

# TestiGara - DiciassettesimaEdizione

Diciassettesima 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: Cambio di casacca. [Punti 1]** Nel Mondo dei Programmatori, la vita politica è molto intensa e piena di sorprese. Nuove liste e partiti si formano e dissolvono di continuo, mentre i Programmatori discutono di quale sia il linguaggio migliore da utilizzare. Per esempio, fino a qualche anno fa si contavano vari movimenti: i Programmatori C Indipendenti (simbolo: c), i Discreti & Continui (simbolo: d), i Programmatori Strutturalisti (simbolo: s) e i Programmatori Lisp Integralisti (simbolo: l). Negli anni, questi raggruppamenti hanno cambiato molte volte nome, al punto che si è preferito numerarli per chiarezza. Così, i Programmatori C Indipendenti sono diventati il partito 0, i Discreti & Continui sono diventati il partito 1, i Programmatori Strutturalisti sono diventati il partito 2, e i Programmatori Lisp Integralisti il partito 3. Si scriva un programma per macchina di Turing che, ricevuta sul nastro una stringa sull'alfabeto {c, d, s, l}, in cui ogni simbolo rappresenta un elettore del corrispondente partito, lasci sul nastro gli stessi elettori, codificati però sull'alfabeto {0, 1, 2, 3}.

*NASTRO INIZIALE NASTRO FINALE*

dccsdlcdd	10021011
c	0
dddd	1111
lcccl	30003

**Esercizio 2: Il Seggio Ben Ordinato. [Punti 3]** In tempo di votazioni, i Programmatori si mettono ordinatamente in fila al seggio (indicato dal simbolo S). Un seggio è ben ordinato se la fila scorre in una unica direzione man mano che gli elettori votano (ovvero: tutti gli elettori sono, sul nastro, alla sinistra del seggio oppure tutti alla destra del seggio), mentre è mal ordinato in caso contrario. Si scriva un programma per macchina di Turing che, ricevuto sul nastro una stringa sull'alfabeto {0, 1, 2, 3, s} rappresentate la fila degli elettori e il seggio, lasci sul nastro il solo simbolo B se il seggio era ben ordinato, M altrimenti. Il seggio vuoto si considera ben ordinato.

*NASTRO INIZIALE NASTRO FINALE*

130201301123230S	B
S0001	B
12321S3021	M

S                      B  
1S12301202013302 M

**Esercizio 3: Calcolo dell'affluenza. [Punti 5]** Si scriva un programma per macchina di Turing che, ricevuto in ingresso una configurazione di seggio (ben ordinato o meno), lasci sul nastro un numero decimale rappresentante il numero totale di elettori presenti nella configurazione.

*NASTRO INIZIALE    NASTRO FINALE*

S01200213	8
123001002S0123301	16
S	0
1S	1

**Esercizio 4: Caccia ai brogli! [Punti 7]** Nel Mondo dei Programmatori il partito dei Programmatori C Indipendenti (il partito 0) ha fiducia assoluta nella fedeltà del proprio elettorato, e avendo ottenuto alle scorse elezioni meno voti di quelli previsti, sospetta che i partiti avversari abbiano trovato il modo di alterare il conteggio delle schede. Si scriva un programma per macchina di Turing che, ricevuto in ingresso una sequenza di schede, ciascuna indicata con un simbolo fra 0 e 3, seguita dal simbolo ':' e poi dal numero di voti attesi espresso in decimale, lasci sul nastro la scritta OK se la votazione appare regolare, KO se si ritiene di poter contestare il risultato.

*NASTRO INIZIALE                      NASTRO FINALE*

011023001:2	KO
120113:1	OK
12312111:0	OK
0120000:5	OK
12334:12	KO
1022000320100300120030:12	OK

**Esercizio 5: Il sistema maggioritario. [Punti 10]** Nel Mondo dei Programmatori è in vigore il sistema maggioritario: il partito che ottiene più voti, vince (e impone a tutti i programmatori di usare il suo linguaggio preferito, fino alle prossime elezioni). Si scriva un programma per macchina di Turing che, ricevuto sul nastro una configurazione di seggio come negli esercizi precedenti, termini lasciando come risposta sul nastro il simbolo del partito che aveva più elettori nella configurazione. In caso di ex-aequo, il programma deve terminare lasciando sul nastro i simboli dei partiti maggioritari ex-aequo, in ordine numerico crescente. In caso di seggio vuoto, il programma deve terminare lasciando il nastro vuoto.

*NASTRO INIZIALE    NASTRO FINALE*

S	
11111S222	1
1100021321001201S	1
S1122330032	23
321S	123

**Esercizio 6: Il Parlamento dei Programmatori. [Punti 11]** Il Parlamento dei Programmatori si organizza in gruppi che fanno riferimento a uno dei 4 partiti. Come in altri consessi si formano ipotetici raggruppamenti di “destra” e “sinistra”, in base a dove i parlamentari seggono, così nel Parlamento dei Programmatori (gente notoriamente amante dell'ordine) si usa avere i partiti disposti in ordine numerico crescente, con il partito 0 seduto all'estrema sinistra e il partito 3 all'estrema destra. Si scriva un programma per macchina di Turing che, data una sequenza di eletti (ciascuno rappresentato da un simbolo fra 0 e 3), faccia ordine e disponga i membri secondo l'ordine in cui essi siederanno in Parlamento,

lasciando sul nastro la sequenza corrispondente.

NASTRO INIZIALE      NASTRO FINALE

13321001000	00000111233
312212333	112223333
3	3
1230	0123
2221002012012	0000111222222

**Esercizio 7: Il sistema proporzionale. [Punti 21]** Dopo che una imprevista vittoria elettorale del Partito Programmatori Pensionati aveva obbligato tutti a scrivere programmi solo in COBOL e FORTRAN, si è deciso di passare al sistema proporzionale. In questo sistema, vengono eletti quattro rappresentanti, distribuiti in maniera proporzionale rispetto al numero di elettori. In particolare, ogni partito ottiene un rappresentante ogni 25% del voto raccolto (così, un partito che ottenga il 50% dei voti avrà due rappresentanti); per la parte residua, il rappresentante viene assegnato al partito che ha ottenuto più voti secondo il metodo dei cosiddetti "massimi resti". Si scriva un programma per macchina di Turing che, ricevuto sul nastro una configurazione di seggio, termini lasciando sul nastro i 4 simboli corrispondenti ai 4 rappresentanti eletti, in ordine numerico crescente, secondo il sistema proporzionale. Per semplicità, si assuma che il numero di elettori presenti sia sempre un multiplo di 4, non superiore a 999, e che non si verifichino mai situazioni di ex-aequo.

*Esempio: supponiamo che gli esiti della votazione siano i seguenti: partito 0: 18 voti, partito 1: 12 voti, partito 2: 6 voti, partito 3: 8 voti. Abbiamo un totale di 44 voti espressi, quindi il 25% corrisponde a 11 voti. Il partito 0 ottiene subito un rappresentante (poiché  $11 \leq 18 < 22$ ), come anche il partito 1; i partiti 2 e 3 non ne ottengono alcuno. I voti corrispondenti ai rappresentanti assegnati vengono quindi sottratti al monte-voti di ciascun partito; questo ci lascia con i seguenti "resti": partito 0: 7 voti, partito 1: 1 voto, partito 2: 6 voti, partito 3: 8 voti. I due rappresentanti mancanti vengono quindi assegnati al partito 3 (che ha un resto di 8 voti) e al partito 0 (che ha un resto di 7 voti), mentre il partito 2 con 6 voti di resto e il partito 1 con 1 voto di resto non ottengono altro. L'esito dell'elezione è dunque: 0013.*

NASTRO INIZIALE      NASTRO FINALE

1111S	1111
01230S3320110	0123
S01230123012301230123	0123
21101S1131112	1112

**Esercizio 8: Rimborsi elettorali [Punti 19]** Nel Mondo dei Programmatori, i diversi partiti ricevono un rimborso fisso per le spese elettorali, che è sempre un multiplo di 12, che va poi diviso fra i rappresentanti eletti. Si scriva un programma per macchina di Turing che, ricevuto sul nastro l'esito di una elezione (ovvero, una sequenza di 4 simboli sull'alfabeto {1, 2, 3, 4}, come dato dall'esercizio precedente), seguito da un ":", dal simbolo del partito per cui stiamo facendo i conti, un altro ":", e infine la cifra del rimborso attribuito al partito in questione, lasci sul nastro il rimborso pro-quota per ciascuno eletto di quel partito. Si assuma che il rimborso viene attribuito solo se c'è almeno un eletto del partito in questione.

NASTRO INIZIALE      NASTRO FINALE

1111:1:12000	3000
1123:1:12000	6000
0123:1:12000	12000
2233:2:144	72
2333:3:1440	480

**Esercizio 9: Al Ballottaggio! [Punti 18]** Anche il sistema proporzionale ha i suoi difetti. Alle ultime

elezioni, è stato eletto esattamente un rappresentante per ciascun partito, e ovviamente i quattro non si sono messi d'accordo su quale linguaggio adottare. Si decide allora di passare al sistema di voto con ballottaggio. In questo sistema, si svolgono due turni di votazione. Se al primo turno uno dei partiti ottiene più del 50% dei voti, l'elezione termina con la vittoria di quel partito. Altrimenti, vengono riproposti al voto, nel secondo turno, i due partiti che hanno ottenuto il maggior numero di voti al primo turno. In tutti i casi di parità, prevale il partito che ha per simbolo il valore numerico più alto. Si scriva un programma per macchina di Turing che, ricevuta in input una configurazione di seggio, termini lasciando sul nastro il simbolo del partito vincitore, oppure i simboli dei due partiti che andranno al ballottaggio, in ordine numerico crescente.

<i>NASTRO INIZIALE</i>	<i>NASTRO FINALE</i>
1100021321001201S	01
31010023021S321001201	01
S22022121222322	2
11100332S	13
31113S332201	13

**Esercizio 10: Politica 2.0. [Punti 30]** Tradizionalmente, i programmatori sono persone asociali, che non parlano facilmente di politica fra di loro. Tuttavia, quest'anno va di gran moda un nuovo *social network*, chiamato Votebook, che consente a un programmatore in fila al seggio di scambiare opinioni, passando per dei server in Antartide, con le persone davanti a lui e con quelle dietro di lui nella fila. Questi scambi di opinioni possono ovviamente influenzare il voto. In particolare, se fra le due persone davanti e le due dietro all'elettore, ce ne sono almeno tre dello stesso partito, l'elettore viene convinto, e si adegua alla loro opinione (influenzando così l'esito finale della votazione).

Questo processo può essere espresso per passi come segue. Si prenda una configurazione di seggio, anche mal ordinata. Ad ogni passo, gli elettori immediatamente accanto al seggio esprimono il loro voto (che viene conteggiato), e lasciano la fila. Tutti gli altri elettori hanno uno scambio di opinioni con gli elettori immediatamente precedenti e successivi nella fila, ed eventualmente cambiano opinione come descritto sopra. Per l'elettore giunto all'inizio della fila (ovvero, accanto al seggio) si considera che gli elettori precedenti siano i primi della fila opposta, se presenti (altrimenti, il primo elettore non cambia opinione); gli elettori in fondo alla fila non hanno elettori successivi, e quindi non cambiano mai opinione. Una volta terminato lo scambio di opinioni, il passo termina. L'intera votazione si svolge in un numero di passi sufficiente a far votare tutti gli elettori.

Si scriva un programma per macchina di Turing che, presa in input una configurazione di seggio, lasci sul nastro i simboli del partito che ha ottenuto la maggioranza dei voti. Si assuma che non vi saranno casi di ex-aequo.

*Esempio: di seguito indichiamo la configurazione iniziale e i vari passi di una votazione (gli spazi intorno al seggio non sono significativi, e vengono mostrati solo per chiarezza):*

132100101220121 S 12030121100200	voti espressi: 1,1	risultati parziali: 0, 2, 0, 0
13210010122012 S 2030121100200	cambi di opinione	
13210000122022 S 2030111100000	opinioni modificate	
13210000122022 S 2030111100000	voti espressi: 2,2	risultati parziali: 0, 2, 2, 0
1321000012202 S 030111100000	cambio di opinione	
1321000012222 S 030111100000	opinioni modificate	
1321000012222 S 030111100000	voti espressi: 2,0	risultati parziali: 1, 2, 3, 0

132100001222 S 30111100000	nessun cambio di opinione		
132100001222 S <b>3</b> 0111100000	voti espressi: 2,3	risultati parziali: 1, 2, 4, 1	
13210000122 S 0111100000	nessun cambio di opinione		
13210000122 S <b>0</b> 111100000	voti espressi: 2,0	risultati parziali: 2, 2, 5, 1	
13210000 <b>12</b> S <u>1</u> 11100000	cambio di opinione		
13210000 <b>12</b> S <u>1</u> 11100000	opinione modificata		
13210000 <b>12</b> S <b>1</b> 11100000	voti espressi: 2,1	risultati parziali: 2, 3, 6, 1	
132100001 S 11100000	nessun cambio di opinione		
13210000 <b>1</b> S <b>1</b> 1100000	voti espressi: 1,1	risultati parziali: 2, 5, 6, 1	
13210000 S <b>1</b> 100000	cambio di opinione		
13210000 S <b>0</b> 100000	opinione modificata		
13210000 S <b>0</b> 100000	voti espressi: 0,0	risultati parziali: 4, 5, 6, 1	
1321000 S <b>1</b> 00000	cambio di opinione		
1321000 S <b>0</b> 00000	opinione modificata		
1321000 S <b>0</b> 00000	voti espressi: 0,0	risultati parziali: 6, 5, 6, 1	
132100 S 00000	nessun cambio di opinione		
132100 S <b>0</b> 0000	voti espressi: 0,0	risultati parziali: 8, 5, 6, 1	
13210 S 0000	nessun cambio di opinione		
1321 <b>0</b> S <b>0</b> 000	voti espressi: 0,0	risultati parziali: 10, 5, 6, 1	
1321 S 000	nessun cambio di opinione		
132 <b>1</b> S <b>0</b> 00	voti espressi: 1,0	risultati parziali: 11, 6, 6, 1	
132 S 00	nessun cambio di opinione		
13 <b>2</b> S <b>0</b> 0	voti espressi: 2,0	risultati parziali: 12, 6, 7, 1	
13 S 0	nessun cambio di opinione		
1 <b>3</b> S <b>0</b>	voti espressi: 3,0	risultati parziali: 13, 6, 7, 2	
1 S	nessun cambio di opinione		
1 S	voti espressi: 1	risultati parziali: 13, 7, 7, 2	
S	fine votazione	risultato finale: 13, 7, 7, 2	risposta: 0

*Notate come nella configurazione iniziale, il partito 0 avesse 10 voti, ma in seguito agli scambi di opinioni ne ha avuti alla fine 13. Il partito 1 aveva anch'esso 10 voti inizialmente, ma alla fine ha avuto soltanto 7 voti. Il partito 2 ha mantenuto i suoi 7 voti, e il partito 3 ha mantenuto i suoi 2 voti iniziali. Il*

*programma deve quindi terminare lasciando sul nastro il solo simbolo "0".*

<i>NASTRO INIZIALE</i>	<i>NASTRO FINALE</i>
132100101220121S12030121100200 0	
00201S11113	1
3S	3
S22	2
0012022S12	2
001011011S	1