

Tra le possibili applicazioni delle calcolatrici elettroniche, esiste un insieme di problemi logico-matematici risolvibili con un certo numero di algoritmi più o meno adatti al mezzo di cui disponiamo: in questo numero ci occuperemo del problema legato alla generazione di tutte le possibili permutazioni di n elementi.

Impostati perciò i codici di n lettere, la TI-59 ci stamperà tutte le $n!$ (n fattoriale) stringhe ottenute "rimiscolando" in tutti i modi possibili le n lettere di partenza. Il valore massimo che potremo considerare sarà 10, ma vedremo che in questo caso ci sono degli insormontabili problemi di tempi di elaborazione.

Lasciamo perciò la parola al nostro lettore, per tornare alla fine con delle considerazioni.

Programma "Permutazioni"

di Aldo Caruso (Firenze)

Il programma, progettato in linguaggio S.O.A. per la TI-59, richiede la ripartizione standard di memoria (479.59) e l'uso della stampante.

Esso esegue le $n!$ permutazioni di n simboli alfanumerici dati, sia diversi tra loro, sia nel caso di più simboli ripetuti. In quest'ultimo caso, ovviamente, alcune permutazioni risulteranno uguali.

In pratica, oltre all'uso strettamente matematico - combinatorio, vi ci potete divertire per trovare in breve tempo tutti gli anagrammi in una parola!

Le permutazioni possono essere ordinate in molti modi. Uno degli ordinamenti che hanno un algoritmo veloce è quello del simbolo adiacente. Con questo metodo ogni permutazione viene ricavata dalla

delle permutazioni di $n-1$ simboli, ed aggiungete poi alla serie l' n -esimo simbolo all'estrema destra. Spostatelo in una posizione alla volta nelle successive permutazioni, finché viene a trovarsi all'estrema sinistra. Tenetelo fermo in tale posizione per un'altra volta, prima di iniziare a spostarlo verso destra. Quando ha raggiunto l'estrema destra, tenetelo fermo per un'altra permutazione e poi ricominciate a spostarlo verso sinistra, e così via.

Ad esempio, per $n=3$:

```

1 2 3
1 3 2
3 1 2
-----
3 2 1
2 3 1
2 1 3
-----
1 2 3

```

La nostra programmabile, però, non può seguire questo metodo, ma deve tenere conto nello stesso momento, delle permutazioni del livello n e di quelle dei livelli precedenti, $n-1$, $n-2$, ecc. In sostanza, il simbolo principale (S_n) viene permutato come sopra visto, fino a che viene a trovarsi alla estremità della colonna. Qui "paga pegno", stando fermo un turno.

Viene interpellato il simbolo che lo segue per importanza (S_{n-1}) che si muove anch'esso inizialmente da destra a sinistra di una mossa. Tocca poi di nuovo ad S_n per la traversata dell'intera colonna, e poi di nuovo ad S_{n-1} per una mossa sola, perché al turno successivo si interPELLA di nuovo S_n , che ora è libero.

Quando ognuno dei simboli inferiori, a forza di singole mosse, è arrivato all'estremità della colonna, viene a sua volta interdetto per un turno, e l'interpello ricomincia da S_n .

Dà notare che i simboli inferiori vengono interdetti quando arrivano all'estremità della loro colonna di livello inferiore, anche se sono ancora all'interno della colonna principale. In pratica, ogni simbolo viene interdetto ed inverte il senso di spostamento quando dopo la permutazione si trova ad avere in adiacenza o nessun sim-

bolo o un simbolo di livello superiore.

Per chiarire l'idea, nella permutazione della parola "ANIMO", l'anagramma viene trasformato in quello successivo mediante spostamento a sinistra del simbolo M, mentre il simbolo O è interdetto:

O A M N I
O M A N I

e a questo punto il simbolo M è interdetto, perché nella permutazione del suo livello ($n=4$) il simbolo O non esiste e, nell'anagramma di ANIM in MANI, il simbolo M si sarebbe venuto a trovare all'estremità della propria colonna.

Continuando l'esempio di $n=4$, con una parola costituita (S_1, S_2, S_3, S_4), quando tutti i simboli S_4, S_3 ed S_2 sono contemporaneamente interdetti, tocca la mossa al simbolo S_1 , il quale però non è simbolo itinerante. Infatti con la permutazione di S_1 si riproduce la parola di partenza, il ciclo si chiude e l'algoritmo è terminato.

Chi volesse approfondire la trattazione scientifica di questo e degli altri tipi di permutazioni, può consultare il volume di Page e Wilson "La combinatoria computazionale", Muzzio ed. Padova, pp. 125 e segg.

Tornando al nostro programma, le istruzioni da seguire sono le seguenti:

1. Memorizzare il programma (455 passi) da tastiera o da scheda.
2. Premere il tasto A per l'inizializzazione. Dopo qualche istante il visore mostra O.
3. Impostare il primo simbolo, usando i codici di stampa alfanumerici, e premere il tasto B. Dopo qualche istante, il visore

INVIATECI I VOSTRI PROGRAMMI!

Se, qualunque sia la vostra macchina, avete realizzato programmi o routine che ritenete possano interessare altri lettori, inviateceli. Saranno esaminati e, se pubblicati, ricompensati con valutazioni approssimativamente fra le 30 e le 100.000 lire, secondo la complessità, la genialità, l'originalità e la presentazione del materiale e della documentazione (listati, diagrammi, commenti ecc.). Per ragioni organizzative non possiamo impegnarci, salvo eventuali accordi presi prima dell'invio, alla restituzione dei materiali, che resteranno di proprietà della redazione che si impegna a non divulgarli (se non tramite la rivista) senza l'autorizzazione dei rispettivi autori.

precedente scambiando uno dei simboli (che chiameremo S_k) con quello adiacente, a sinistra o a destra.

Il problema è quindi quello di individuare S_k ed il senso della permutazione, nonché quello di arrestare il programma alla $n!$ -esima permutazione.

L'algoritmo è piuttosto difficile da comprendere. Per una prima comprensione descrittiva, si noti che, per $n=2$, le permutazioni sono 12 e 21. Se $n>2$, fate n copie

Esempi del programma Permutazioni	
12	BCAD DA+BC C+ABD CDBA+ +DBCA
21	BACD DAB+C +CABD CBAID+ +BACD
123	BBCA DABC+ +CADB CBAID+ DB+CA
123	DBCA DABC+ +C+ADB CB+DA DBC+A
312	DBAC DAB+C CA+DB C+EDA DBCA+
321	BDAC DA+CB CAD+B +CBDA DBAC+
231	BDAC DA+CB CAD+B +CBDA DBAC+
231	BDAC DA+CB CAD+B +CBDA DBAC+
213	BADC +DABC CDAB+ C+BAD DB+AC
123	ABCD +ADCB CDAB+ CB+ADI D+BAC
	ABCD+ +A+DCB CD+AD CBAID+ +DBAC
ABCD	ABCD+ +AD+CB C+DAB CBAID+ +BACD
ABDC	AB+CD AID+C +CDAE BCAD+ B+DAC
ABDC	A+BCD ADCB+ +DCAE BCI+D BD+AC
DABC	+ABCD ACDB+ D+CAE BC+AD BDA+C
DABC	+ABCD ACDB+ D+CAE BC+AD BDA+C
DABC	+ABCD ACDB+ D+CAE BC+AD BDA+C
ADCB	A+BDC AC+DB DCA+B +BCAD BADC+
ADCB	AB+DC A+CDB DCAE+ +BCDA BADI+C
ACBD	ABD+C +ACDB DCBA+ B+CDA BA+DC
CABD	A+DAB ACBD+ +CDBA BDC+A B+A+D
CADB	ADBC+ A+CBD DC+BA BCI+D +BADC
CDAB	ADB+C AC+BD D+CBA BCI+D +BACD
DCBA	AD+BC ACB+D +DCBA BDC+A B+ACD
DCBA	A+DAB ACBD+ +CDBA BDC+A BA+C
CDAB	+ADBC CABID+ C+DBA BD+CA BAC+D
CBDA	+DABC CAB+D C+DBA BD+CA BAC+D
CBAD	D+ABC CA+BD CDB+A +BDCA ABCD+

eliminare le etichette, sostituendole con i salti assoluti, per avere un discreto aumento della velocità di elaborazione e per avere una leggera diminuzione di passi di programma. Anche in questo caso però non riteniamo

che si riesca a scendere molto al di sotto dell'"anno": chi avrebbe il coraggio di impegnare la propria calcolatrice per così tanto tempo?

Forse qualche lettore sarà in grado di

proporre un altro programma in grado di stampare tutte le 10! permutazioni in un tempo ragionevole? Oppure riuscirà ad aumentare il valore di n?

Buon lavoro!!!

L'ANGOLO DELLE TI-L'ANGOLO DELLE TI-L'ANGOLO DELLE TI-L'ANGOLO DELL

Nei numeri scorsi abbiamo più volte richiesto la collaborazione dei lettori, nel segnalare notizie, consigli, curiosità, nel funzionamento delle calcolatrici TI-58 e TI-59. Questa volta il contributo all'"angolo", proviene in gran parte da un simpatico lettore di Bologna, Stefano Laporta, il quale ci ha inviato, oltre ad alcuni programmi interessanti, un insieme di "Routinette e consigli utili". Vediamo dunque cosa ci scrive.

1) Vogliamo provare i flag da tastiera? Impossibile fino ad oggi...

La sequenza risolutrice è semplice: Ifllg n 999.

Se il display lampeggia vuol dire che il flag n è settato (l'errore segnalato dal display nasce infatti dal volere effettuare un salto al passo 999, inesistente in qualunque caso), se invece il display rimane inalterato il flag è "spento". In entrambi i casi, per la nostra tranquillità, il contatore di programma non si sposta.

2) Stranezze con le TI.

Registri usati da alcune funzioni

Funzione	registri	contenuto
P/R	x t HIR (1+n) HIR 7 HIR 8	t · sen(x) t · cos(x) cos(x) x t
R/P	x t HIR (1+n) HIR 7 HIR 8	atan(x/t)+90 · (1-sign(t)) $\sqrt{x^2 + t^2}$ x ² t t
DMS	HIR (1+n) HIR (2+n) HIR 8	36 risultato di DMS Fraz(x) 100 · Fraz(x)
INV DMS	HIR (1+n) HIR (2+n) HIR 8	100 risultato di INV DMS Fraz(60 · x) Fraz(60 · x)+Int(100 · risult.)
\bar{x}	x t HIR (1+n)	R01/R03 R04/R03 R01

Esiste un certo numero di sequenze che "mandano in pallone" la nostra macchinetta (niente paura! non le succede niente). Alcune sono, diciamo così, "lecite", altre sono "illecite". Per "lecite" intendiamo le sequenze che provocano un errore nella calcolatrice e che è relativamente facile eseguire anche senza accorgersene: un esempio è premere consecutivamente due tasti di operazioni oppure calcolare il logaritmo di 0. Alcune volte queste sequenze si usano di proposito nei programmi con lo scopo di evidenziare un certo risultato, richiamando l'attenzione dell'operatore.

"Illecite" sono invece quelle sequenze non propriamente intuitive, nel senso che bisogna mettersi lì a provare e riprovare, nell'intento di trovare qualcosa di strano.

Un insieme di tali sequenze nasce quando vogliamo impostare delle istruzioni al di là del limite imposto dalla ripartizione.

Ad esempio, dopo aver acceso la calcolatrice, impostiamo GTO 479 (per la TI-59) o GTO 239 (per la TI-58): premiamo CLR e LRN per entrare nel modo di apprendimento, ben consci che potremo impostare solo un passo di programma, dopodiché la calcolatrice uscirà automaticamente dal modo di apprendimento.

Se invece, diabolicamente, impostiamo un'istruzione più lunga (ad es. anche STO 00), sul display vedremo la "protesta" della calcolatrice: si avrà ".00000000 0".

Di peggio succede con istruzioni a più byte, fino ad arrivare alla "Dsz Ind 00 Ind 00" per la quale il display cambierà più volte a mano a mano che si premono i tasti.

Il lettore di Bologna ci segnala in particolare:

Dsz 2 +/- 123456789

a seguito della quale il display mostrerà file di cifre ben strane. Altri "segnali di pazzia" si avranno nel caso che si imposti, preventivamente, la notazione esponenziale.

Complichiamo poi il tutto accendendo la stampante e ponendo la calcolatrice in "TRACE" (funzionamento che consente di seguire l'esecuzione passo-passo).

Dopo aver premuto LRN Dsz 2 avremo subito sul display qualcosa di sconcertante: ".0000*0 0002" dove al posto dell'asterisco compare in realtà uno "0" formato dai quattro segmenti superiori della singola "cifra", così come potrebbe apparire il simbolo di "gradi".

Un altro esempio riguarda invece un piccolo "buco" nel complicato sistema operativo della calcolatrice.

Supponiamo di avere una TI-59 e di impostare la ripartizione 5 Op 17 (559.49 sul display): con GTO 482 ci portiamo dunque al passo 482 della memoria di programma.

Ripristiniamo ora da tastiera la ripartizione originaria con 6 Op 17 (479.59 sul display) mentre, ricordiamo, il contatore di programma è posto sul vecchio passo 482, che ora è al di là del limite consentito.

Se ora premiamo LRN, come è lecito aspettarsi, la calcolatrice, per il motivo appena accennato, non entra in modo di apprendimento: però la possiamo ingannare premendo BST (che fuori dal modo di apprendimento non funziona, normalmente), ottenendo a sorpresa la visualizzazione di un fantomatico passo 481! Il bello è che anche premendo varie volte il tasto LRN si ritornerà sempre a quel passo, che apparentemente la calcolatrice assume come "buono".

Ma non è finita: impostiamo ora 5 Op 17 e andiamo in modo LRN. Contrariamente ad ogni logica, premendo il tasto SST, dal passo 481 arriviamo al passo... 472!

3) Ancora sull'HIR (vedi MCmicrocomputer n. 4).

Il lettore segnala che i malfunzionamenti da noi riscontrati sui "vecchi" modelli di TI-58 e 59 (operazioni con x² e con numeri in valore assoluto minore di 1), non si riscontrano invece sulle più recenti TI-58C, forse a causa di migliorie apportate al sistema operativo della calcolatrice.

4) La tabella mostra, per le funzioni riportate sulla sinistra i registri usati (x, t, HIR) con i loro contenuti. In particolare "n" è il numero di operazioni lasciate in sospeso prima dell'esecuzione della funzione.

Ad esempio può essere utile la DMS per isolare la parte decimale di un numero e per moltiplicarla automaticamente per 100: impostato 4.51 DMS, con HIR 18 otterremo appunto 51.

Chiudiamo questa puntata con una specie di scherzo: visto che abbiamo parlato di stranezze, segnaliamo di aver trovato, dopo anni di esperienza con le TI, una sequenza di istruzioni che blocca completamente la calcolatrice, la quale — poverina! — non risponderà più alla pressione dei tasti e mostrerà soltanto la "C" di elaborazione sulla sinistra del display: in questo caso l'unico rimedio è spegnere la calcolatrice e riaccenderla.

Qualche lettore sarà capace di trovare questa sequenza, della quale non diciamo di più, che veramente mette in crisi la nostra calcolatrice?

P.P.