

The Fire Nation's Vaccine Factory



Source: <https://metro.tempo.co/read/1433143/ombudsman-khawatir-akan-ada-jual-beli-hak-penerima-vaksin-covid-19>

Description

In facing the COVID 19 pandemic in the Fire Nation, the government decided to develop a vaccine to stop the spread of the virus. The result was a few districts were chosen to find the best vaccine formula. Every district has one factory as the representative and the Fire Nation will have a total of P factories with unique names.

There are N types of material available for every factory to produce a vaccine. Every material has a name M_i quality rating K_i . Vaccine factories will attempt to produce vaccines with different formulas every time. A formula has a format of a sequence of *non-negative* numbers: A_1, A_2, \dots, A_n , with A_i being the quantity(formula) for material M_i that is used to produce vaccines. Vaccines produced with said formula will have a quality score equal to $(A_1 \times K_1 + A_2 \times K_2 + \dots + A_n \times K_n) \bmod 1000 + 1$ and a medical trial cost of $A_1 + A_2 + \dots + A_n$. Of course, every factory has limited stock for every material.

EXAMPLE:

There are 5 types of material (e.g. material M_1, M_2, M_3, M_4 , and M_5) with their quality ratings in order being 2, 1, 4, 3, and 5. Factory X has stock for those 5 materials, in order each 5, 7, 1, 3, and 4. If factory X tries to produce vaccine X1 with formula 2, 1, 1, 0, and 0, they will produce a vaccine with a quality rating of $(2 \times 2 + 1 \times 1 + 1 \times 4 + 0 \times 3 + 0 \times 5) \bmod 1000 + 1 = 10$ and medical trial cost of $2 + 1 + 1 + 0 + 0 = 4$. Meanwhile, if factory X tries to produce vaccine X2 with formula 2, 1, 2, 0, and 0, it will fail since the material stock is insufficient.

The Fire Nation will give a time limit of H days to all factories to develop the vaccine before the vaccine trial. For those H days, every factory will do their needed activities. Each day, a factory can do these activities:

- **Restock certain vaccine material** from a trusted distributor.

- **Produce vaccines** with a certain formula. After producing a vaccine, every material's stock will decrease according to the quantity used. The produced vaccine will be stored first in a factory warehouse before being distributed.
- **Distributing an amount of vaccines** from the factory warehouse to the vaccine bank owned by the Fire Nation's Department of Health. Vaccines that are produced earlier will have priority to be distributed first compared to vaccines produced later.

Every **H** days, The Fire Nation's Department of Health will choose a vaccine from the vaccine bank to trial. However, not all vaccines will be trialed because the department has limited funds of **C**. The department will choose select vaccines to trial with the requirement being **the total quality score from all vaccines being trialed are as maximum as possible, with a total trial cost a maximum of C**.

To fulfill data and report needs, the department will store data of vaccines produced by all factories. The information stored consists of **vaccine name, vaccine quality score, vaccine trial cost, and vaccine status (is in vaccine bank or not)**. Other than that, the department will also record the activity of all factories for work assessment needs. The information needed is **the amount of vaccines produced and the amount of vaccines distributed**.

The Health minister of the Fire Nation, Mr. Seda, feels that the work mentioned above will be hard to complete if we only relied on man power. Hence, he asks you as a skilled *programmer* to create a program that can help the department in working. Of course, you being a citizen that loves your country is ready to create this program.

Input

The first line contains an integer **N** that states the amount of material types available in the Fire Nation. **N** rows after it will contain a string **M_i** and integer **K_i** that states the name of the vaccine material and its quality score.

The next lines contain an integer **P** that denotes the amount of factories in the Fire Nation. Then, there will be **P** amount of information (each about factory 1, 2, ..., till P) regarding factory name and initial vaccine material stock in the factory. Each information will consist of **1+N** lines with conditions as follows:

- The first line contains the factory name **S_i**.
- **N** lines following contain a string **M_i** and integer **J_i** that states the name of the vaccine material and quantity of the vaccine material.

The next line is an integer **H**, the amount of days given by the government. Then, there will be **H** amount of information (each for days 1, 2, ..., till H) regarding activity for that day. Each information will contain **1+Q** lines with conditions as follows:

- The first line is integer **Q** that states the amount of activity on that day.
- The next **Q** lines will contain one of these queries:
 - **RESTOCK S M Y**
Factory with name **S** adds material **M** with a quantity of **Y**. The amount of each material being stocked varies according to the factory's demand. **S** and **M** are guaranteed to be a valid factory name and material.
 - **PRODUCE S V A₁ A₂ ... A_n**

Factory with name **S** produces vaccine with name **V** using formula $A_1 A_2 \dots A_n$ and trial cost according to the above description. If the amount of material needed to produce vaccine **V** is insufficient, ignore this query. The name of each vaccine is guaranteed to be **unique** (There are no vaccines with the same name). **S** is guaranteed to be a valid factory name.

○ **DISTRIBUTE S Z**

Factory with name **S** distributes **Z** amount of vaccines from the factory warehouse to the department's vaccine bank. If the amount of vaccines in the warehouse is more than 0 and **Z** exceeds the amount of vaccines in the warehouse, then all vaccines in the warehouse will be distributed. If the amount of vaccines in the warehouse is 0, then ignore this query. **S** is guaranteed to be a valid factory name.

The final line will be one of these queries:

- **CEK_KINERJA_PABRIK**
Produce output of performance information from all factories.
- **CEK_SEMUA_VAKSIN**
Produce output of information all vaccines that has been produced by all factories
- **CEK_TOTAL_KUALITAS_UJICOBAC**
Produce maximum output of the total value of vaccine quality that can be tested in the vaccine bank. C is an integer representing the funds needed to run a vaccine trial

Output

For each day, print the output in the following format "Hari ke-i: **W X**" (without the quotes) starting from day-1. **W** represents the total number of vaccines that have been produced by all factories and **X** states the number of vaccines available in the vaccine bank until day-i.

Next, print an output according to the final query line with conditions as follows:

- For query **CEK_KINERJA_PABRIK**, there will be **P** lines of information that are printed with the format "**S_i E_i F_i**" (without quotations). **S** declares the name of the factory. **E** declares the amount of vaccines that were produced by that factory. **F** declares the amount of vaccines distributed to the bank by that factory. The order of factory information to be printed follows the order the information of the factories were stored when first inputted.
- For query **CEK_SEMUA_VAKSIN**, there will be **W** lines of information to be printed with the format "**V_i L_i B_i D_i**" (without quotations). **V** declares the vaccine name. **L** declares the vaccine's quality score. **B** declares the trial cost. **D** declares the status whether the vaccine is in the bank or not (**D** is TRUE if the vaccine is in the bank otherwise it is FALSE). The order of vaccine information to be printed is ordered according to the vaccine quality score starting from the **largest** (descending). If there are vaccines with the same quality score then they will be ordered according to their trial cost starting from the **smallest** (ascending). If there are vaccines with the same trial cost, then they will be ordered according to their vaccine name ordered lexicographically.
- For query **CEK_TOTAL_KUALITAS_UJICOBAC**, print a number that states the maximum total vaccine quality score that can be trialed in the vaccine bank with a cost of **C**.

Constraints

$$1 \leq N \leq 10$$

$$1 \leq K \leq 20$$

$$1 \leq P \leq 100$$

$$0 \leq J \leq 2.000.000$$

$$1 \leq H \times \sum Q_i \leq 100.000$$

$$0 \leq A \leq 20$$

$$1 \leq Y \leq 2.000.000$$

$$1 \leq Z, C \leq 400$$

$$1 \leq |M|, |S|, |V| \leq 10$$

M, S, and V only consist of characters A-Z and 0-9.

For query **CEK_TOTAL_KUALITAS_UJICOBA**, it's guaranteed that there will be a minimum of 1 vaccine and a maximum of 400 vaccines in the vaccine bank

The amount of material stock in a factory will not exceed 2.000.000

Input Example 1

```
2
APPLE 1
BANANA 2
2
UTARA
APPLE 100
BANANA 100
SELATAN
APPLE 50
BANANA 50
1
5
PRODUCE UTARA UT01 2 1
PRODUCE UTARA UT02 0 2
PRODUCE SELATAN SL01 3 2
DISTRIBUTE SELATAN 1
DISTRIBUTE UTARA 2
<LAST_QUERY>
```

Output Example 1

If <LAST_QUERY> = CEK_KINERJA_PABRIK

```
Hari ke-1: 3 3
UTARA 2 2
SELATAN 1 1
```

If <LAST_QUERY> = CEK_SEMUA_VAKSIN

```
Hari ke-1: 3 3
SL01 8 5 TRUE
UT02 5 2 TRUE
UT01 5 3 TRUE
```

If <LAST_QUERY> = CEK_TOTAL_KUALITAS_UJICOBA 5

Hari ke-1: 3 3
10

Explanation 1

In the first input, there are 2 types of material, namely **APPLE** and **BANANA**. Then, there are 2 factories named **UTARA** and **SELATAN**. The following table is the information about materials and the amount of initial stock at each plant.

Material Name	Quality	Stock in UTARA	Stock in SELATAN
APPLE	1	100 pcs	50 pcs
BANANA	2	100 pcs	50 pcs

Day-1

In Day-1, there are 5 queries which must be done, namely:

- PRODUCE UTARA UT01 2 1
This query produces the UT01 vaccine and is stored in the UTARA factory warehouse. The UT01 vaccine will have a quality value $(2 \times 1 + 1 \times 2) \bmod 1000 + 1 = 5$ and a trial cost of $2 + 1 = 3$. Material stock decreases according to the amount material used.
- PRODUCE UTARA UT02 0 2
This query produces the UT02 vaccine and is stored in the UTARA factory warehouse. The UT02 vaccine will have a quality value $(0 \times 1 + 2 \times 2) \bmod 1000 + 1 = 5$ and a trial cost of $0 + 2 = 2$. Material stock decreases according to the amount material used
- PRODUCE SELATAN SL01 3 2
This query produces the SL01 vaccine and is stored in the SELATAN factory warehouse. The SL01 vaccine will have a quality value $(3 \times 1 + 2 \times 2) \bmod 1000 + 1 = 8$ and a trial cost of $3 + 2 = 5$. Material stock decreases according to the amount of material used.
- DISTRIBUTE SELATAN 1
This query distributes 1 vaccine in the SELATAN factory warehouse. SELATAN factory has 1 vaccine (SL01), so that one vaccine is distributed.
- DISTRIBUTE UTARA 2
This query distributes 2 vaccines in the UTARA factory warehouse. UTARA factory has 2 vaccines (UT01 and UT02), so those vaccines are distributed.

<LAST_QUERY>

- CEK_KINERJA_PABRIK
The following is a breakdown of the performance for each plant.
 - **UTARA factory**
 - Produces **2 vaccines** (UT01 and UT02)
 - Distributes **2 vaccines** (UT01 and UT02) to the vaccine bank
 - **SELATAN factory**
 - Produces **1 vaccine** (SL01)
 - Distribute **1 vaccine** (SL01) to the vaccine bank
- CEK_SEMUA_VAKSIN

The order of vaccines according to quality, trial cost, and name is SL01, UT02, and UT01. UT02 and UT01 have the same quality but UT02 has less trial cost than UT01.

- CEK_TOTAL_KUALITAS_UJICOBA 5

Currently, there are 3 vaccines in the vaccine bank. With a maximum fund $C = 5$, we can choose a trial combination as follows:

- UT01 and UT02 with a total quality score of $5 + 5 = 10$ and cost of $3 + 2 = 5$.
- Just UT01 with a total quality score of 5 and cost of 3.
- Just UT02 with a total quality score of 5 and cost of 2.
- Just SL01 with a total quality score of 8 and cost of 5.

From all those combinations, the maximum total quality score is 10.

Input Example 2

```
4
APPLE 4
BANANA 2
COCONUT 7
DURIAN 1
3
UTARA
APPLE 10
BANANA 15
COCONUT 5
DURIAN 10
TIMUR
APPLE 5
BANANA 5
COCONUT 5
DURIAN 5
SELATAN
APPLE 3
BANANA 10
COCONUT 10
DURIAN 3
3
5
RESTOCK UTARA COCONUT 2
PRODUCE UTARA UTR01 3 2 0 3
RESTOCK SELATAN DURIAN 3
RESTOCK SELATAN APPLE 3
PRODUCE SELATAN SS01 1 1 1 3
5
PRODUCE TIMUR SNVC01 3 0 1 2
PRODUCE TIMUR SNVC02 3 1 1 2
RESTOCK TIMUR APPLE 10
RESTOCK TIMUR DURIAN 10
DISTRIBUTE TIMUR 9
5
PRODUCE UTARA UTR02 0 0 3 1
```

```
PRODUCE UTARA SNVC03 1 2 3 10
PRODUCE SELATAN SNVC04 2 4 0 0
DISTRIBUTE UTARA 2
DISTRIBUTE SELATAN 1
<LAST_QUERY>
```

Output Example 2

If <LAST_QUERY> = CEK_KINERJA_PABRIK

```
Hari ke-1: 2 0
Hari ke-2: 3 1
Hari ke-3: 5 4
UTARA 2 2
TIMUR 1 1
SELATAN 2 1
```

If <LAST_QUERY> = CEK_SEMUA_VAKSIN

```
Hari ke-1: 2 0
Hari ke-2: 3 1
Hari ke-3: 5 4
UTR02 23 4 TRUE
SNVC01 22 6 TRUE
UTR01 20 8 TRUE
SNVC04 17 6 FALSE
SS01 17 6 TRUE
```

If <LAST_QUERY> = CEK_TOTAL_KUALITAS_UJICоба 12

```
Hari ke-1: 2 0
Hari ke-2: 3 1
Hari ke-3: 5 4
45
```

Explanation 2

In the second input, there are 4 types of materials, that are: **APPLE**, **BANANA**, **COCONUT**, and **DURIAN**. And there are 3 factories that are named **UTARA**, **TIMUR**, and **SELATAN**. Below is the table of information regarding the material and the initial stocks at each factory.

Material name	Quality score	Stock in UTARA	Stock in TIMUR	Stock in SELATAN
APPLE	4	10 pcs	5 pcs	3 pcs
BANANA	2	15 pcs	5 pcs	10 pcs
COCONUT	7	5 pcs	5 pcs	10 pcs
DURIAN	1	10 pcs	5 pcs	3 pcs

The following is the explanations regarding queries done each day.

Day-1

On day 1, there are 5b queries performed, they are:

- **RESTOCK UTARA COCONUT 2**
This query increases the amount of **COCONUT** material in **UTARA** factory to 7.
- **PRODUCE UTARA UTR01 3 2 0 3**
This query will produce UTR01 vaccine and stored in UTARA factory warehouse. The UTR01 vaccine will have a quality value $(3 \times 4 + 2 \times 2 + 0 \times 7 + 3 \times 1) \bmod 1000 + 1 = 20$ and a trial cost of $3 + 2 + 0 + 3 = 8$. Material stock decreases according to the amount of material used.
- **RESTOCK SELATAN DURIAN 3**
This query increases the number of DURIAN materials in SELATAN factory to 6.
- **RESTOCK SELATAN APPLE 3**
This query increases the amount of APPLE Materials in SELATAN factory to 6.
- **PRODUCE SELATAN SS01 1 1 1 3**
This query will produce SS01 vaccine and stored in SELATAN factory warehouse. SS01 vaccine will have a quality value of $(1 \times 4 + 1 \times 2 + 1 \times 7 + 3 \times 1) \bmod 1000 + 1 = 17$ and a trial cost of $1 + 1 + 1 + 3 = 6$. Material stock decreases according to the amount of material used

The following is a breakdown of stocks and vaccines made on day-1

Material Name	Quality	Stock in UTARA	Stock in TIMUR	Stock in SELATAN
APPLE	4	7 pcs	5 pcs	5 pcs
BANANA	2	13 pcs	5 pcs	9 pcs
COCONUT	7	7 pcs	5 pcs	9 pcs
DURIAN	1	7 pcs	5 pcs	3 pcs

Vaccine Name	Quality	Trial cost	Created by	Vaccine Location
UTR01	20	8	UTARA factory	UTARA factory
SS01	17	6	SELATAN factory	SELATAN factory

As of Day-1, there're 2 vaccines that have already been produced and no vaccines are in the vaccine bank of Ministry of Health, so it is printed Hari ke-1 : 2 0

Day-2

In Day-2, there are 5 queries have to be done, namely:

- **PRODUCE TIMUR SNVC01 3 0 1 2**
This query will produce SNVC01 vaccine and stored in TIMUR factory warehouse. SNVC01 vaccine will have a quality value of $(3 \times 4 + 0 \times 2 + 1 \times 7 + 2 \times 1) \bmod 1000 + 1 = 22$ and a trial cost of $3 + 0 + 1 + 2 = 6$. Material stock decreases according to the amount of material used
- **PRODUCE TIMUR SNVC02 3 1 1 2**
This query was supposed to produce SNVC02 vaccine and stored in TIMUR factory warehouse. But unfortunately, the materials of APPLE are only 2 left after we run the last query. Therefore this query is ignored

- **RESTOCK TIMUR APPLE 10**
This query *increases the amount of APPLE materials in the TIMUR factory warehouse* to 10.
- **RESTOCK TIMUR DURIAN 10**
This query *increases the amount of DURIAN materials in the TIMUR factory warehouse* to 10.
- **DISTRIBUTE TIMUR 9**
This query distributes 9 vaccines in the EAST factory warehouse. But because UTARA factory has only 1 vaccine (SNVC01), therefore only 1 vaccine that is distributed.

The following is a breakdown of stocks and vaccines made on day-2

Material Name	Quality	Stock in UTARA	Stock in TIMUR	Stock in SELATAN
APPLE	4	7 pcs	12 pcs	5 pcs
BANANA	2	13 pcs	5 pcs	9 pcs
COCONUT	7	7 pcs	4 pcs	9 pcs
DURIAN	1	7 pcs	13 pcs	3 pcs

Vaccine Name	Quality	Trial cost	Created by	Vaccine Location
UTR01	20	8	UTARA factory	UTARA factory
SS01	17	6	SELATAN factory	SELATAN factory
SNVC01	22	6	TIMUR factory	Vaccine Bank

As of Day-2, there're 3 vaccines that have already been produced and 1 vaccine are in the vaccine bank of Ministry of Health, so it is printed Hari ke-2 : 3 1

Day-3

In Day-3, there are 5 queries have to be done, that is:

- **PRODUCE UTARA UTR02 0 0 3 1**
This query produces the UTR02 vaccine and is stored in the UTARA factory warehouse. The UTR02 vaccine will have a quality value $(0 \times 4 + 0 \times 2 + 3 \times 7 + 1 \times 1) \bmod 1000 + 1 = 23$ and a trial cost of $0 + 0 + 3 + 1 = 4$. Material stock decreases according to the amount material used.
- **PRODUCE UTARA SNVC03 1 2 3 10**
This query produces the SNVC03 vaccine and is stored in the UTARA factory warehouse. But unfortunately, the materials of DURIAN left are only 6 after we run the last query. Therefore this query is ignored
- **PRODUCE SELATAN SNVC04 2 4 0 0**
This query produces the SNVC04 vaccine and is stored in the SELATAN factory warehouse. The SNVC04 vaccine will have a quality value $(2 \times 4 + 4 \times 2 + 0 \times 7 + 0 \times 1) \bmod 1000 + 1 = 17$ and a trial cost of $2 + 4 + 0 + 0 = 6$. Material stock decreases according to the amount material used.
- **DISTRIBUTE UTARA 2**
This query distributes 2 vaccines in the UTARA factory warehouse. UTARA factory has 2 vaccines (UTR01 and UTR02), so those vaccines are distributed.
- **DISTRIBUTE SELATAN 1**
This query distributes 1 vaccine in the SELATAN factory warehouse. SELATAN factory has 2

vaccines (SS01 and SNVC02), so SSC01 vaccine is the only one distributed, because it was produced earlier than SNVC02 vaccine.

The following is a breakdown of stocks and vaccines made on day-3

Material Name	Quality	Stock in UTARA	Stock in TIMUR	Stock in SELATAN
APPLE	4	7 pcs	12 pcs	3 pcs
BANANA	2	13 pcs	5 pcs	5 pcs
COCONUT	7	4 pcs	4 pcs	9 pcs
DURIAN	1	6 pcs	13 pcs	3 pcs

Vaccine Name	Quality	Trial cost	Created by	Vaccine Location
UTR01	20	8	UTARA factory	Vaccine Bank
SS01	17	6	SELATAN factory	Vaccine Bank
SNVC01	22	6	TIMUR factory	Vaccine Bank
UTR02	23	4	UTARA factory	Vaccine Bank
SNVC04	17	6	SELATAN factory	SELATAN factory

As of Day-3, there're 5 vaccines that have already been produced and 4 vaccines are in the vaccine bank, so it is printed Hari ke-3 : 5 4

<LAST_QUERY>

- CEK_KINERJA_PABRIK

Below are the work details for each factory.

- **UTARA factory**

- Produce **2 pcs** of vaccines (UTR01 and UTR02)
 - Distribute **2 pcs** of vaccines (UTR01 and UTR02) to the Vaccine Bank

- **TIMUR factory**

- Produce **1 pcs** of vaccine (SNVC01)
 - Distribute **1 pcs** of vaccine (SNVC01) to the Vaccine Bank

- **SELATAN factory**

- Produce **2 pcs** vaccines (SS01 and SNVC04)
 - Distribute **1 pcs** of vaccine (SS01) to the Vaccine Bank

- CEK_SEMUA_VAKSIN

The order of vaccines according to quality, trial cost, and name is UTR02, SNVC01, UTR01, SNVC04, and SS01. SNVC04 and SS01 have the same quality and trial cost but SNVC01 is lexicographically smaller than SS01.

- CEK_TOTAL_KUALITAS_UJICOBA 12

Currently there are 4 vaccines in the Vaccine Bank. With a maximum fund of C = 12, the following trial combinations can be selected:

- UTR01 and UTR02 with a total quality of $20 + 23 = 43$ and cost $8 + 4 = 12$.
- SS01 and SNVC01 with a total quality of $17 + 22 = 39$ and cost $6 + 6 = 12$.
- SS01 and UTR02 with a total quality $17 + 23 = 40$ and cost $6 + 4 = 10$.
- SNVC01 and UTR02 with a total quality $22 + 23 = 45$ and cost $6 + 4 = 10$.
- Only UTR01, only SS01, only SNVC01, and only UTR02.

From all the combinations, the maximum total of quality is 45.

Additional Information

Use the correct ADT to help you reduce the complexity of the algorithm. Sorting with the help of Python functions or methods such as **sorted()**, **sort()**, **using PriorityQueue, etc.** that makes it possible to do automatic sorting are **NOT** allowed (you must write your own sort function).

Test case distributions

Variation of <LAST_QUERY>	Test case range
CEK_KINERJA_PABRIK	1 - 10
CEK_SEMUA_VAKSIN	11 - 30
CEK_TOTAL_KUALITAS_UJICOBA C	31 - 50