ישנם N מועמדים להשתתף במשחקי הרעב. מועמדים אלו נדרשים להסתדר בתור להרשמה. כל מועמד מיוצג על-ידי מספר שלם בין 0 ל – N-1. התור מיוצג על-ידי מערך של מספרים שלמים, בו המועמד הראשון בתור נמצא באינדקס 0, המועמד השני בתור באינדקס 1 וכך הלאה.
עקב אווירת החשדנות הכללית, אף אחד לא מוכן שאחד מ 3 האנשים שמאחוריו בתור יהיה מישהו שמאיים עליו. כלומר, האחרון בתור אדיש לסדר של התור, המועמד שלפני האחרון לא מוכן שהמועמד האחרון יאיים עליו, ובאופן כללי מועמד במקום i לא מוכן שהמועמדים במקום i+1, i+2, i+3 יאיימו עליו.
נתונה מטריצה דו מימדית threats המתארת את יחסי האיום בין המועמדים. במיקום threats[i][j] מסומן 1 אם מועמד j מאיים על מועמד i ו 0 אחרת. לא ניתן להניח שהמטריצה סימטרית, כלומר i יכול לאיים על j בלי ש j יאיים על i.

עליכם לכתוב פונקציה

```
bool orderQueue(int threats[N][N], int queue[N]);
```

המקבלת את מטריצת האיומים וכותבת לתוך המערך queue את מספר המועמד בכל מקום בתור.
אם לא ניתן לסדר את התור יש להחזיר false. אחרת יש להחזיר true.

לדוגמה, עבור N=4 המטריצה:

	0	0	0
1		0	0
1	1		0
1	1	1	

קיים פתרון אחד, התור (משמאל לימין):

0	1	2	3

והפונקציה תחזיר true.

הסבר: מועמד 3 מאוים מכל שאר המתמודדים, ולכן הוא חייב להיות אחרון (אף אחד לא יכול לעמוד אחריו).
מועמד 2 מאוים ממתמודדים 0 ו – 1 ולכן חייב להיות לפני האחרון.
מועמד 1 מאוים ממועמד 0 ולכן מועמד 1 יהיה שני בתור, ו – 0, עליו אף אחד לא מאיים, יהיה ראשון.

לעומת זאת, עבור המטריצה:

	1	0	0
1		0	0
1	1		0
1	1	1	

אין פיתרון (כי שחקנים 0 ו- 1 מאיימים אחד על השני, ויש רק ארבעה מקומות בתור) והפונקציה תחזיר false.