

Esercizio 1 del 19/4/2021

In questo esercizio una matrice Palude di booleani 8x8 viene usata per rappresentare una palude, dove i valori true indicano le zone sicure, mentre i false indicano le sabbie mobili. Vogliamo scrivere un programma che data una tale matrice determini se esiste un cammino nella palude che attraversi solo zone sicure (true) e che porti dalla riga 0 alla riga 7. Il cammino deve sempre andare avanti, cioè se sono arrivato alla riga i , al prossimo passo devo andare alla riga $i+1$, naturalmente se è possibile senza cadere nelle sabbie mobili. Inoltre se sono nella posizione j della riga i , potrò andare solo nelle posizioni $j-1$, j e $j+1$ della riga $i+1$. Insomma la prossima posizione deve essere contigua all'attuale. Se nessuna di queste 3 posizioni è sicura, il cammino si ferma. Vediamo un esempio.

Esempio: consideriamo una matrice 4 x 4 per semplicità.

```
0 1 0 1
1 0 1 0
0 0 0 1
0 0 1 0
```

esiste il cammino (0,3)(1,2)(2,3)(3,2) dove la prima componente di ogni coppia è la riga (si parte dalla prima riga e si arriva all'ultima) e la seconda componente è la posizione usata in quella riga.

Se la matrice fosse

```
0 1 0 1
1 0 1 0
0 0 0 1
0 1 0 0
```

non ci sarebbe cammino visto che (3,1) non è contiguo a (2,3).

L'idea suggerita per risolvere l'esercizio è la seguente. Trasformiamo la matrice Palude partendo dalla seconda riga e procedendo verso l'ultima. Per ogni elemento j che sia true della riga corrente i controlliamo se nella riga precedente ($i-1$) c'è un valore true che sia in almeno 1 di queste posizioni: $j-1$, j o $j+1$. Ovviamente dovremo fare attenzione di considerare solo indici di colonna ($j-1$, j e $j+1$) che siano in $0..7$.

Esempio. se applicassimo questo algoritmo alla prima matrice dell'esempio precedente, la matrice non verrebbe modificata, ma se lo applicassimo alla seconda matrice, otterremmo:

```
0 1 0 1
1 0 1 0
0 0 0 1
0 0 0 0
```

che rivela che non ci sono cammini. Attenzione che qualsiasi riga può diventare di soli 0. Ovviamente, non appena questo capitasse l'algoritmo andrebbe fermato visto che non ci sarebbe alcuna cammino.

Si tratta di fare una funzione che trasforma la matrice secondo l'algoritmo descritto prima. Si chiede anche una funzione che nel caso la matrice risultante avesse almeno un 1 nell'ultima riga (che significherebbe che almeno un cammino esiste) ricostruisca in un array Path di 8 interi un cammino tra la prima e l'ultima riga di Palude. L'idea per la funzione che calcola il cammino è di usare la matrice risultante dall'algoritmo, partendo dall'ultima riga e risalendo verso la prima riga seguendo sempre il passo più a sinistra.

Esempio. Se il nostro algoritmo producesse una matrice così:

```
0 1 0 1
0 0 1 1
0 1 0 1
0 0 1 1
```

allora inizieremmo la costruzione del cammino mettendo Path[3]=2, poi Path[2]=1, poi Path[1]=2 e Path[0]=1.

Richiedere di costruire un cammino particolare serve per poter fare i test automatici.

Correttezza:

Specificare la PRE e la POST di tutte le funzioni che definite. Scegliete una funzione importante e dimostrate che è corretta rispetto alla sua PRE e POST.