

F. 區間最大獨立集詢問 (Indset)

本題為互動題，限用 C++ 上傳。

問題描述

小雲最近在學習「最大權重獨立集 (Maximum-Weight Independent Set, MWIS)」的演算法。

根據定義，一張無向圖裡的頂點子集合 I 要被稱作「獨立集」的話，需要滿足 I 集合中任兩個頂點在原圖上皆互不相鄰的條件。而「最大權重獨立集」則是所有可能的「獨立集」中，點權重總和最大的一組。

今天小雲發現，假如這張無向圖是一條直鏈 (Chain) 的話，那要找到「最大權重獨立集」變得超級簡單！他向小成分享這件事之後，小成卻反問：「那你知道怎麼有效率的回答一條直鏈上面的『區間最大權重獨立集詢問』嗎？」

經過一番研究後，小雲發現，即使在不知道直鏈上每個節點的具體權重下，也能找到它的最大權重獨立集，甚至能用來解決區間詢問。於是，他列了以下這道難題給小成：

「給定一條包含 n 個頂點編號為 $1, 2, \dots, n$ 的直鏈 (chain)，其中對於任何的 $1 \leq i < n$ ，頂點 i 與頂點 $i + 1$ 之間皆有一條無向邊，且對於任何 $1 \leq i \leq n$ ，頂點 i 的權重為一個正整數 w_i ，請回答 q 筆『區間最大權重獨立集詢問 (Range MWIS Query)』。」

「在區間最大權重獨立集詢問中，對於滿足 $1 \leq l \leq r \leq n$ 的任意區間，你必須回答我頂點 $l, l + 1, \dots, r$ 之間的最大權重獨立集為何。」

小雲接著補充。

「當然，在一無所知的情況下不可能解決這個問題，所以我允許你執行數次『權重和比較詢問』：任選兩個頂點的子集合，我會告訴你哪一個子集合的頂點權重和比較大。」

請協助小成，在執行儘量少次『頂點子集合權重比較』的情況下，回答所有待詢問區間裡的最大權重獨立集！

實作細節

你需要實作兩個函式 `init()` 與 `range_MWIS_query()`：

```
void init(int n);
```

- 對於每一筆測試資料，正式評分程式會呼叫你實作的 `init()` 函式恰好 1 次。
- n 代表頂點的數量。

```
std::vector<int> range_MWIS_query(int l, int r);
```

- 對於每一筆測試資料，正式評分程式會呼叫你實作的 `range_MWIS_query()` 函式恰好 q 次。
- 保證在呼叫完 `init()` 後才會呼叫此函式。
- `range_MWIS_query()` 需要回傳一個陣列 $x_1, x_2, \dots, x_m$ 。
- 陣列 x 代表了該詢問區間的最大權重獨立集包含的頂點編號。
- 對於所有 $1 \leq i \leq m$ ，皆須保證 $l \leq x_i \leq r$ 。
- 對於所有 $1 \leq i < j \leq m$ ，皆須保證 $|x_i - x_j| > 1$ 。

此外，在實作時可以呼叫 `compare_subsets()` 這個函式。

```
bool compare_subsets(const std::vector<int>& a, const std::vector<int>& b);
```

- a 是一個陣列，其描述了 $\{1, 2, \dots, n\}$ 的子集合。
- b 是一個陣列，其描述了 $\{1, 2, \dots, n\}$ 的子集合。
- a 內不能有重複的數字。
- b 內不能有重複的數字。
- 若集合 a 內的頂點的權重和比集合 b 小，則該函式會回傳布林值 `true`，否則會回傳布林值 `false`。
- 範例評分程式內的 `compare_subsets()` 實作與實際評分程式內的實作完全相同。

互動範例

一個可能被評為 Accepted 的互動例子顯示如下：

評分程式端	參賽者端
呼叫 <code>init(5)</code> 。	
	呼叫 <code>compare_subsets([1], [2])</code> 。
回傳 <code>true</code> 。	
	呼叫 <code>compare_subsets([3], [4])</code> 。
回傳 <code>false</code> 。	
	回傳 <code>void()</code>
呼叫 <code>range_MWIS_query(2, 5)</code>	
	呼叫 <code>compare_subsets([2, 5], [1, 3, 5])</code> 。
回傳 <code>true</code> 。	
	回傳 <code>[2, 4]</code>
呼叫 <code>range_MWIS_query(1, 5)</code>	
	回傳 <code>[1, 3, 5]</code>

測資限制

- $1 \leq n \leq 2\,000$ 。
- $1 \leq q \leq 2\,000$ 。
- $1 \leq l \leq r \leq n$ 。

範例評分程式

範例評分程式採用以下格式輸入：

```
n q
w1 w2 ... wn
l1 r1
l2 r2
⋮
lq rq
```

請注意，正式的評分程式一定不會採用以上格式輸入。請不要自行處理輸入輸出。

範例評分程式首先呼叫 `init(n)`，接著範例評分程式會呼叫 q 次 `range_MWIS_query(li, ri)`。接著，若範例評分程式偵測到從 `init` 或 `range_MWIS_query` 對 `compare_subsets` 的呼叫有任何不合法，此程式將輸出

```
Wrong Answer: msg
```

後並終止程式執行，其中 msg 為下列其中之一錯誤訊息：

- **Invalid vertex number: v:** 你的程式傳入 `compare_subsets` 的集合中有不介在 $1 \sim n$ 之間的數字 v 。
- **Duplicate vertex numbers: v:** 你的程式傳入 `compare_subsets` 的集合中有重複的數字 v 。

否則，範例評分程式將會以下列格式印在標準輸出中：

```
m1
x1,1 x1,2 ... x1,m1
m2
x2,1 x2,2 ... x2,m2
⋮
mq
xq,1 xq,2 ... xq,mq
Accepted: Qinit Qquery
```

其中，

- m_i 為第 i 次呼叫 `range_MWIS_query()` 時你回傳的陣列長度。
- $x_{i,j}$ 為第 i 次呼叫 `range_MWIS_query()` 時你回傳的陣列的第 j 項。
- Q_{init} 與 Q_{query} 為根據你的程式呼叫 `compare_subsets` 的次數得來的數值，詳細定義請見評分說明欄位。

評分說明

對於每一筆測試資料，若你的程式在函式 `init()` 中呼叫 `compare_subsets` 的次數為 x 、在第 i 次 `range_MWIS_query()` 中呼叫 `compare_subsets` 的次數為 y_i ，則定義 Q_{init} 與 Q_{query} 為：

$$\begin{cases} Q_{init} = \left\lceil \frac{x}{n} \right\rceil \\ Q_{query} = \max_{1 \leq i \leq q} y_i \end{cases}$$

根據 Q_{init} 與 Q_{query} ，你將得到兩個分數比重 W_{init} 與 W_{query} ：

$$W_{init} = \begin{cases} 1.0 & \text{若 } Q_{init} \leq 3 ; \\ 1.0 - 0.07 \cdot (Q_{init} - 3) & \text{若 } 3 < Q_{init} \leq 10 ; \\ 0.5 - 0.04 \cdot (Q_{init} - 10) & \text{若 } 10 < Q_{init} \leq 20 ; \\ 0 & \text{若 } Q_{init} > 20。 \end{cases}$$

$$W_{query} = \begin{cases} 1.0 & \text{若 } Q_{query} \leq 1 ; \\ 1.0 - 0.1 \cdot (Q_{query} - 1) & \text{若 } 1 < Q_{query} \leq 10 ; \\ 0 & \text{若 } Q_{query} > 10。 \end{cases}$$

你的最終比重 W 會是兩者相乘，也就是：

$$W = W_{init} \cdot W_{query}$$

本題共有 3 組子任務，條件限制如下所示。每一組可有一或多筆測試資料，你在該子任務的得分為所有測試資料中分數比重 W 的最小值，乘以該子任務的總分。

子任務	分數	額外輸入限制
1	16	$l = 1$ 。
2	11	對於所有詢問的區間 $[l, r]$ ，區間 $[1, r]$ 的最大權重獨立集唯一且不包含頂點 $l + 1$ 。
3	73	無額外限制。