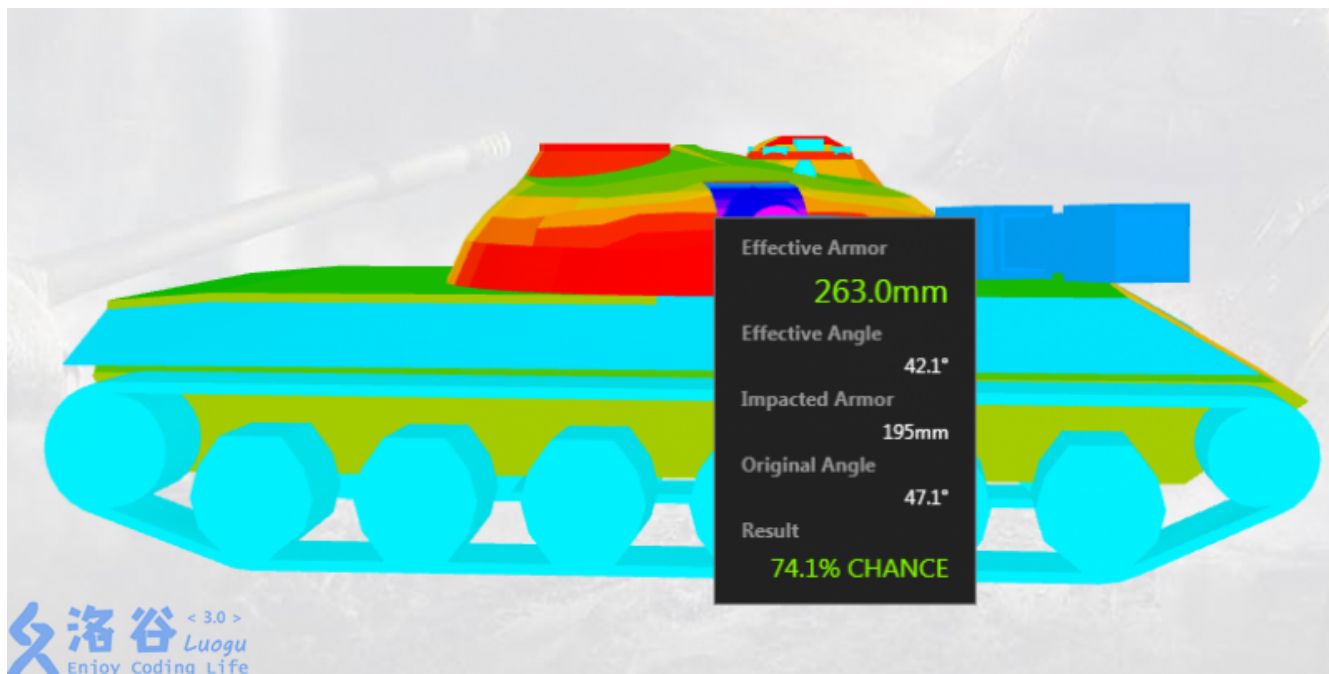
样例解释1:



可以看到原始角度是16度，炮弹是APCR，有2度转正，转正之后实际角度变成了 $16-2=14$ 度，垂直装甲厚度185mm，等效装甲厚度190.7mm，炮弹最低穿深为 $270 \times 0.75 = 202.5$ mm，所以一定可以击穿

样例解释2:

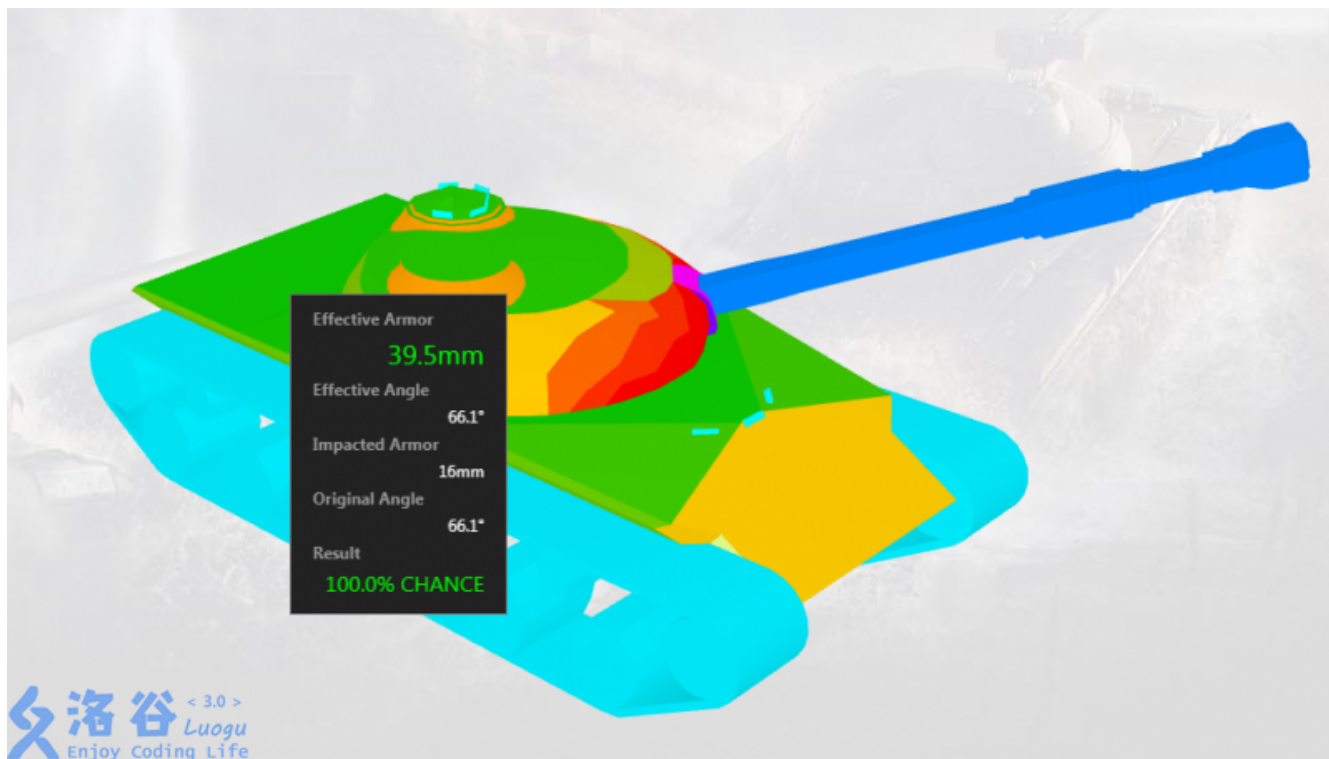




可以看到原始角度是47.1度，炮弹是AP，有5度转正，转正之后实际角度变成了 $47.1 - 5 = 42.1$ 度，垂直装甲厚度195mm，等效装甲厚度263mm，炮弹最低穿深为 $299 \times 0.75 = 224.25$ mm，最高穿深为 $299 \times 1.25 = 373.75$ mm，所以可能击穿

样例解释3:

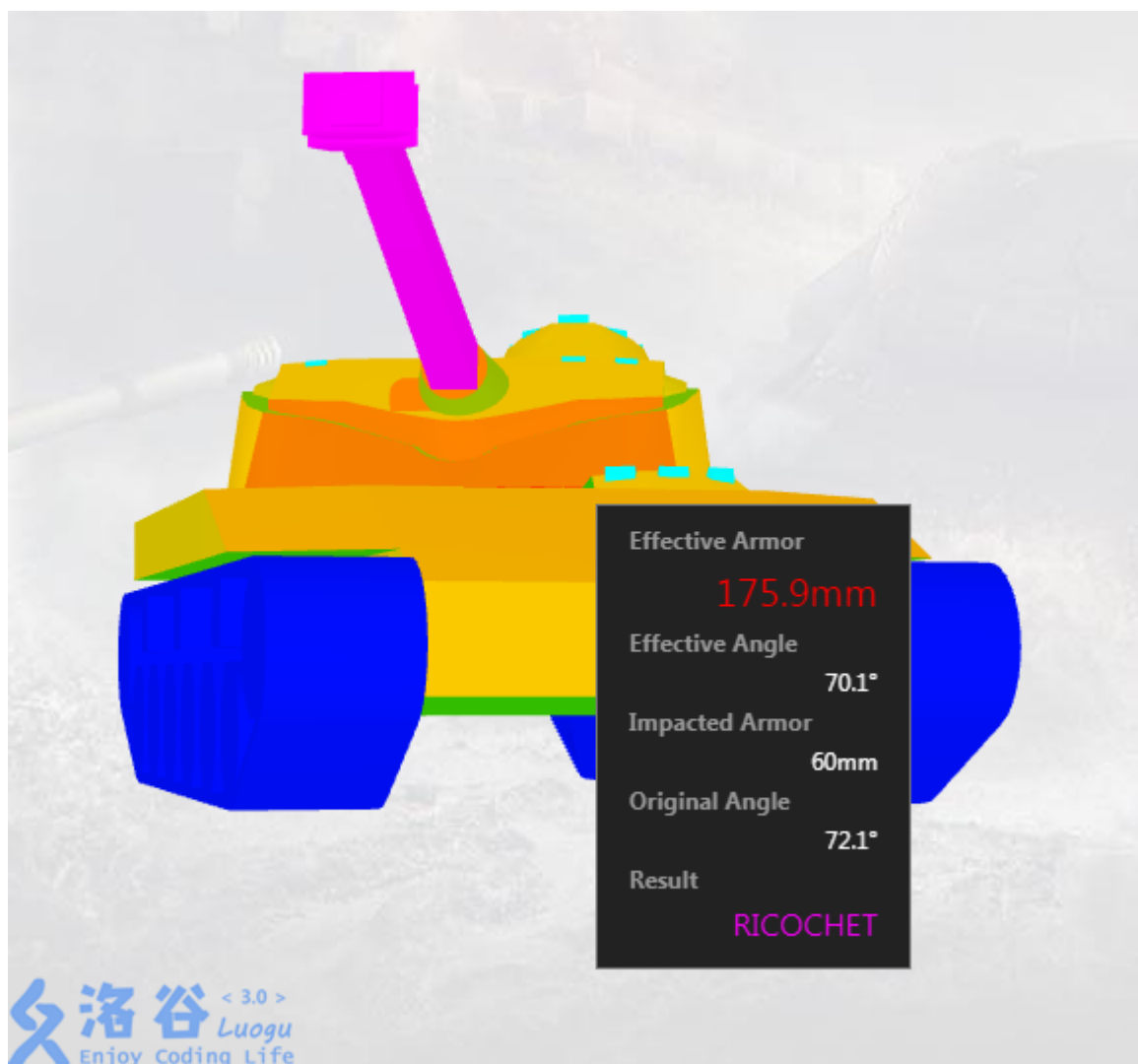




可以看到原始角度是66.1度，炮弹是HE，没有转正，垂直装甲厚度16mm，等效装甲厚度39.5mm，炮弹最低穿深为 $53 \times 0.75 = 39.75\text{mm}$ ，所以一定可以击穿

样例解释4：





可以看到原始角度是72.1度，炮弹是APCR，有2度转正，转正后实际角度为 $72.1 - 2 = 70.1$ 度，垂直装甲厚度60mm，等效装甲厚度175.9mm，但是由于转正后超过了70度，会发生跳弹，所以无法击穿

B

题目描述

给定一个长度为 n 的序列 a ，一共有 m 个操作。

每次操作的内容为：给定 x, y ，序列中所有 x 会变成 y 。

同时我们有一份代码：

```
int ans = 2147483647;
for (int i = 1; i <= n; i++) {
    for (int j = i + 1; j <= n; j++) {
        if (a[i] == a[j])
            ans = std::min(ans, j - i);
    }
}
std::cout << ans << std::endl;
```

请在每次修改后输出代码运行的结果。

输入格式

第一行两个数 n, m 。

第二行 n 个数，依次表示序列 a 中的每个元素。

然后 m 行每行两个数 x 和 y ，表示序列中所有 x 会变成 y 。

输出格式

输出 m 行，即对于每次修改操作，输出一行一个数表示进行了修改后的答案。

样例输入

```
5 5
2 7 6 3 8
2 3
6 1
8 1
1 2
2 3
```

样例输出

```
3
3
2
2
1
```

数据范围

本题采用子任务评测。

对于 30% 的数据，满足 $1 \leq n, m \leq 1000$ 。

对于另外 30% 的数据，满足 $n \leq 1000$ 。

对于 100% 的数据，满足 $1 \leq n, m \leq 10^5$, $1 \leq a_i \leq n$ 。

C

题目描述

输入一个 **01** 串 $S1$ 。你需要输出一个最短的 **01** 串 $S2$ ，使得 $S2$ 在 $S1$ 中从未出现过。

如果有多个可行的解，你需要输出字典序最小的那一个。

输入格式

一行一个字符串，表示 $S1$ 。

输出格式

一行一个字符串，表示 $S2$ 。

样例输入

```
10100011
```

样例输出

```
110
```

数据范围

本题采用子任务评测。

对于 10% 的数据，满足输入数据为样例。

对于 30% 的数据，满足 $|S1| \leq 1000$ 。

对于 50% 的数据，满足 $|S1| \leq 1000000$ 。

对于另外 20% 的数据，时限为 2s。

对于 100% 的数据，满足 $1 \leq |S1| \leq 16777216$ 。

D

题目描述

给定一个序列 A ，定义一个区间 $[l, r]$ 的最大子段和为 $[l, r]$ 的一段连续子区间的和，满足其为 $[l, r]$ 所有连续子区间和中最大的一个。

求出所有 $1 \leq l \leq r \leq n$ 的区间 $[l, r]$ 的最大子段和的和，答案对 2^{64} 取模。

具体来说，假设答案为 x ，你需要输出最小的非负整数 y ，满足 x 和 y 对 2^{64} 同余。

输入格式

第一行一个数 n 。

第二行 n 个数，第 i 个表示 A_i 。

输出格式

一行一个数表示答案。

样例输入1

```
5
1 -1 1 -1 1
```

样例输出1

```
11
```

样例输入2

```
5
1 -2 3 -4 5
```

样例输出2

```
39
```

数据范围

本题采用子任务评测。

对于 30% 的数据，满足 $1 \leq n \leq 1000$ 。

对于另外 20% 的数据，满足 $0 \leq A_i$ 。

对于 100% 的数据，满足 $1 \leq n \leq 10^5$ ， $-10^9 \leq A_i \leq 10^9$ 。