

Ventitreesima Gara di Informatica per studenti delle Scuole Superiori

Esercizi di gara

AVVISI:

- Se non specificato altrimenti negli esercizi, le sequenze iniziali su nastro si intendono *non vuote*, ovvero contenenti almeno un simbolo.
- Per numero decimale si intende un numero positivo o nullo rappresentato con le cifre 0, 1, 2, ..., 9, senza zeri iniziali non significativi; per esempio 0 e 19 sono numeri validi, mentre 0032 deve essere scritto come 32.
- Nel fornire le soluzioni, ricordarsi di pulire il nastro finale da ogni simbolo che non costituisca la risposta!
- Ogni volta che si salva la soluzione di un esercizio con il simulatore della macchina di Turing, il “timestamp” dell’esercizio viene aggiornato con il tempo trascorso fino a quel momento.

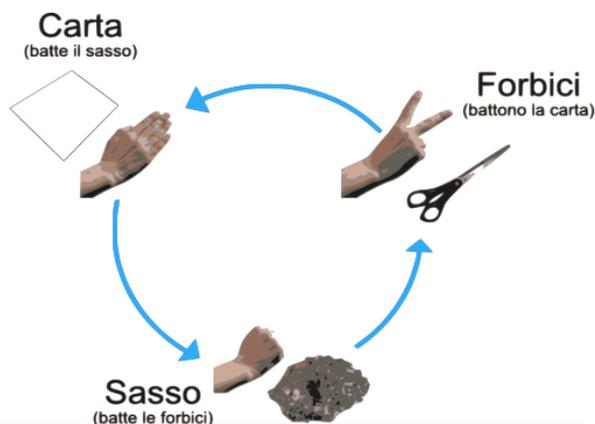
Esercizio 1: Tiro alla fune [Punti 1]. È una bella giornata di primavera! Cosa c’è di meglio di qualche gioco all’aperto in compagnia? Cominciamo con una gara di tiro alla fune per far divertire i più piccolini. I bimbi si dividono in due squadre, A e B; siccome sono un po’ indisciplinati, non è detto che le due squadre abbiano lo stesso numero di membri, ma ogni squadra avrà almeno un membro. Codifichiamo una sfida di Tiro alla fune usando il simbolo A per i giocatori della squadra a sinistra, B per quelli della squadra a destra, e = per la fune (che può essere lunga a piacere). Si scriva un programma per macchina di Turing che, ricevuta sul nastro di input la codifica di una sfida, lasci sul nastro il simbolo della squadra che vince la sfida, assumendo che ciascun bimbo tiri con la stessa “forza” di tutti gli altri. In caso di parità, il programma deve lasciare sul nastro il simbolo “=”.

<i>NASTRO INIZIALE</i>	<i>NASTRO FINALE</i>
AAAA===BB	A
AAAA=BBBBBBBBBB	B
A=====B	=
AAAA===BBBBBB	B

Esercizio 2: Pari o dispari [Punti 2]. Mentre i bimbi giocano al tiro alla fune, due amici un po’ più grandicelli si sfidano in una gara di pari o dispari. In questo gioco, uno dei giocatori pronostica “Pari!”, l’altro risponde “Dispari!”, e poi in maniera sincronizzata entrambi i giocatori aprono la mano mostrando il numero prescelto, fra 0 (pugno chiuso) e 5 (tutte le dita aperte). Vince chi ha pronosticato correttamente se la somma dei due numeri sia pari o dispari. Si scriva un programma per macchina di Turing che, ricevuti sul nastro di input i due numeri prescelti dai giocatori, lasci sul nastro il simbolo “P” se la somma è pari, oppure “D” se è dispari.

<i>NASTRO INIZIALE</i>	<i>NASTRO FINALE</i>
34	D
55	P
04	P
14	D
00	P

Esercizio 3: La morra cinese [Punti 2]. Dopo un po' il pari o dispari viene a noia. Per cambiare, i nostri giocatori passano alla morra cinese: in questo gioco, i due giocatori scelgono e mostrano in contemporanea, con la posizione delle dita, uno di tre simboli: Forbici, Carta o Sasso, come mostrato nella figura a lato. Le forbici tagliano la carta; la carta avvolge il sasso; il sasso rompe le forbici. Quando i simboli dei due giocatori (che codificheremo con F, C, S) sono diversi fra di loro, uno dei due necessariamente batte l'altro; se invece entrambi i giocatori scelgono lo stesso simbolo, si ha un pareggio. Si scriva un programma per macchina di Turing che, ricevuta sul nastro una coppia di simboli dall'alfabeto {F, C, S}, lasci sul nastro il simbolo vincitore, oppure il simbolo "=" in caso di un pareggio.



<i>NASTRO INIZIALE</i>	<i>NASTRO FINALE</i>
CF	F
CS	C
SS	=
CC	=
FC	F

Esercizio 4: Il tiro alla fune dei grandi [Punti 5]. I bambini hanno finalmente mollato la fune per passare ad altri giochi. È arrivato il momento di una sfida fra grandi! In questo caso, le due squadre saranno composte dallo stesso numero di giocatori; i giocatori hanno però livelli di forza assai differenti. Rappresentiamo ogni giocatore con un numero fra 0 e 9, che corrisponde alla forza con cui quel giocatore tira la fune; la fune stessa viene rappresentata con "=", come nell'esercizio 1. La forza complessiva di una squadra è data dalla somma delle forze dei singoli giocatori. Si scriva un programma per macchina di Turing che, ricevuta sul nastro di input la codifica di una sfida, lasci sul nastro il simbolo "A" se risulta vincitrice la squadra di sinistra, "B" se vince quella di destra, o "=" in caso di parità.

<i>NASTRO INIZIALE</i>	<i>NASTRO FINALE</i>
3842=3366	B
3842=3365	=
8===6	A
40=====17	B

Esercizio 5: Il salto della cavallina [Punti 3]. I bimbettini scalmanati amano i giochi di movimento. Uno particolarmente popolare è il salto della cavallina. Un giocatore prende il ruolo della cavallina (lo indicheremo con C), mentre altri giocatori si pongono accovacciati davanti a lui, a una qualche distanza, e fungono da ostacoli (li indicheremo con #). La cavallina può correre lungo il percorso libero (che indicheremo con .) e saltare un singolo ostacolo, purché dopo ci sia un tratto di percorso libero. Si arresta invece di fronte a due ostacoli consecutivi, oppure alla fine del percorso libero. Si scriva un programma per macchina di Turing che, ricevuta sul nastro di input la codifica di un percorso, con la cavallina all'estremità sinistra e in moto verso destra, lasci sul nastro la situazione del percorso al momento dell'arresto della cavallina.

<i>NASTRO INIZIALE</i>	<i>NASTRO FINALE</i>
C.#.#.#.##	.#.#.#C##
C...#...###...#	...#.C####...#
C#...#...#.....#	#...#...#.....C#
C...#...	...#...C

Esercizio 6: Il torneo di morra cinese [Punti 8]. La morra cinese dell'esercizio 3 si rivela più appassionante del previsto. Si stabilisce che una partita vinta valga 1 punto per il vincitore e 0 per il perdente, mentre un pareggio vale mezzo punto ciascuno.

Codifichiamo un torneo di n partite indicando sul nastro gli n simboli scelti dal primo giocatore, seguiti da una virgola, e poi gli n simboli del secondo giocatore, intendendo che il primo simbolo del primo giocatore si scontri col primo simbolo del secondo giocatore, e così via. Si scriva un programma per macchina di Turing che, ricevuta sul nastro di input la codifica di un torneo, lasci sul nastro il punteggio del primo giocatore, seguito da una virgola, e da quello del secondo giocatore. Si usi la notazione anglosassone con il punto decimale per separare parte intera e parte frazionaria dei punteggi.

<i>NASTRO INIZIALE</i>	<i>NASTRO FINALE</i>
CCFSS, FCCSC	2, 3
CCF, CSC	2.5, 0.5
C, C	0.5, 0.5
F, C	1, 0
CCCCCCCC, SFCFFSCC	3.5, 4.5

Esercizio 7: Beach volley fra amici [Punti 12]. Un po' gioco, un po' sport, una bella partita a beach volley è sempre un ottimo modo per trascorrere un po' di tempo con gli amici. Per semplicità, giocheremo alla *beach volley lineare*; il campo è composto da 8 caselle per parte, con il fondo di sabbia rappresentato dal simbolo S, e con la rete al centro rappresentata dal simbolo #; la rete è conteggiata come una casella, quindi l'intero campo è ampio 17 caselle. Lungo il campo sono disposti i giocatori, in numero variabile (è un gioco un po' caotico); ogni giocatore è rappresentato da una cifra fra 1 e 9, che indica quanto lontano (in caselle) quel giocatore, se riceve la palla, la tirerà verso il campo avversario. Inizialmente la palla è in mano al giocatore più esterno della squadra che "batte", che si trova sempre sulla linea di fondo campo e quindi nella casella più esterna. Se la palla tocca terra nel campo di una squadra, la squadra opposta vince il punto. Se la palla finisce oltre la linea di fondo del campo di una squadra, quella squadra vince il punto. Se una qualunque squadra fa più di tre tocchi, la squadra opposta vince il punto. Se la palla atterra sulla rete, il punto si considera pareggiato e nessuna delle due squadre vince il punto. Si scriva un programma per macchina di Turing che, data una configurazione del campo, lasci sul nastro il punteggio della partita dopo che è stata eseguita una battuta da ciascuna delle due squadre, nella notazione $x-y$ dove x e y siano il punteggio (che sarà un intero fra 0 e 2) della squadra a sinistra e a destra della rete, rispettivamente.

<i>NASTRO INIZIALE</i>	<i>NASTRO FINALE</i>
4sss16ss#ss5ss8s2	0-2
3sss162s#ss5ss8s2	0-1
5sss163s#ss8ss3s2	0-2
9s21113s#ss8ss3s2	1-1

Esercizio 8: Caccia al tesoro [Punti 15]. Per far divertire un po' i bimbi prima che cali il sole, organizziamo una caccia al tesoro! L'area in cui cercare è disseminata di indizi, che il giocatore deve identificare e rispettare alla lettera. Anzi, alle lettere, secondo la corrispondenza seguente:

A	Vai tre caselle a destra
B	Vai quattro caselle a sinistra
C	Vai tre caselle a destra; se trovi una C, vai ancora una casella a destra
D	Vai a sinistra di due caselle; se trovi una A, vai una casella a destra
E	Vai a destra finché non trovi una B o una C, e segui l'indizio che hai trovato
F	Cambia la F in B, e torna all'inizio del percorso
G	Vai alla fine del percorso, segui l'eventuale indizio che trovi sull'ultima casella
T	Il tesoro!
X	Posizione di partenza, vai a destra finché non trovi un indizio
.	Casella vuota: nessun indizio

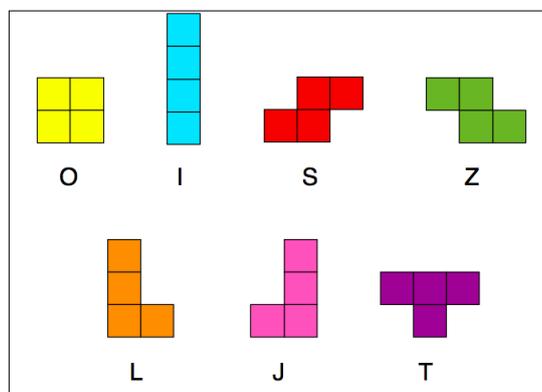
Scopo della caccia è trovare il tesoro (T). Se la ricerca si ferma su una casella vuota, oppure esce dal campo di gioco, la ricerca si considera fallita. Si scriva un programma per macchina di Turing che, ricevuta sul nastro di input la codifica di un terreno di gioco con i relativi indizi, lasci sul nastro la posizione dall'inizio del percorso in cui si trovava il tesoro (espressa come un numero decimale, in cui il primo simbolo si conta come posizione 0), se la ricerca si è conclusa trovando il tesoro, oppure la situazione del gioco se ci si è fermati senza trovare il tesoro, in cui la posizione finale del giocatore è marcata con una Y.

NASTRO INIZIALE	NASTRO FINALE
X...AADAG.B...T...BBD	15
X.....F	X...Y...B
X....T...D	5
AAAX..B	AAAX.YB
..X.....	..X.....Y

Esercizio 9: Buca in uno! [Punti 24]. A differenza della pallavolo, il golf si gioca bene anche da soli -- che a volte è proprio quello che ci vuole per rilassarsi. Supponiamo di essere a distanza di d metri dalla buca, e di lanciare la pallina, in direzione della buca, con un angolo di esattamente 45° rispetto all'orizzonte e con velocità v metri al secondo. Si scriva un programma per macchina di Turing che, ricevuto sul nastro di input il valore d seguito da uno spazio e quindi dal valore v , lasci sul nastro la distanza fra il punto in cui la pallina atterrerà e la buca. Ovviamente, non siamo in Kansas, ma su un pianeta privo di atmosfera, leggermente più grosso della Terra (l'accelerazione di gravità g qui è di esattamente 10 m/s^2), con terreno parecchio appiccicoso (la pallina si arresta immediatamente al contatto col terreno). Tutti i calcoli possano essere fatti approssimando a una cifra dopo la virgola.

NASTRO INIZIALE	NASTRO FINALE
20 7	15.1
10 4	8.4
7 12	7.4
33 16	7.4
10 10	0

Esercizio 10: Tetris [Punti 35]. Ormai è sera; tutti a casa! Il momento ideale per giocare a qualche videogioco. Nel famoso videogioco *Tetris*, un pezzo di forma irregolare (un *tetramino*, formato da quattro blocchetti) scelto fra quelli indicati nella figura accanto, e con una orientazione casuale ma parallela agli assi cartesiani, viene lasciato cadere all'interno di un contenitore rettangolare su una pila di altri pezzi analoghi. Il pezzo si ferma quando tocca un blocchetto già presente, oppure la base del contenitore. Inoltre, se con l'aggiunta del pezzo si vengono a formare una o più righe complete di blocchetti, tutte le righe complete vengono rimosse, e gli eventuali blocchetti soprastanti "cadono" verso il basso, fino a incontrare un ostacolo, in una simulazione di legge di gravità. Nel gioco originale, il giocatore ha la possibilità di intervenire durante la caduta del pezzo, sia ruotandolo di 90° in qualunque direzione, sia spostandolo di una posizione a destra o sinistra. Nella nostra versione invece assumeremo che il giocatore abbia già fatto le sue mosse, e simuleremo soltanto la caduta del pezzo (con i suoi effetti). Codificheremo il contenitore con la lettera A per la parete sinistra, B per il fondo, C per la destra (l'alto è aperto); tutte le celle libere all'interno del contenitore saranno indicate dal carattere



A	.	.	T	T	T	.	.	.	C				A	C
A	.	.	.	T	C				A	C
A	C				A	C
A	C				A	C
A	C				A	C
A	C				A	C
A	C				A	C
A	C				A	C
A	C				A	C
A	C				A	.	.	T	T	T	.	.	C
A	C				A	.	.	.	T	.	.	.	C
A	B	B	B	B	B	B	B	B	C				A	B	B	B	B	B	B	B	C

A	C				A	C	
A	C				A	C	
A	C				A	C	
A	C				A	C	
A	C				A	C	
A	C				A	C	
A	T	.	C				A	C	
A	T	T	.	C				A	C	
A	.	.	I	.	.	.	T	I	C				A	C	
A	.	.	I	I	C				A	.	.	I	.	.	.	I	C	
A	O	O	I	Z	Z	.	.	I	C				A	.	.	I	.	.	T	I	C	
A	O	O	I	.	Z	Z	.	I	C				A	O	O	I	.	Z	Z	T	I	C
A	B	B	B	B	B	B	B	B	C				A	B	B	B	B	B	B	B	C	

Si scriva un programma per macchina di Turing che, ricevuta sul nastro di input la rappresentazione di una situazione iniziale del Tetris (ottenuta concatenando tutte le righe, dall'alto verso il basso - si tratterà di una sequenza di esattamente 130 caratteri sull'alfabeto {A,B,C,O,I,S,Z,L,J,T,}), lasci sul nastro la situazione finale, dopo la caduta e la cancellazione delle righe complete, rappresentata nello stesso modo.

<i>NASTRO INIZIALE</i>	<i>NASTRO FINALE</i>
<i>Vedi esempi illustrativi</i>	<i>Vedi esempi illustrativi</i>