

1.0 second(s), 16 MB

เวลาเขียนโปรแกรมภาษา C เราสามารถเอาเนื้อหาของไฟล์หนึ่ง ๆ เข้ามาใส่ในไฟล์อีกไฟล์หนึ่ง ได้ด้วยการใช้คำสั่ง `#include` ยกตัวอย่างเช่น ถ้า `main.c` และ `lib.h` มีเนื้อหาดังต่อไปนี้

<b>main.c</b>	<b>lib.h</b>
<code>#include "lib.h"</code> <code>int main() { return f(); }</code>	<code>int f() { return 0; }</code>

เวลาคอมไพล์ คอมไพเลอร์ภาษา C จะเขียนเนื้อหาของ `main.c` ใหม่ โดยเอาเนื้อหาของ `lib.h` ไปแทรกไว้ที่คำสั่ง `#include "lib.h"` ใน `main.c` ดังนั้น เนื้อหาใหม่ของ `main.c` คือ

<b>main.c (หลังถูกคอมไพเลอร์ประมวลผล)</b>
<code>int f() { return 0; }</code> <code>int main() { return f(); }</code>

อนึ่ง ในไฟล์ที่ถูกไฟล์อื่น include อาจมีคำสั่ง `#include` อยู่ก็ได้ และในไฟล์หนึ่งอาจมีการ include ไฟล์อื่นๆ มากกว่าหนึ่งไฟล์ได้ ยกตัวอย่างเช่น

<b>main.c</b>	<b>lib1.h</b>
<code>#include "lib1.h"</code> <code>#include "lib3.h"</code> <code>int main() { return f1()+f3(); }</code>	<code>#include "lib2.h"</code> <code>int f1() { return f2(); }</code>
<b>lib2.c</b>	<b>lib3.h</b>
<code>int f2() { return 2; }</code>	<code>int f3() { return 3; }</code>

จะได้ว่าเวลาคอมไพเลอร์ภาษา C จะเขียนเนื้อหาของ `main.c` ใหม่ ดังนี้

<b>main.c (หลังถูกคอมไพเลอร์ประมวลผล)</b>
<code>int f2() { return 2; }</code> <code>int f1() { return f2(); }</code> <code>int f3() { return 3; }</code> <code>int main() { return f1()+f3(); }</code>

อย่างไรก็ดีหากผู้เขียนโปรแกรมไม่ระมัดระวังก็อาจทำให้เกิดปัญหาได้สองประการคือ

1. ไฟล์เดียวกันถูก include มากกว่าหนึ่งครั้ง เช่น

<b>main.c</b>	<b>lib1.h</b>
<code>#include "lib1.h"</code> <code>#include "lib3.h"</code> <code>int main() { return f1() +f3(); }</code>	<code>#include "lib2.h"</code> <code>int f1() { return f2(); }</code>
<b>lib2.c</b>	<b>lib3.h</b>
<code>int f2() { return 2; }</code>	<code>#include "lib2.h"</code> <code>int f3() { return 3+f2(); }</code>

ในกรณีเมื่อคอมไพเลอร์ทำการคอมไพล์ main.c ไฟล์ lib2.c จะถูก include สองครั้ง ครั้งหนึ่ง จากไฟล์ lib1.h และอีกครั้งจาก lib3.h ซึ่ง ทำให้ฟังก์ชัน f2() ถูกนิยามสองครั้ง ซึ่ง อาจทำให้คอมไพล์ไม่ผ่านได้

2. ไฟล์ include กันเป็นวงกลม เช่น

<b>main.c</b>	<b>lib1.h</b>
<code>#include "lib1.h"</code> <code>int main() { return f1()+f3(); }</code>	<code>#include "lib2.h"</code> <code>int f1() { return f2(); }</code>
<b>lib2.c</b>	<b>lib3.h</b>
<code>#include "lib3.h"</code> <code>int f2() { return 2; }</code>	<code>#include "lib1.h"</code> <code>int f3() { return 3+f2(); }</code>

สังเกตว่าเมื่อคอมไพเลอร์ภาษา C คอมไพล์ไฟล์ main.c แล้วไฟล์ lib1.h จะ include ไฟล์ lib2.h ซึ่ง include ไฟล์ lib3.h ซึ่ง include ไฟล์ lib1.h และสามารถวนไปเช่นนี้เรื่อย ๆ โดยไม่จำกัด

### งานของคุณ

กำหนดไฟล์โปรแกรมภาษา C มาให้ N ไฟล์ แต่ละไฟล์จะถูกระบุด้วยตัวเลขตั้งแต่ 1 ถึง N พร้อมทั้ง ข้อมูลว่าไฟล์ใด include ไฟล์ใดบ้าง จงเขียนโปรแกรมเพื่อตอบคำถามว่าสำหรับไฟล์ทุกๆ ไฟล์ เมื่อคอมไพเลอร์ภาษา C ทำการคอมไพล์ไฟล์นั้นแล้วจะเกิดปัญหาใดปัญหาหนึ่ง ในสองปัญหาที่กล่าวถึงข้างบนหรือไม่

### ข้อมูลนำเข้า

บรรทัดแรก มีจำนวนเต็ม N ( $1 \leq N \leq 1,000$ )

บรรทัดที่ i+1 (สำหรับทุก ๆ  $1 \leq i \leq N$ ) บอกว่าไฟล์ i ทำการ include ไฟล์ใดบ้าง ซึ่ง บรรทัดที่ i + 1 จะระบุในรูปแบบดังนี้

k a1 a2 ... ak

โดยที่  $k$  และทุก ๆ  $a_j$  ที่  $1 \leq j \leq k$  เป็นจำนวนเต็มที่ไม่เป็นลบและมีค่าไม่เกิน  $N$  บรรทัดดังกล่าวมีความหมายว่าไฟล์  $i$  ทำการ include ไฟล์  $a_1, a_2, \dots$ , และ  $a_k$  เรารับประกันว่าเลข  $a_j$  จะมีค่าไม่ซ้ำกัน (ในไฟล์หนึ่ง ๆ จะไม่มีการ include ไฟล์เดียวกันซ้ำสองครั้ง) และในบรรทัดที่  $i + 1$  จะไม่มี  $a_j$  ใดๆ ที่มีค่าเท่ากับ  $i$  (ไฟล์แต่ละไฟล์จะไม่ include ตัวเอง) นอกจากนี้รับประกันว่าจำนวนการ include ทั้งหมดจะไม่เกิน 3,000 ครั้ง

### ข้อมูลส่งออก

มี  $N$  บรรทัด แต่ละบรรทัดมีข้อความ YES หรือ NO ถ้าเมื่อคอมไพเลอร์ภาษา C คอมไพล์ไฟล์  $i$  แล้วเกิดปัญหาใดปัญหาหนึ่งในปัญหาสองข้อข้างต้น ให้พิมพ์ YES ในบรรทัดที่  $i$  หากไม่เกิดปัญหาใดขึ้นเลย ให้พิมพ์ NO

ที่มา: Young Thai Online Programming Competition 2008

ข้อมูลนำเข้า	ข้อมูลส่งออก
2	NO
1 2	NO
0	
3	YES
1 2	YES
1 3	YES
1 1	
4	YES
2 2 3	NO
1 4	NO
1 4	NO
0	