

TestiGara - DiciannovesimaEdizione

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: Inaugurazione. [Punti 1] Nella piccola cittadina tedesca di Maschinestadt, in Turingia, sta per aprire un nuovo circolo culturale. Il proprietario, Alan, ha invitato una serie di abitanti all'inaugurazione, e ha segnato su una lista una “S” per ogni invito accettato, e una “N” per ogni invito rifiutato. Si scriva un programma per macchina di Turing che, ricevuta sul nastro di input una sequenza di simboli “S” e “N”, lasci sul nastro una sequenza di tante “S” quanti sono le “S” nella sequenza di input (la lunghezza della risposta corrisponderà al numero di persone che hanno accettato l'invito all'inaugurazione).

NASTRO INIZIALE NASTRO FINALE

SNSSNNSSS SSSSSS

NS S

SSS SSS

NNNNNN

Esercizio 2: Una misura di successo. [Punti 1] Una inaugurazione di circolo a cui partecipano meno di 10 persone è un fallimento. Si scriva un programma per macchina di Turing che, ricevuta sul nastro di input una sequenza di simboli sull'alfabeto {S, N} come nell'esercizio precedente, lasci sul nastro il solo simbolo “S” (per Successo) se la sequenza di input comprendeva almeno 10 “S”, e “F” (per Fallimento) in caso contrario.

NASTRO INIZIALE

SNSNSNNSSSNSSSNSSSNSS

SNNNSNSNNNNNSNNNNNSNNNNNNN

SSSSSS

NNNN

NASTRO FINALE

S

F

F

Esercizio 3: Più siamo più ci divertiamo.. [Punti 2] Alle feste di Alan partecipano sempre più persone. All'ingresso, viene segnata con una “X” su un nastro ogni persona entrata. A fine serata, si vuole sapere quante persone hanno partecipato alle attività. Si scriva un programma per macchina di Turing che, ricevuta sul nastro di ingresso una sequenza di “X”, lasci sul nastro il numero decimale corrispondente

alla lunghezza della sequenza (ovvero, al numero di persone che sono entrate nella serata). Per questo esercizio, è possibile che la sequenza di input sia vuota (ovvero, nessuno è entrato); in questo caso il programma deve lasciare sul nastro il numero 0.

<i>NASTRO INIZIALE</i>	<i>NASTRO FINALE</i>
XXXXXX	6
XXXXXXXXXXXXXXXXXXXXXXXXXXXX	24
X	1
	0

Esercizio 4: Il giusto mix [Punti 5] Si sa, una festa funziona meglio se c'è un giusto mix di ragazzi e ragazze. Per questo motivo, il lunedì sera il club di Alan offre l'ingresso gratis ai rappresentanti del genere che in quel momento è minoritario. Si scriva un programma per macchina di Turing che, ricevuta sul nastro di input una sequenza di simboli “M” (che indica l'ingresso di un ragazzo) e “F” (che indica l'ingresso di una ragazza), lascia sul nastro un simbolo “F” se deve essere offerto l'ingresso gratis alle ragazze, oppure “M” se deve essere offerto ai ragazzi. In caso di parità, per cavalleria, è gratuito l'ingresso per le ragazze.

<i>NASTRO INIZIALE</i>	<i>NASTRO FINALE</i>
MFMMMFMMFMMM	F
FFFFFFF	M
MFFFMFFFFMFFFFMM	M
FFFMFMFFF	F
M	F

Esercizio 5: Ingressi e uscite. [Punti 8] Il martedì sera il circolo di Alan è ad ingresso libero: si svolgono dibattiti culturali sul tema dell'Entscheidungsproblem. Per qualche motivo, questi sono meno popolari delle feste del lunedì; entrano tante persone, ma molte escono (qualcuna, visibilmente pallida in volto) dopo pochi minuti. Alan chiede al personale all'ingresso di segnare con “M” e “F” gli ingressi, come nell'esercizio precedente, e con “N” e “E” le rispettive uscite. Si scriva un programma per macchina di Turing che, ricevuta sul nastro di ingresso una sequenza di simboli in {M, F, N, E} (con la garanzia che per ogni genere le uscite non possono superare gli ingressi), lasci sul nastro come risultato la sequenza “MkFh” in cui k e h sono espressi come numeri decimali, e indicano quante persone per il corrispondente genere sono ancora dentro il circolo nel momento in cui si analizza la sequenza.

<i>NASTRO INIZIALE</i>	<i>NASTRO FINALE</i>
MMFMFFNNEFMNEEF	M2F2
MMMNNN	M0F0
MMFMFMFFMENFMMFFFNMFNNNF	M3F10
FFFF	M0F4
MFMFMFMMMMMMFMME	M13F4

Esercizio 6: Posti a tavola. [Punti 15] Il mercoledì sera il circolo organizza una cena fra i soci, che siedono tutti a una grande tavola circolare (viene usato un nastro infinito per trasportare le vivande, sul modello dei ristoranti Kaiten-zushi). Come al solito, la serata è più gradevole se ogni commensale uomo ha almeno un membro del gentil sesso accanto, e viceversa. Ovviamente, non sempre questo è possibile: dipende dal numero dei commensali. Si scriva un programma per macchina di Turing che, ricevuta sul nastro di input una sequenza di “M” e “F” che rappresentano gli ingressi come negli esercizi precedenti, lasci sul nastro la risposta “OK” se esiste una disposizione dei commensali al tavolo circolare che garantisce la proprietà di gradevolezza di cui sopra, e “KO” in caso contrario.

NASTRO INIZIALE NASTRO FINALE

MFMFMFMF OK

MFMFMM OK

FFFFFFM KO

MMFMMFMMFFM OK

MMMMMM KO

Esercizio 7: Bingo binario. [Punti 17] Il giovedì sera è dedicato agli anziani, e Alan organizza per loro un Bingo un po' semplificato. Ogni giocatore ha una cartella, composta da 4×4 caselle; in ogni casella è presente un simbolo “0” o “1”. Viene poi estratta una sequenza di 4 simboli fra “0” e “1”. Il giocatore che ha nella propria cartella l'esatta sequenza estratta, in orizzontale (da sinistra a destra), verticale (dall'alto al basso) o diagonale (da alto-sinistra a basso-destra) vince il premio per quella mano. Si scriva un programma per macchina di Turing che, ricevuta sul nastro di ingresso una codifica della cartella (in cui le 4 righe di simboli sono riportate in sequenza, da sinistra a destra, iniziando dalla riga più in alto), seguita da un simbolo “#” e poi dalla sequenza di 4 simboli estratta; lasci sul nastro “SI” se la cartella è vincente, “NO” in caso contrario.

NASTRO INIZIALE NASTRO FINALE

0100010111000101#1111 SI

0100010111000101#0101 SI

0100010111000101#0011 NO

1111111111111111#1111 SI

1111111111111111#1011 NO

Esercizio 8: Cinema, cinema! [Punti 21] Il venerdì sera è giorno di cineforum presso il club di Alan. Il circolo è senza scopo di lucro, per cui il costo del biglietto viene calcolato dividendo il costo di noleggio del film per il numero di persone che partecipano alla serata. Per semplicità, il costo è sempre un numero intero di euro, approssimando in alto la quota individuale. L'eventuale (piccolo) avanzo viene destinato a fondo cassa per altre iniziative. Inoltre, il costo del biglietto è sempre inferiore a €10: se il numero di partecipanti non è sufficiente a garantire questa condizione, la serata viene annullata (e non si paga il noleggio). Si scriva un programma per macchina di Turing che, ricevuto sul nastro di input il costo di noleggio espresso in euro, seguito da un simbolo “/” e poi dal numero di partecipanti, lasci sul nastro il costo del biglietto seguito dal simbolo “%” e dal residuo di fondo cassa, oppure “NO” se la serata va annullata per via del costo del biglietto troppo alto.

NASTRO INIZIALE NASTRO FINALE

140/20 7%0

150/20 8%10

150/5 NO

315/68 5%25

Esercizio 9: Il Social Network [Punti 30] Per pubblicizzare le feste danzanti del sabato sera, Alan ricorre ai social network. Il problema consiste nell'identificare le persone che hanno maggiore influenza (intesa come: numero di amici e di amici degli amici), e fornire loro un invito gratis a condizione che costoro pubblicizzino l'evento con un post sulla loro bacheca. Diamo una rappresentazione del social network in cui gli abitanti dotati di presenza online sono indicati con una lettera dell'alfabeto fra “A” e “Z” (Maschinestadt è una città molto piccola). Gli amici di una persona sono indicati, fra parentesi, subito dopo il nome della persona. Così, per esempio, A(BF)B(CDEF)C(AEF) indica che A è amico di B e F, e B è amico di C, D, E e F, C è amico di A, E e F (si noti che la relazione di amicizia su questo social network non è simmetrica: A è amico di B, ma B non è amico di A). Nel nostro esempio, A ha 2 amici (B,F); di questi, B ha 4 amici (C,D,E,F), e F nessuno; complessivamente quindi la cerchia (ovvero: amici e amici

degli amici, escludendo la persona in questione) di A comprende (B,C,D,E,F) e la sua influenza è pari a 5. Analogamente, B ha 4 amici (C,D,E,F); di questi, C ha 3 amici (A,E,F), e gli altri nessuno. La cerchia di B comprende (A,C,D,E,F) e la sua influenza è quindi pari a 5. Quanto a C, la sua cerchia comprende (A,B,E,F) e la sua influenza è dunque 4. Si scriva un programma per macchina di Turing che, data in input la descrizione di un social network nel formato indicato sopra, lasci sul nastro il nome della persona di massima influenza fra quelle presenti nel social network. In caso di parità, il programma deve lasciare su nastro il nome della persona fra gli ex-æquo che compare per prima nella descrizione in input.

<i>NASTRO INIZIALE</i>	<i>NASTRO FINALE</i>
A(BF)B(CDEF)C(AEF)	A
A(CFG)B(CDEF)C(F)D(AB)G(A)	D
A(B)B(A)D(E)E(F)F(GHI)	E
L(Z)C(AD)D(AB)E(AB)A(FGH)M(TUVXYZ) C	C
Z(ABC)A(BC)B(CDE)Y(Z)	Z

Esercizio 10: Reazione e diffusione. [Punti 25] La domenica il circolo resta chiuso, e Alan può dedicarsi a un altro dei suoi interessi: la modellazione dei fenomeni biologici tramite equazioni di reazione-diffusione. Nella loro forma più generale, le equazioni di reazione-diffusione possono essere scritte come $\partial u / \partial t - d \nabla^2 u = f(u, x, t)$ dove d è il coefficiente di diffusione e $f(u, x, t)$ rappresenta il termine non omogeneo di reazione. Queste equazioni possono modellare, fra l'altro, il modo in cui due sostanze chimiche semplici possono interagire per creare strutture complesse come le macchie del manto dei leopardi, oppure il modo in cui popolazioni di animali possono crescere, soffrire carestie e diminuire di numero, o spostarsi verso territori ancora vergini. Noi ci concentreremo però su un caso più semplice, quello di una reazione-diffusione di due sostanze (che chiameremo U e V) in una sola dimensione: si pensi, come caso concreto, a due sostanze chimiche che regolano la colorazione di un singolo pelo. Modelliamo la struttura unidimensionale come una serie di loci, rappresentati come coppie di celle, in cui la prima contiene la concentrazione di U, espressa con un numero decimale fra 0 (assente) e 9 (massima concentrazione), e la seconda contiene la concentrazione di V, espressa con una lettera fra A (assente) e J (massima concentrazione). Si applicano ad ogni locus le seguenti due regole:

1. **Diffusione:** se la concentrazione di una sostanza in un locus è maggiore di quella nel locus successivo, si decrementa di uno la concentrazione nel primo, e si aumenta di uno nel secondo. Questa regola non si applica all'ultimo locus (quello più a destra) della struttura.
2. **Reazione:** se in un locus si trova una concentrazione di almeno 6U e non più di 3V, allora vengono consumati 3U e trasformati in 1V (ovvero: si decrementa la concentrazione di U di 3, e si aumenta quella di V di 1).

Si scriva un programma per macchina di Turing che, data sul nastro di ingresso la configurazione iniziale di una struttura, ne calcoli ripetutamente l'evoluzione, arrestandosi soltanto quando l'applicazione delle due regole sopra dette non altera la concentrazione delle sostanze già realizzata (in questo caso, si è raggiunta la configurazione finale della struttura). Esempio: Partendo dalla prima configurazione, l'applicazione delle due regole di cui sopra ad ogni passo produce la seguente sequenza di configurazioni:

5E 8A 5E 2A → 5D 4C 5D 3B → 4C 5D 4C 4C → 4C 4C 5D 4C → 4C 4C 4C 5D

L'ultima configurazione è finale, in quanto non si applica nessuna delle due regole.

<i>NASTRO INIZIALE</i>	<i>NASTRO FINALE</i>
5E8A5E2A	4C4C4C5D
7A4D9J2C3A3D8I2E8A9I	3B4D4D4D5E5E5E5E5F9I
9A9A9A9A9A	3C3C3C3C3C

0A8E	0A8E
0J9J2B5B9B8B	0E3E4E4E5F5F
0A8D0E0C2A9E8I7D3A4C0A1C2I	0A1C2C2C2D3D4D4D4D4D4D4I
9A9A0J0J0J	1B2C2J2J2J