

快取取代演算法 (Cache Replacement Algorithm)

問題敘述

LRU (Least Recently Used) 是一種常見的快取取代演算法 (Cache Replacement Algorithm)，這種演算法的取代策略是『資料越久未曾被使用，將越優先被置換掉』。

我們將模擬兩種方式來實作 LRU。第一種方式會將資料依最近一次取用的時間擺置，取用時間點越近的擺在越上面。若有資料需加入快取而快取已滿時，則會將最下面的資料擠出快取。舉例而言，假設快取最多可以容納 3 個資料，而被存取的資料依序為 4 7 0 7 1 0 1 2 1 2 7 1 2 7 1 2，則快取的內容變化如下圖所示：

		0	7	1	0	1	2	1	2	7	1	2
	7	7	0	7	1	0	1	2	1	2	7	1
4	4	4	4	0	7	7	0	0	0	1	2	7

第二種方式使用計數器的方式來實作，每個快取頁面會對應到一個計數器，資料存到快取頁面時，對應的計數器會記錄目前的時間戳記。當快取滿的時候，系統會檢視看哪一個快取頁面對應的時間戳記最舊，將新的資料存到那個快取頁面，並更新該頁面的時間戳記。舉例來說，假設快取最多可以容納 3 個資料，而被存取的資料依序為 4 7 0 7 1 0 1 2 1 2 7 1 2 7 1 2，則快取的內容變化如下圖所示：

4	4	4	4	1	1	1	1	1	1	1	1	1
	7	7	7	7	7	7	2	2	2	2	2	2
		0	0	0	0	0	0	0	0	7	7	7

請你寫程式來模擬 LRU 快取取代演算法的這兩種實作方式。

輸入格式

輸入的第一行有兩個正整數 N ($1 \leq N \leq 50,000$) 與 C ，以空格來間隔。第一個數字 N 代表有多少個快取頁面。第二個數字 C 有可能是 1、2 或 3，用來指定所需輸出的資訊。第二行是一連串的整數 (不超過 500,000 個)，以空格來區隔，代表一連串的存取資料。第三行是一個查詢資料 K 。

輸出格式

若第一行輸入的 $C=1$ ，請你的程式判斷經過輸入第二行一連串的資料存取之後，資料 K 是否在快取裡。若是，請輸出 YES，否則輸出 NO。

若第一行輸入的 $C=2$ ，請根據第一種方式實作 LRU，並由上到下輸出最後快取的內容。

若第一行輸入的 $C=3$ ，請根據第二種方式實作 LRU，並由上到下輸出最後快取的內容。

輸入範例 1 3 1 4 7 0 7 1 0 1 2 1 2 7 1 2 2	輸出範例 1 YES
--	----------------------

輸入範例 2 3 2 25 11 3 200 18 11 7 5 93 11	輸出範例 2 93 5 7
--	-------------------------

輸入範例 3 3 3 25 11 3 200 18 11 7 5 93 11	輸出範例 3 7 5 93
--	-------------------------

輸入範例 4 5 2 1 2 3 2 2	輸出範例 4 2 3 1
--------------------------------------	------------------------

(輸出時若有未使用的快取頁面，跳過不輸出。)

評分說明

本題共有三組測試題組，條件限制如下所示。每一組可有一或多筆測試資料，該組所有測試資料皆需答對才會獲得該組分數。

子任務	分數	額外輸入限制
1	20	$1 \leq N \leq 5$ ，且資料不超過 10 個
2	30	$1 \leq N \leq 20$ ，且資料不超過 500 個
3	50	$1 \leq N \leq 50,000$ ，且資料不超過 500,000 個