**A Maximum Sum Array**

In this problem, you will be given an array of integers. You have to find a sub array of the given array where the sum of the integers of that sub array is maximum among all the sub arrays.

**Input**

First line contains an integer **N (1<= N <= 1000000)**, indicating the number of integers in the array. Next line contains **N** space separated integers, **x (-1000000 <= x <= 1000000)**.

**Output**

You have to print the sum of the integers in the sub-array mentioned in this problem.

|  |  |
| --- | --- |
| **Sample input** | **Sample output** |
| 5  1 3 0 5 4 | 13 |
| 5  -10 2 4 8 -3 | 14 |

# **B Magic Set**

You know Tom and Jerry. They like to play together all the time. One day, Tom was busy with someone. Naturally, Jerry was disturbing him. Tom made a plan and asked Jerry a question to get rid of him for a while. Tom asked to play a game. Rules of the play are:

* Tom gives a set of integers to Jerry.
* Jerry Needs to find all the subsets of the given set.
* Then, Jerry needs to find the product of the integers for each subset.
* One of the subset is called magic set where the product of the elements are maximum. Jerry needs to find this array.

## **Input**

First line consists of a integer N (**0 <= N <= 1000**) indicating the number of elements of the set. Next line contains the elements of the set. All the elements are integers in range of **[-100000, 100000].**

## **Output**

Print the magic set. All the elements of the set should be printed in ascending order and space separated. If there is multiple possible set, then print the set with maximum number of elements.

|  |  |
| --- | --- |
| **Sample input** | **Sample output** |
| 5  -1 3 4 7 8 | 3 4 7 8 |
| 5  1 2 3 4 7 | 1 2 3 4 7 |

# **C Set Operations**

You will be given **N** disjoint sets containing positive integers. Your task is to conditionally perform some operations on those sets according to the command in the following table:

|  |  |
| --- | --- |
| **Command** | **Explanation** |
| **1 k** | Print the set containing the item k.  If k does not exist, ignore the command. |
| **2 k1 k2** | Union the sets containing item k1 and k2. Ignore the command if k1 and k2 belong to same set or any of them does not exist.  If command is not ignored, print the set |
| **3 k1 k2** | Move item k1 from its current set to the set containing item k2. If k1 and k2 both belong to same set or k1 does not exist, ignore the command.  If k2 does not exist create a new set and move the item k1 to that set.  If the command was not ignored, print all the modified sets. Order of the print will be:  New set of k1  Old set of k1 |

Note: elements of a set will be printed in ascending order.

**Input:**

An integer N, denoting number of sets.

Next N lines contains the elements of each set. A 0 denotes the end of a set.

An integer C identifying number of commands to perform

Next C lines contains C commands as stated in the table above.

**Output:**

Output of the commands

**Condition:**

**1 <= N, Elements in a set, k1, k2 <100000**

**1 <= C <= 50**

**k1 is never equal to k2**

|  |  |
| --- | --- |
| **Sample Input** | **Sample Output** |
| 4  1 2 3 4 5 0  6 7 8 9 0  10 11 12 13 14 0  15 0  7  1 13  2 2 13  3 13 16  3 13 15  2 8 13  3 16 13  1 16 | {10, 11, 12, 13, 14}  {1, 2, 3, 4, 5, 10, 11, 12, 13, 14}  {13}  {1, 2, 3, 4, 5, 10, 11, 12, 14}  {13, 15}  {}  {6, 7, 8, 9, 13, 15} |

**D Book Fair**

Book fair is very popular in the whole world. For COVID-19 pandemic, Book fair is going to be held online in various developed countries. As a consequence, maintaining online databases of different books is very much needed. In this problem, you will be given a list of books to store and answer some queries from users. Format of the queries are as follows:

* 1 **genre**: You need to print all the Books under the **genre** ascending order according to their ids.
* 2 **year**: You need to print all the Books published in the **year** ascending order according to their ids.

**Input**

First line is an integer **N (1<= N <= 1000)** which indicates the number of books.

Next **N** lines will provide the information of the books containing **id** (a 4-digit integer), **title** (length will be no more than 25), **genre** (length will be no more than 25), **author** (length will be no more than 25) and **publication year** (a 4-digit integer). Note that all the names will contain only alphabets.

Next line will contain an integer **T (1 <= T <= 1000)** indicating the number of queries.

Next **T** lines each contains a query according to the description.

**Output**

For each line of query, print the necessary things. See the sample for print format.

|  |  |
| --- | --- |
| **Sample Input** | **Sample Output** |
| 3  1254 Mrittu Vuture AmiLikhiNai 2020  3542 Khaoa Oitihashik GopalVar 1800  9547 Ghum Pouranik KumvKoron 1700  3  1 Pouranik  1 Oitihashik  2 2020 | 9547 Ghum Pouranik KumvKoron 1700  3542 Khaoa Oitihashik GopalVar 1800  1254 Mrittu Vuture AmiLikhiNai 2020 |
| 6  1254 Mrittu Vuture AmiLikhiNai 2020  3542 Khaoa Oitihashik GopalVar 1800  9547 Ghum Pouranik KumvKoron 1700  1354 Mrittu Vuture AmiLikhiNai 2020  7542 Khaoa Oitihashik GopalVar 1800  8547 Ghum Pouranik KumvKoron 1700  3  2 1500  1 Vuture  2 1700 | 1254 Mrittu Vuture AmiLikhiNai 2020  1354 Mrittu Vuture AmiLikhiNai 2020  8547 Ghum Pouranik KumvKoron 1700  9547 Ghum Pouranik KumvKoron 1700 |