

PROVA SCRITTA DI INFORMATICA
ESAME DI AMMISSIONE
SETTEMBRE 2013

SCUOLA SUPERIORE
UNIVERSITÀ DEGLI STUDI DI UDINE

Esercizio 1. Strade a senso unico. Una città di dimensione n è percorsa da n strade in senso longitudinale e n strade in senso latitudinaria (in Figura 1 a sinistra è rappresentato il caso $n = 3$). Le strade sono molto strette, quindi il comune ha deciso che ogni tratto di strada compreso tra due incroci può essere percorso in un solo verso (senso unico). In Figura 1 a destra è mostrato un possibile assegnamento di sensi unici nel caso $n = 3$. Un assegnamento di sensi unici è *senzato* se da ogni incrocio è possibile raggiungere ogni altro incrocio, rispettando i sensi unici.

Dato un assegnamento di sensi unici per una città di dimensione n , si vuole verificare se questo è senzato. Si elabori un metodo per risolvere il problema proposto.

Data una città di dimensione n si vuole produrre un assegnamento di sensi unici senzato. Si elabori un metodo per risolvere il problema proposto. Si valuti la bontà della soluzione proposta in base alla lunghezza dei percorsi.

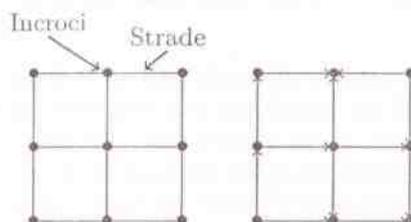


FIGURE 1. Caso $n = 3$

Esercizio 2. Tiling della retta. Si considerino delle tessere simili a quelle del domino. Ogni tessera, oltre ad essere etichettata con due numeri compresi tra 0 e 6, ha una freccia che indica il verso in cui può essere usata. Quindi per semplicità possiamo rappresentare ogni tessera tramite una coppia ordinata (x, y) di numeri compresi tra 0 e 6. Vorremmo disporre le tessere lungo una retta (infinita) modo da ricoprirla. Due tessere vicine devono riportare lo stesso numero sui lati adiacenti. Per esempio, una tessera di tipo $(1, 2)$ deve essere preceduta da una tessera di tipo $(x, 1)$ e seguita da una tessera di tipo $(2, y)$, con x e y qualsiasi. Purtroppo non abbiamo a disposizione tutti i tipi di tessere, ma solo un insieme S di tipi di tessere. Per ogni tipo di tessera dell'insieme S abbiamo a disposizione infinite tessere.

Dato S si vuole decidere se è possibile ricoprire la retta usando solo tessere dei tipi presenti in S . Per esempio, se $S = \{(1, 2), (2, 3)\}$, non è possibile ricoprire la

retta, mentre se $S = \{(1, 2), (2, 1)\}$, è possibile farlo. Si elabori un metodo che dato S permetta di decidere se è possibile ricoprire la retta.

Un ricoprimento della retta è detto *periodico* se è costituito da una sequenza finita di tessere ripetuta infinite volte. È detto *aperiodico* se non è periodico. Esistono ricoprimenti aperiodici? In caso di risposta negativa si fornisca una dimostrazione della non esistenza, mentre in caso di risposta positiva si mostri un esempio di ricoprimento aperiodico e si caratterizzino gli insiemi S che consentono la costruzione di ricoprimenti aperiodici.

Esercizio 3. Noci di Cocco. Tre marinai approdano su un'isola e trovano un mucchio di noci di cocco e una scimmia. Il primo marinaio prende metà delle noci più mezza noce e se ne va. Il secondo marinaio prende metà delle noci rimaste più mezza noce e se ne va. Il terzo marinaio prende metà delle noci rimaste più mezza noce e lascia l'ultima noce rimasta alla scimmia. Quante noci di cocco c'erano all'inizio?

Si risolva il problema generalizzato al caso in cui approdino sull'isola k marinai e alla scimmia vengano lasciate h noci.

Esercizio 4. Il Tesoro dei Pirati. Cinque pirati, A , B , C , D , ed E devono dividersi 100 monete d'oro. A è il pirata più importante dei cinque, quindi proporrà una spartizione. Tutti i pirati (compreso A) voteranno a favore o contro la spartizione proposta da A . Se la maggioranza voterà a favore, il tesoro verrà diviso come stabilito, altrimenti A verrà ucciso e il comando passerà a B che procederà in modo analogo. Se anche B dovesse venire ucciso si passerà a C e così via. Nelle votazioni con un numero pari di votanti, in caso di parità, si procede con la spartizione delle monete.

Quale spartizione deve proporre A per essere sicuro di non morire e massimizzare il proprio guadagno?

Indicazioni Generali. La descrizione dei metodi va fatta spiegando prima di tutto a parole le idee di base e poi fornendo una descrizione più formale degli stessi. Tale descrizione può essere fatta nel formalismo che si ritiene più opportuno. È possibile utilizzare un linguaggio di programmazione standard. Nel presentare i programmi si possono tralasciare dettagli non centrali, quali l'acquisizione dei dati, la stampa dei risultati, il controllo della consistenza dei dati in ingresso. Si raccomanda comunque di commentare il codice proposto.