

Fondamenti di Informatica

Prova d'esame del 14/1/2014

Regole d'esame:

1. È vietato parlare con altri studenti
2. È vietato consultare appunti, dispense, libri, in qualunque formato
3. È vietato tenere accesi i telefoni cellulari

Esercizio

Sia $M=(m_{ij})$ una matrice di dimensione $m \times n$. Un *percorso* su M è una sequenza $(e_0, e_1, \dots, e_{m-1})$ di celle di M tale che:

- $e_0=m_{0,0}$ (cioè la prima cella della sequenza è la cella di posizione 0,0);
- per ogni $i=0,1,\dots,m-1$, $e_i=m_{ij}$ per qualche j tale che $0 \leq j \leq n-1$ (cioè la i -esima cella della sequenza si trova sulla riga i -esima);
- per ogni $i=0,1,\dots,m-2$, se $e_i=m_{ij}$, allora $e_{i+1}=m_{i+1,k}$ con $k \in \{j-1, j, j+1\}$ (cioè ogni cella (tranne la prima) si trova o sulla stessa colonna della cella che la precede, oppure sulla colonna successiva a quella occupata dalla colonna che la precede, oppure sulla colonna precedente la colonna occupata dalla cella che la precede);

Un esempio di percorso su una matrice M è mostrato in figura. Le celle che costituiscono il percorso sono mostrate in grigio.

e_0									
	e_1								
	e_2								
		e_3							
	e_4								
		e_5							
			e_6						
				e_7					
					e_8				
				e_9					

Un percorso può essere descritto mediante un array A di dimensione $m-1$ i cui elementi assumono uno dei tre valori -1, 0, 1. L'elemento i -esimo dell'array indica la "posizione" della cella e_{i+1} rispetto alla cella $e_i=a_{ij}$. Se $A[i]=-1$ allora $e_{i+1}=a_{i+1,j-1}$, se $A[i]=0$ allora $e_{i+1}=a_{i+1,j}$, se $A[i]=+1$ allora $e_{i+1}=a_{i+1,j+1}$. Ad esempio, il percorso illustrato in figura può essere descritto mediante il seguente array:

+1	0	+1	-1	+1	+1	+1	+1	-1
----	---	----	----	----	----	----	----	----

Ogni oggetto della classe **Matrice** rappresenta una matrice di interi positivi di dimensione $m \times n$. La classe ha il seguente scheletro:

```
class Matrice {
```

```
    private int[][];
```

```
    /* crea un oggetto Matrice il cui contenuto è rappresentato dalla matrice m */
```

```
    public Matrice(int[][] m){...}
```

```
    /* Restituisce la somma degli elementi appartenenti al percorso descritto dall'array a. Se la  
    dimensione dell'array a o i suoi valori non sono corretti, il metodo restituisce -1 */
```

```
    public int valorePercorso(int[] a){...}
```

```
    /* restituisce un array che descrive il percorso della matrice costruito secondo la seguente regola:  
    per ogni cella del percorso, la cella successiva è quella di valore minimo tra le tre possibili */
```

```
    public int[] trovaPercorso(){...}
```

```
    /* Restituisce una rappresentazione testuale della matrice */
```

```
    public String toString(){...}
```

```
}
```

Si scriva la classe **Matrice** ed una classe **ProvaMatrice** che contiene il solo metodo **main** e che esegue le seguenti azioni:

- legge una matrice di interi dall'utente;
- crea un oggetto **Matrice m** il cui contenuto è la matrice creata e lo stampa;
- legge un percorso dall'utente (descritto come una sequenza di -1, 0 e 1) e stampa il valore del percorso dato;
- utilizzando il metodo **trovaPercorso()** ottiene il percorso nella matrice costruito secondo la regola descritta in precedenza;
- stampa il percorso ottenuto.

Note:

1. Tutti i file creati devono essere salvati nella cartella C:\fi0114 del proprio PC
2. La cartella C:\fi0114 contiene le classi **InputWindow** e **OutputWindow**.
3. Indicare il proprio cognome e nome, la propria matricola e il numero del PC su cui si sta lavorando negli appositi spazi in cima a questo foglio ed anche come commento in testa alla definizione delle classi che si scrivono su file.