

### Esercizio a tempo n.3 del 2/4/2020

Un array `bool B[6][6]` rappresenta un campo minato, dove `false` rappresenta una mina e `true` un punto senza pericolo. Si tratta di trovare un passaggio che attraversi solo posizioni senza pericolo (`true`) e che porti dalla prima all'ultima riga di B. Ad ogni passo del cammino si deve avanzare di una riga e ci si può muovere, rispetto alla colonna attuale, sulla colonna di sinistra, sulla stessa colonna o sulla colonna di destra. Ovviamente se siamo su una colonna al margine di B (colonna 0 o 5) non è possibile muovere a sinistra/destra.

Esempio1: per semplicità consideriamo che B sia 4x4 sia riempita come segue (0=false e 1=true):

```
0 1 0 0
```

```
0 0 1 0
```

```
0 1 0 0
```

```
1 1 1 1
```

in questo array esistono vari cammini, per esempio: 1,2,1,0, dove indico le colonne usate dal cammino per le righe da 0 a 3.

Quindi in questo caso la risposta che dovremmo stampare sarebbe: un cammino esiste. Invece per l'array:

```
0 1 0 0
```

```
0 0 1 0
```

```
1 0 0 0
```

```
1 1 1 1
```

la risposta da calcolare è: non c'è cammino. Infatti il cammino arriva alla colonna 1, ma non riesce ad arrivare alla riga 2.

Consigli per risolvere il problema. L'idea è semplice: partendo dalla riga 0 di B, si determina una nuova riga 1 tale che contiene 1 dove c'era già 1 e in più quell'elemento è raggiungibile (con le mosse consentite) da un elemento 1 della riga precedente. Se in questo modo si produce una riga di soli 0, significa che non c'è cammino, mentre se si arriva alla riga finale con qualche 1, allora il cammino c'è.

Esempio 2: il primo array verrebbe trasformato dalla procedura appena delineata nel modo seguente:

```
0 1 0 0
```

```
0 0 1 0
```

```
0 1 0 0
```

```
1 1 1 0
```

in cui solo l'ultima riga cambia nel suo ultimo elemento a indicare che nessun cammino arriva in quell'elemento.

Invece per il secondo array, la procedura produrrebbe:

```
0 1 0 0
```

```
0 0 1 0
```

```
0 0 0 0
```

0 0 0 0

la terza riga diventa di tutti 0 e quindi la procedura può terminare con fallimento. La quarta riga non serve neppure calcolarla.

Seguendo questa idea, il main dichiara 2 array bool [6][6], B e B1. Nel primo il main legge i 36 valori 0/1 che rappresentano il campo minato, il secondo serve per contenere le nuove righe calcolate. Il main prevede che vengano realizzate 2 funzioni.

In primo luogo void newriga(bool (\*B)[6], bool (\*B1)[6], int r) che soddisfa le seguenti PRE e POST:

PRE=( $0 \leq r \leq 4$ ) && (le righe r di B e B1 sono definite)

POST=(calcola, nel modo descritto prima, la nuova riga r+1 di B1 usando la riga r+1 di B e la riga r di B1)

La seconda funzione è bool tutta\_false(bool (\*X)[6], int r) che restituisce true sse la riga r di X è tutta di 0.

**Correttezza:** dimostrare la correttezza della funzione newriga rispetto alla PRE e POST date.