Problem A
AibohphobiA

Time limit: 3 seconds
Memory limit: 1024 megabytes

Problem Description

You are given a rectangular grid of M rows and N columns. The rows and columns are indexed from 0 to $M-1$ and from 0 to $N-1$ respectively. In each grid cell (i, j), there is a lowercase letter character $A[i, j]$. This grid represents a maze, and the goal to solve the maze is to find a walk going from $(0, 0)$ to $(M-1, N-1)$. The walk consists of several steps. In each step you can choose one of the four directions (going from a grid cell to a neighboring cell that shares an edge.) Notice that it is okay to revisit a cell multiple times during the walk, including the starting cell $(0, 0)$ and the ending cell $(M-1, N-1)$. If you record all characters along the walk, you'll get a string that represents this walk.

Truckski is not a fan of palindromes, so he would like to find a walk that does not contain *any palindromic substrings* of length at least two, which he called a *good walk*. A string $s_1 s_2 \cdots s_k$ is called a palindrome, if it reads the same after reversing the string, i.e., $s_1 s_2 \cdots s_k = s_k s_{k-1} \cdots s_1$. A *substring* of a string can be obtained by removing a (possibly empty) prefix and a (possibly empty) suffix.

Now, there are Q interesting locations $\{(r_i, c_i)\}_{i=1}^{Q}$ that Truckski wishes to visit. For each location (r_i, c_i), can you help Truckski to find the length of the longest good walk that visits the location grid cell (r_i, c_i) at least once? If there are arbitrarily long good walks please output `-1`. If there does not exist any good walk, please output `-2`.

Input Format

The first line contains an integer T, indicating the number of test cases. For each test case, there are two integers M and N in the first line. In each of the following M lines there is a string of length N, the c-th character in the r-th line is the character $A[r, c]$. The next line contains an integer Q. In each of the following Q lines there are two integers r_i and c_i indicating the location of interest.

Output Format

For each interesting location, output the length of the longest good walk that visits this location at least once, or `-1` if the good walk can be arbitrarily long, or `-2` if there does not exist such a good walk.

Technical Specification

- $T \leq 20$
- $2 \leq M \leq 100$
- $2 \leq N \leq 100$

- $1 \leq Q \leq 100$
- For all i such that $1 \leq i \leq Q$, $0 \leq r_i < M$ and $0 \leq c_i < N$.
- For each grid cell (r, c), $A[r, c] \in \{\texttt{a}, \texttt{b}, \ldots, \texttt{z}\}$ is a lowercase letter.

Sample Input 1

```
3
3 5
abbba
bccab
cabcc
2
0 1
1 0
3 4
aaba
bbaa
abab
1
1 1
4 4
abca
cxxb
bxxc
acba
1
0 1
```

Sample Output 1

```
9
9
-2
-1
```

Note

This problem is not the easiest problem in this contest.

Problem B
Balanced Seesaw Array

Time limit: 3 seconds
Memory limit: 1024 megabytes

Problem Description

Bob likes to play seesaw. He thinks that it would be really funny if the seesaw is in a balanced state. It means that the seesaw is not tilted to the left and right. After playing the seesaw, Bob thinks about a problem related to the balanced seesaw.

Let $A = [a_1, a_2, \ldots, a_m]$ denote an array of length m. Bob thinks that $[a_1, a_2, \ldots, a_m]$ is a *balanced seesaw array* if there exists an integer k between 1 to m such that $\sum_{i=1}^{m}(i-k)a_i = 0$.

Bob gets an array $A = [a_1, a_2, \ldots, a_n]$ as his birthday gift, and he is curious about whether some non-empty subarray is a *balanced seesaw array*. More formally, he is interested in whether $[a_\ell, a_{\ell+1}, \ldots, a_r]$ is a *balanced seesaw array* for some specified pair (ℓ, r) where $1 \leq \ell \leq r \leq n$. Bob also finds that the elements in its array will change by time, it will have the following two types of changes.

1. $a_\ell, a_{\ell+1}, \ldots, a_r$ are increased by x.
2. $a_\ell, a_{\ell+1}, \ldots, a_r$ are changed to x.

For convenience, Bob will give you the array $A = [a_1, a_2, \ldots, a_n]$ first. Then, there are q operations. Each operation will be one of the following three types.

- $1\ \ell\ r\ x$: means that $a_\ell, a_{\ell+1}, \ldots, a_r$ are increased by x.
- $2\ \ell\ r\ x$: means that $a_\ell, a_{\ell+1}, \ldots, a_r$ are changed to x.
- $3\ \ell\ r$: means that Bob is curious about whether the subarray $[a_\ell, a_{\ell+1}, \ldots, a_r]$ is a *balanced seesaw array*. You should output "`Yes`" or "`No`" for each operation type 3.

Input Format

The first line of input contains two integers n and q. n is the length of the array, and q is the number of operations. The second line contains n integers a_i to define the array. Each of the following q lines is an operation described in the problem statement.

Output Format

Please output "`Yes`" or "`No`" to indicate whether $[a_\ell, a_{\ell+1}, \ldots, a_r]$ is a *balanced seesaw array* for each type 3 operation.

Technical Specification

- $1 \leq n \leq 100000$
- $1 \leq q \leq 1200000$
- $-1000 \leq a_i \leq 1000$
- $-10000 \leq x \leq 10000$

- For $1 \le i \le n$, you may assume that $|a_i| \le 1.5 \times 10^9$ after any operation.
- $1 \le \ell \le r \le n$

Sample Input 1

```
3 6
1 2 3
3 1 1
3 1 3
1 1 1 2
3 1 3
2 2 2 0
3 2 3
```

Sample Output 1

```
Yes
No
Yes
Yes
```

Problem C
Correct

Time limit: 3 seconds

Memory limit: 1024 megabytes

Problem Description

Your best friend, Charlie, participated Taiwan Online Programming Contest (TOPC), which is a preliminary contest of the International Collegiate Programming Contest (ICPC).

According to the rules, teams are ranked according to the most problems solved. Teams who solve the same number of problems are ranked by the least total time. The total time is the sum of the time consumed for each solved problem. The time consumed for a solved problem is the time elapsed from the beginning of the contest to the submission of the correct answer plus 20 penalty minutes for every rejected code for that problem. There is no time consumed for a problem that is not solved.

Charlie's team only solved one problem, and the correct answer was submitted at HH:MM AM. Fortunately, they did not submit any rejected code to the judge system. Please write a program to compute the time consumed for the problem solved by Charlie's team. You may assume that TOPC started at 9:00 AM.

Input Format

The input contains two space-separated integers HH and MM.

Output Format

Output the time consumed for the only problem solved by Charlie's team.

Technical Specification

- $9 \leq HH \leq 11$.
- $0 \leq MM \leq 59$.

Sample Input 1

```
9 0
```

Sample Output 1

```
0
```

Sample Input 2

```
11 59
```

Sample Output 2

```
179
```

Almost blank page

Problem D
Distance and Tree

Time limit: 3 seconds

Memory limit: 1024 megabytes

Problem Description

Graph problems are popular in competitive programming, and problems related to distanceis and trees appear frequently. Let us start with some definitions.

A *set* is a collection of distinct elements. An *undirected simple graph* G is a pair (V, E), where V is a set and E is a set of unordered pairs of V's elements. For a graph $G = (V, E)$, we call V as G's vertex set and E as G's edge set. Elements in V are vertices, and elements in E are edges.

Let u and v be vertices in V. A *path* from u to v of length k is a sequence of edges $e_1, e_2, \ldots, e_k \in E$ such that there exists a sequence of distinct vertices, $v_1, \ldots, v_{k+1}$, satisfying the following conditions.

- $u = v_1$.
- $v = v_{k+1}$.
- $e_i = \{v_i, v_{i+1}\}$.

If p is a path from u to v, then u and v are *connected* by p.

We can define distances and trees now. Given two vertices $u, v \in V$, the *distance* $\delta(u, v)$ from u to v is 0 if $u = v$. If there exists a path from u to v, then $\delta(u, v)$ is the minimum number of edges required to form a path from u to v. Otherwise, $\delta(u, v) = \infty$. A *tree* is an undirected graph in which any distinct two vertices u and v are connected by exactly one path.

Danny gives you a sequence of non-negative integers $d_1, d_2, \ldots, d_n$ and asks you to construct a tree $G_T = (V_T, E_T)$ satisfying the following conditions.

- The vertex set $V_T = \{p_1, \ldots, p_n\}$ is a set of points on a two dimensional Euclidean plane. For $1 \le k \le n$, the coordinate of p_k is $(\cos k\theta, \sin k\theta)$ where $\theta = \frac{2\pi}{n}$.
- For any two distinct edges $\{p_a, p_b\}$ and $\{q_a, q_b\}$ in E_T, the line segments $\overline{p_a p_b}$ and $\overline{q_a q_b}$ do not intersect unless those two edges share a common vertex (that is, $\{p_a, p_b\} \cap \{q_a, q_b\} \neq \emptyset$).
- There exists a vertex r such that $\delta(r, p_k) = d_k$ for $1 \le k \le n$. We call r as the root of G_T.

If there exists such tree graph, please output the edge set E_T. Otherwise, output `-1`.

Input Format

The first line contains a positive integer n indicating the number of vertices of the tree to be constructed. The second line contains n non-negative integers $d_1, \ldots, d_n$, the sequence given by Danny.

Output Format

If there does not exist such a tree G_T, output `-1`. Otherwise, output $n-1$ lines to represent the edge set E_T. The i-th line should contain two space-separated integers u_i and v_i. The i-th edge in E_T should be $\{p_{u_i}, p_{v_i}\}$. If there are multiple solutions, you may output any of them.

Technical Specification

- $2 \leq n \leq 100000$
- For $1 \leq k \leq n$, $0 \leq d_k \leq n-1$.

Sample Input 1

```
5
0 1 2 1 3
```

Sample Output 1

```
-1
```

Sample Input 2

```
5
1 1 0 1 1
```

Sample Output 2

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

Problem E
Etched Emerald Orbs

Time limit: 3 seconds
Memory limit: 1024 megabytes

Problem Description

An archaeologist team found a tomb of the ancient tribe and discovered 2^{125} emerald orbs inside the tomb. The ancient tribe etched a numbers on each emerald orb. The archeologists spent two decades realizing that the ancient tribe etched each emerald orb with a unique number. Moreover, the numbers are from 1 to 2^{125} in the ancient language.

Eddy, the only mathematician in the archaeologist team, recently figured out the relation between the number k and the emerald orb numbered k. The weight of the emerald orb numbered k is exactly $\frac{1}{k}$ grams. Since the number on each emerald orb is distinct from the number on any other emerald orb, there are no two emerald orbs having the same weight.

Eddy proposes a hypothesis: the ancient tribe used these emerald orbs to represent weight less than 1 gram. It is trivial that the emerald orb numbered k can represent $\frac{1}{k}$ gram. Then, Eddy tries to represent $\frac{2}{k}$ grams for $3 \le k \le 4 \times 10^{18}$ with two emerald orbs. He successfully finds that the emerald orbs numbered 2 and 6 can represent $\frac{2}{3} = \frac{1}{2} + \frac{1}{6}$ grams. Similarly, the emerald orbs numbered 3 and 15 can represent $\frac{2}{5} = \frac{1}{3} + \frac{1}{15}$ grams.

Can you write a program to help Eddy to check whether two emerald orbs can represent $\frac{2}{k}$ grams for a given integer k? If there are multiple combinations of two emerald orbs representing $\frac{2}{k}$ grams, output the combination minimizing the sum of the numbers etched on them. If there is no such combination, output `-1`.

Input Format

The input contains only one positive integer k.

Output Format

If there is no solution, output `-1`. Otherwise, output two distinct integers x and y separated by a blank where $\frac{2}{k} = \frac{1}{x} + \frac{1}{y}$ and $1 \le x < y \le 2^{125}$. If there are multiple solutions, output the solution minimizing $x + y$.

Technical Specification

- $3 \le k \le 4 \times 10^{18}$.

Sample Input 1

```
5
```

Sample Output 1

```
3 15
```

Sample Input 2

```
7
```

Sample Output 2

```
4 28
```

Problem F
Finalists

Time limit: 3 seconds

Memory limit: 1024 megabytes

Problem Description

Frank is trying his best to win a slot in Asia Pacific to the World Finals. But he could not find the selection rules to estimate his chance. Last night, Frank's coach found the following document.

```
=====================================================================
World Finals team selection rules from Asia Pacific (2022 cycle)

Due to COVID-19 outbreak, this year's Asia Pacific rules will
again be quite different from those of usual years. The plan
for having an Asia Pacific semi-final contest (play-off contest)
has been cancelled and postponed to the next year.

Only the following six regionals will be held this year.

Ho Chi Minh, Vietnam
Jakarta, Indonesia
Manila, Philippines
Seoul, Korea
Taoyuan, Taiwan
Yokohama, Japan

We have set a team participation rule. This rule is a special one
for this year (2022 cycle) only.

Teams from a hosting country (Indonesia, Japan, Korea, Philippines,
Taiwan and Vietnam) can only join the regional held in their own
country. Universities from non-hosting countries should choose a
regional. Note that all teams from a single university have to join
the same regional.

World Finals team selection rules are as follows.

Apply Steps A1 through A4 below in this order.

A1. Define the site score of a regional as follows.
```

```
0.56 * number of universities in regional
+ 0.24 * number of teams in regional
+ 0.14 * number of universities in preliminary contests
+ 0.06 * number of teams in preliminary contests
+ 0.3 * number of foreign teams

Only teams or universities solving at least one problem are counted.

A2. Sort our six regionals according to their site scores.

A3. Let N be the number of World Finals slots given from the ICPC
Headquarters. Until the sum of the number of allocated slots
reaches N, allocate slots to regionals, one by one, in the
descending order of their site scores. When all six regionals
are allocated one slot each, continue from the regional with the
highest site score in the round robin fashion.

A4. For each regional, let M be the number of slots allocated to the
regional through the step A3. Select the M highest ranked
universities. The team with the highest rank in the university
is automatically selected.
=====================================================================
```

Assume that the document is the final selection rules for teams from Asia Pacific. Given `N` and the necessary numbers to calculate the site scores, please write a program to compute the number of slots allocated to Taiwan.

Input Format

There are seven lines in the inputs. The first line contains `N`, the number of World Finals slots given from the ICPC Headquarters. Each of the following six lines consists of one string s and five non-negative integers p_t, p_u, r_t, r_u, f. The string and the numbers are separated by spaces. s is a country hosting a regional contest. p_t is the number of teams solving at least one problem in the preliminary contests hosted by the country s. p_u is the number of universities solving at least one problem in the preliminary contests hosted by the country s. r_t is the number of teams solving at least one problem in the regional contest hosted by the country s. r_u is the number of universities solving at least one problem in the regional contest hosted by the country s. f is the number of foreign teams solving at least one problem in the regional hosted by the country s.

Output Format

Output the number of slots allocated to Taiwan.

Technical Specification

- `N` is at least 6 and at most 50.
- Each line has a distinct s.
- s is among the following countries: Vietnam, Indonesia, Philippines, Korea, Taiwan, and Japan.
- $0 \leq p_u \leq p_t \leq 5000$
- $0 \leq r_u \leq r_t \leq 300$
- $0 \leq f \leq 30$
- You may assume that distinct regional contests have different site scores.
- The size of an input file is no more than 64 kilobytes.

Sample Input 1

```
17
Japan        500   95   40  30   5
Vietnam      400   50  150  40  20
Indonesia    700   25   80  35  20
Taiwan       200   30  100  35   1
Korea        600  100  100  70   0
Philippines   50   10   40  15  15
```

Sample Output 1

```
3
```

Sample Input 2

```
16
Japan        500        95   40  30   5
Taiwan       200  30   100  35   1
Indonesia    700  25    80  35  20
Philippines   50  10     40  15  15
Korea        600  100     100  70   0
Vietnam      400   50      150  40  20
```

Sample Output 2

```
2
```

Note

The document in the problem statement is not official. Please refer to Prof. CJ Hwang's blog `https://icpcasia.wp.txstate.edu/` for the official rules.

Almost blank page

Problem G
Geekflix

Time limit: 3 seconds

Memory limit: 1024 megabytes

Problem Description

George got COVID-19 in this morning. He must stay at home in the next seven days for quarantine. As a geek, George only watches Geekflix, the video streaming service for geeks, for recreation during his quarantine period. Geekflix provides n video streams numbered from 1 to n, and Geekflix also gives some geeky coins to the audiences in their quarantine periods. When George watches stream i for the k-th time in his quarantine period, George gets $\max(a_i - (k-1)b_i, 0)$ geeky coins.

The Geekflix app arranges the video streams on a circle. For $1 < i \leq n$, the previous stream of stream i is stream $i-1$. The previous stream of stream 1 is stream n. For $1 \leq i < n$, the next stream of stream i is stream $i+1$. The next stream of stream n is stream 1. When George opens the Geekflix app on his TV, the Geekflix app points the cursor at stream 1.

The remote controller has three buttons: previous, next, and play. When George presses the previous button, the Geekflix app points the cursor to the previous stream. When George presses the next button, the Geekflix app points the cursor to the next stream. When George presses the play button, the Geekflix app plays the stream pointed by the cursor. The cursor points to the same stream after playing.

George may press the buttons m times during his quarantine period, and he wants to get as many geeky coins as possible. What is the maximum number of geeky coins that George can get during his quarantine period?

Input Format

The first line contains two positive integers n and m. n is the number of video streams, and George may press the buttons m times. The second line contains n positive integers $a_1, \ldots, a_n$, and the third line contains n non-negative integers $b_1, \ldots, b_n$. These $2n$ integers define the number of geeky coins awarded to George when the Geekflix app plays video streams.

Output Format

Output the maximum number of geeky coins that George can get during his quarantine period.

Technical Specification

- $1 \leq n \leq 200$
- $1 \leq m \leq 1000$
- $0 \leq b_i \leq a_i \leq 5000$ for $1 \leq i \leq n$.

Sample Input 1

```
3 10
10 10 10
5 3 1
```

Sample Output 1

```
67
```

Sample Input 2

```
5 10
1 2 3 4 5
0 1 2 3 4
```

Sample Output 2

```
16
```

Problem H
Heximal

Time limit: 5 seconds

Memory limit: 1024 megabytes

Problem Description

The base-6 numeral system is also called the heximal numeral system. We say a string $h_k h_{k-1} \cdots h_1 h_0$ is a heximal number if $h_i \in \{0, 1, 2, 3, 4, 5\}$ for every $i \in \{0, 1, \ldots, k\}$ and $h_k = 0$ implies $k = 0$. The value represented by $h_k h_{k-1} \cdots h_1 h_0$ in the heximal numeral system is $\sum_{i=0}^{k} h_i 6^i$. For example, the value of the heximal number 12345 equals the value of the decimal number $1865 = 1 \times 6^4 + 2 \times 6^3 + 3 \times 6^2 + 4 \times 6 + 5$.

Harry asks you to convert a very large base-10 number N to base-6. Since the conversion result can be very long, it is too hard for Harry to verify the result by himself. So, you just need to tell Harry the length of the conversion result. For example, if $N = 1865$, then you just need to tell Harry the length of the conversion result is 5.

Input Format

The input contains exactly one integer N in decimal.

Output Format

Output the length of the base-6 representation of N.

Technical Specification

$0 \le N < 10^{500000}$.

Sample Input 1

```
1865
```

Sample Output 1

```
5
```

Sample Input 2

```
6
```

Sample Output 2

```
2
```

Sample Input 3

```
5
```

Sample Output 3

```
1
```

Problem I
Invitation

Time limit: 3 seconds

Memory limit: 1024 megabytes

Problem Description

Iris works for the host of the 2022 ICPC Taoyuan Regional Contest. Due to COVID-19, the ICPC regional contests in Taiwan could not invite any leader to the opening ceremony in the past few years. The host of the 2022 ICPC Taoyuan Regional Contest is eager to invite leaders in Taoyuan City to attend the opening ceremony.

There are n leaders numbered from 1 to n in Taoyuan City, and Iris's task is to invite some leaders to attend the ceremony. Leader i is available from time slot ℓ_i to time slot r_i. If Iris wants to invite k leaders $a_1, a_2, ...a_k$, then all of them must have a common available time slot. It means that Iris has to find a time slot x such that $\ell_{a_i} \leq x \leq r_{a_i}$ for $1 \leq i \leq k$.

Iris is curious about the number of combinations of k leaders available at the same time? You need to give the answers for all k between 1 and n. The combinations may be extremely numerous, please output the number of combinations modulo 998244353.

Input Format

The first line of input contains one integer n, the number of leaders. The following n lines indicate the leaders' available time slots. The i-th line of these n lines contains two numbers ℓ_i and r_i. The i-th leader is available at time ℓ_i to r_i.

Output Format

Print n numbers. The k-th number is the number of combinations of k leaders having a common available time slot. Please modulo the answer with 998244353.

Technical Specification

- $1 \leq n \leq 100000$
- $0 \leq \ell_i \leq r_i \leq 1000000000$ for $1 \leq i \leq n$.

Sample Input 1

```
3
1 2
2 3
3 4
```

Sample Output 1

```
3 2 0
```