



第二題：登入電腦 (Login)

本題為互動題。

問題描述

阿強家中有一台電腦，登入會有很多遊戲可以玩，但是不見得可以登入成功。若登入失敗電腦就會被鎖定一整天。登入畫面由 n 個圓點組成，任兩個圓點之間皆有一條線連接，線有可能是紅線，也有可能是藍線，沒有第三種可能的顏色。

阿強每一次可以選取一些圓點，然後電腦會自動掃描過所有兩端都被選取的線，並改變其顏色（若原本是紅色，就會變成藍色、若原本是藍色就會變成紅色）。如果阿強可以將所有的線都變成紅色的，那就能夠順利登入。

不過最近阿強電腦的登入畫面被新版的螢幕保護程式蓋過去了：所有的紅線和藍線通通被隱藏起來了。取而代之的是，阿強每次可以選擇一個圓點和一個整數 $d \geq 2$ ，並且藉此得知該圓點透過不超過 d 條紅線數能到達的圓點們。不過每進行一次操作就會耗掉不少時間，你能幫幫阿強在天黑之前正確地登入電腦嗎？



實作細節

你需要完成以下函式：

```
void Login(int n);
```

- `Login(n)` 這個函式的參數 n 是登入畫面的圓點數量。這些圓點的編號是 $1, 2, \dots, n$ 。
- 對於每一筆測試資料，評分程式只會呼叫你實作的 `Login` 函式恰好 1 次。

你的程式可以呼叫以下輔助函式：

```
int PerfectScoreOperationLimit();  
void ReplaceByComplement(const std::vector<int>& subset);  
std::vector<int> ReachableVertices(int vertex, int distance);  
void Reset();
```

- `PerfectScoreOperationLimit()` 函式會回傳一個整數 M （請參閱評分說明）。
- 輔助函式的總呼叫次數不得超過 $10M$ 次。
- `ReplaceByComplement(S)` 函式會將所有登入畫面中，那些兩端皆在集合 S 內的所有邊變換其顏色。
- 集合 S 的大小至少為 2，所有數字皆相異且介於 1 到 n 之間。
- `ReachableVertices(v, d)` 函式會回傳從編號 v 的圓點沿著不超過 d 條紅色的線能抵達的圓點編號。
- $1 \leq v \leq n$ 。
- $2 \leq d \leq n$ 。
- `Reset()` 函式會將狀態回溯至初始狀態。
- 評分程式呼叫並執行 `Login(n)` 函式後，所有登入畫面上的線段必須要是紅色。

如果不符合上述條件限制，你的程式會被判為 **Wrong Answer**；否則你的程式會被判斷為 **Accepted**，並以評分說明中的方式來決定分數。

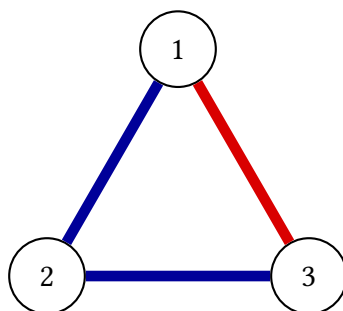


測資限制

- $3 \leq n \leq 30$ 。
- $M = 10000$ 。
- 可以證明對任意輸入，皆有辦法透過上述函式介面將所有線段都變成紅色。

互動範例

考慮以下的測試資料：



一個被評分程式判斷為 **Accepted** 的互動例子顯示如下：

評分程式端	參賽者端
呼叫 <code>Login(3)</code> 。	
	呼叫 <code>ReachableVertices(1, 2)</code> 。
回傳 <code>{1, 3}</code> 。	
	呼叫 <code>ReplaceByComplement({1, 3})</code> 。
	呼叫 <code>ReplaceByComplement({1, 2, 3})</code> 。
	回傳。



評分說明

對於每一筆測試資料，你將會得到的**分數比重** S 值如下：

- 如果你的程式的互動不符合限制，或是 Login 回傳後登入畫面仍有藍色線條，則 $S = 0.0$ 。
- 令 Q 為你的程式呼叫輔助函式的總次數，且 M 為 `PerfectScoreOperationLimit()` 回傳之數值。則

$$S = \begin{cases} 1.0 & \text{若 } Q \leq M \\ M/Q & \text{其他情況} \end{cases}$$

本題共有 4 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，你的得分是該組所有測試資料之得分比重 S 中最低者，乘以該子任務的分數。

子任務	分數	額外輸入限制
1	23	$n \leq 10$ 。
2	12	登入畫面的紅色線條形成一棵樹。
3	37	見備註。
4	28	無額外限制。

- 子任務 3 的備註：登入畫面上，若隨意移除若干圓點以及相鄰的線，對於剩下的任何兩個圓點 u 和 v ，保證有以下三件事情之一發生：
 - (一) u 和 v 之間是一條紅色的線。
 - (二) 存在一個沒有被移除的圓點 w ，使得 (u, w) 和 (w, v) 都是紅色的線。
 - (三) 不存在一條從 u 到 v 僅由紅色線條組成的路徑。



範例評分格式

範例評分程式以下列格式讀取輸入：

第一列有一個數字 n ，代表登入畫面的圓點個數。接下來有 $n - 1$ 列，第 i 列有一個長度為 $n - i$ 的字串，所有字串都只以 R 和 B 字元組成。第 i 列字串的第 j 個字元 (i, j 均從 1 開始編號) 代表連接編號為 i 和 $i + j$ 的圓點之間的線條顏色：R 代表紅色、B 代表藍色。

```

n
C1,2C1,3C1,4 ... C1,n
C2,3C2,4 ... C2,n
⋮
Cn-1,n

```

請注意: 使用自己上傳的測試資料進行測試時, 沒有下列 MSG 描述的情形時你總會得到 **Accepted**。如果你的程式被評為 **Accepted**，範例評分程式會輸出你的程式呼叫輔助函式的總次數，以及 S 之值。

如果你的程式被評為 **Wrong Answer**，範例評分程式輸出 Wrong Answer: MSG，其中 MSG 格式與意義如下：

- failed to login：函式執行完畢以後，登入畫面仍然存在藍色的線。
- too many queries!：使用輔助函式超過 $10M$ 次。
- bad subset：傳給 ReplaceByComplement 的參數不符合規定。
- bad source vertex：傳給 ReachableVertices 的參數不符合規定。
- bad distance：傳給 ReachableVertices 的參數不符合規定。
- corrupted stub：可能改到 stub 了，請重新下載 stub.cpp。