

Simon 41C

di Paolo Baiardi - Varazze (SV)

Sono uno studente di 17 anni, che attualmente frequenta un liceo scientifico e, come appassionato del mondo degli elaboratori in generale, desidero poter contribuire alla vostra rivista.

Questo programma simula il noto gioco elettronico "Simon", nel quale vengono posti dei suoni e colori associati ad essi in sequenza; il giocatore deve ricordarli correttamente e ripetere la stessa sequenza premendo i settori opportuni. Sul 41, il gioco si svolge allo stesso modo, ma data l'evidente impossibilità del calcolatore di generare diversi colori, ho rimediato sui numeri. Si opera in questo modo: per iniziare si pone il 41 nel modo USER e si preme 'a', il calcolatore chiede: RAND.?

cioè su quanti settori si basa la sequenza (i settori sono al massimo 10 e sono assegnati alle prime due file di tasti, da A a E e da F a J) dando ad esempio cinque la calcolatrice assumerà che la sequenza debba contenere solo numeri corrispondenti ai tasti da A a E, dando 10 tutti i tasti, o qualsiasi combinazione intermedia. Dopo aver impostato il numero premere R/S e verrà posta questa domanda:

SEQ.?
cioè la lunghezza della sequenza (ricordare che essa non può superare S - 5 dove S è il size del 41 prescelto in quel momento), rispondere e premere R/S.

Dopo di ciò passerà del tempo dipendente dalla lunghezza della sequenza, in seguito comparirà READY, ciò significa che si può iniziare. Premere R/S e si otterrà il primo numero della sequenza, seguito dal corrispondente avviso acustico e, dopo qualche secondo, il numero seguente (a differenza del Simon, il 41 inizia con due numeri), comparirà poi un "?" e ciò indica che è necessario riprodurre la sequenza; premere il tasto corrispondente al primo numero, attendere la comparsa del successivo "?" dare il secondo e attendere.

Questi i tasti corrispondenti (nel modo USER):

- a = inizio del gioco
- A = 0
- B = 1
- C = 2
- D = 3
- E = 4

- F = 5
- G = 6
- H = 7
- I = 8
- J = 9

Compariranno i primi due numeri, seguiti dal terzo, con i rispettivi avvisi acustici, poi il punto interrogativo e così sino alla fine della sequenza annunciata all'apparire del messaggio "OKAY".

In caso di errore o di tempo eccessivo per la risposta (3" = massimo) si avrà questa segnalazione di errore:

ERROR: n
seguito da un beep, dove n è il numero corretto della sequenza, e il programma ripartirà dalla LABEL a.

Il Simon è un giochetto niente male, e i possessori di un HP41C/CV non disdegneranno certo di programmarlo sulla loro macchinetta anziché doverlo acquistare.

* * *

Pur non avendo la pretesa di imitare il gioco originale con le luci e i colori, il programma di Paolo funziona in modo perfettamente analogo. Con la 41C senza espansioni di memoria, avanzano 30 registri dati, questo significa che è possibile giocare con sequenze lunghe al massimo 25 numeri,

il che è più che sufficiente per atterrare anche Pico della Mirandola, specialmente se il numero dei settori scelti è alto.

Il programma fa uso di un generatore di numeri casuali programmabile che genera numeri interi da 0 a n-1, dove n è il numero dei settori che si vogliono utilizzare. All'inizio del gioco una routine provvede a caricare con numeri casuali da 0 a n-1 tanti registri dati quanti sono indicati dal numero impostato come risposta alla domanda "SEQ?"; dopodiché provvede a mostrare i primi due per poi passare a confrontarli con i numeri impostati dal giocatore; se non ci sono errori, l'operazione si ripete, ma questa volta i numeri mostrati sono tre, e così via finché il giocatore riuscirà a ricordare la sequenza che via via diviene sempre più lunga; se una volta arrivati alla lunghezza massima impostata per la sequenza, il giocatore risponderà ancora correttamente, il gioco termina con un "OKAY" altrimenti, al primo errore, il gioco termina con un messaggio ERROR: N, dove N è il numero che è stato sbagliato nella sequenza. Per una buona rappresentazione numerica, è bene che la macchina sia disposta in FIX 0 e CF 29.

| Simon 41C | | | | |
|-----------|--------------|---------------|---------------|-----------------|
| 01*LBL A | 27 TONE 6 | 53 STO 00 | 79 RCL 02 | 105 DSE 03 |
| 02 0 | 28 GTO 00 | 54 STO 01 | 80 FRC | 106 GTO 04 |
| 03 TONE 0 | 29*LBL H | 55*LBL 01 | 81 .001 | 107 RCL 01 |
| 04 GTO 00 | 30 7 | 56 XEQ 02 | 82 + | 108 STO 00 |
| 05*LBL B | 31 TONE 7 | 57 STO IND 00 | 83 STO 02 | 109 GTO 03 |
| 06 1 | 32 GTO 00 | 58 ISG 00 | 84 RCL 01 | 110*LBL 06 |
| 07 TONE 1 | 33*LBL I | 59 GTO 01 | 85 STO 00 | 111 *ERROR: " |
| 08 GTO 00 | 34 8 | 60 RCL 01 | 86*LBL 04 | 112 ARCL IND 00 |
| 09*LBL C | 35 TONE 8 | 61 STO 00 | 87 *?" | 113 AVIEW |
| 10 2 | 36 GTO 00 | 62 .001 | 88 AVIEW | 114 BEEP |
| 11 TONE 2 | 37*LBL J | 63 STO 02 | 89 PSE | 115 GTO a |
| 12 GTO 00 | 38 9 | 64 *READY" | 90 PSE | 116*LBL 05 |
| 13*LBL D | 39 TONE 9 | 65 TONE 9 | 91 PSE | 117 ISG 00 |
| 14 3 | 40 GTO 00 | 66 TONE 9 | 92 GTO 06 | 118 GTO 03 |
| 15 TONE 3 | 41*LBL a | 67 PROMPT | 93*LBL 00 | 119*LBL 02 |
| 16 GTO 00 | 42 *RAND. ?" | 68*LBL 03 | 94 RCL IND 00 | 120 RCL 04 |
| 17*LBL E | 43 PROMPT | 69 CLD | 95 X*Y? | 121 9821 |
| 18 4 | 44 STO 05 | 70 RCL IND 00 | 96 GTO 06 | 122 * |
| 19 TONE 4 | 45 *SEQ. ?" | 71 VIEW X | 97 ISG 00 | 123 .211327 |
| 20 GTO 00 | 46 PROMPT | 72 TONE IND X | 98 GTO 07 | 124 + |
| 21*LBL F | 47 5 | 73 CLX | 99*LBL 08 | 125 FRC |
| 22 5 | 48 + | 74 ISG 02 | 100 *OKAY" | 126 STO 04 |
| 23 TONE 5 | 49 1 E3 | 75 GTO 05 | 101 BEEP | 127 RCL 05 |
| 24 GTO 00 | 50 / | 76 RCL 02 | 102 BEEP | 128 * |
| 25*LBL G | 51 6 | 77 INT | 103 PROMPT | 129 INT |
| 26 6 | 52 + | 78 STO 03 | 104*LBL 07 | 130 END |

Resist 41C

di Roberto Gullino - Torino

Spett. MCmicrocomputer

il programma che vi invio non è una mia realizzazione, bensì una 'traduzione' dal BASIC del programma 'RESIST' di Bo Arnklit, pubblicato tempo fa su un'altra rivista. Il programma, per chi non l'ha visto, serve a risolvere il seguente problema: data una resistenza di valore qualsiasi, trovare due resistenze standard che in parallelo diano un valore che approssimi, entro una percentuale prestabilita, quello della resistenza data.

Le soluzioni possibili sono tre:

- 1) la resistenza è già, entro la percentuale stabilita, compresa nella serie standard;
- 2) esiste una combinazione con resistenza equivalente entro la tolleranza richiesta;
- 3) non si rientra in questa tolleranza.

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.

La versione RPN di RESIST che ho realizzato è ai limiti di capacità di memoria della HP 41-C (61 registri), per cui sono necessarie, dopo il caricamento del programma, alcune operazioni 'a mano', come l'introduzione dei valori delle resistenze serie E12 nei registri da R14 a R26, e l'introduzione della tolleranza desiderata in R05 (se PERC=1%, si deve digitare 0.01).

Il programma parte battendo XEQ 01 dopo avere digitato la resistenza desiderata: si passa subito alla subroutine 00 per confrontarla con le resistenze standard; se essa è, entro l'errore richiesto, compresa nella E12, allora la calcolatrice emette un tono acuto e visualizza la resistenza standard; se non lo è si ripassa alla subroutine con $R=2 * RD$ per trovare la prima resistenza di valore maggiore o uguale a RD. Con questo valore si passa, dalla LBL 11, alla routine di ricerca facendo tutte le combinazioni possibili con resistenze di valore più elevato, fino a quando l'errore non diventa positivo; se a questo punto l'errore ottenuto è superiore a quello voluto, si ripete il procedimento adottando un valore inferiore della serie standard.

L'iterazione prosegue fino a quando la

ELENCO VARIABILI

| Nome | registro di memoria |
|--------|--|
| RD | : resistenza desiderata R00 |
| R | : variabile di calcolo R01 |
| MINERR | : minimo errore ottenuto R03 |
| P | : ordine di grandezza di RY R06 |
| R(I) | : resistenza standard lesima IND09 |
| I | : contatore RY R09 |
| RY | : 1° resistenza in parallelo R01 |
| ER | : errore relativo fra RD e RY |
| PERC | : errore ammesso R05 |
| P1 | : ordine di grandezza di riferimento R07 |
| M | : ordine di grandezza di RX R11 |
| II | : contatore di riferimento R08 |
| N | : contatore RX R10 |
| R(N) | : resistenza standard Nesima IND10 |
| RX | : 2° resistenza in parallelo R02 |
| ERR | : errore fra (RY//RD) e RD R04 |

Nei registri R12 e R13 ci sono le RY e RX relative alla migliore combinazione ottenuta; nei registri da R14 a R26 ci sono i valori delle resistenze standard della serie E12, che sono: 1, 1.2, 1.5, 1.8, 2.2, 2.7, 3.3, 3.9, 4.7, 5.6, 6.8, 8.2, 10.

| | |
|------------------|---------------|
| 150*LBL 13 | 171 FIX 2 |
| 151 TONE 5 | 172 1 E2 |
| 152 *OK: ERR.=- | 173 * |
| 153 RCL 03 | 174 ARCL X |
| 154 FIX 2 | 175 *Iz* |
| 155 1 E2 | 176 AVIEW |
| 156 * | 177 ENG 2 |
| 157 ARCL X | 178 CLA |
| 158 *Iz* | 179 ARCL 12 |
| 159 AVIEW | 180 *I/// |
| 160 ENG 2 | 181 ARCL 13 |
| 161 CLA | 182 AVIEW |
| 162 ARCL 01 | 183 STOP |
| 163 *I/// | 184*LBL 15 |
| 164 ARCL 02 | 185 TONE 9 |
| 165 AVIEW | 186 *R. E12=- |
| 166 STOP | 187 ENG 2 |
| 167*LBL 14 | 188 ARCL 2 |
| 168 TONE 0 | 189 AVIEW |
| 169 *MIN. ERR.=- | 190 STOP |
| 170 RCL 03 | 191 END |

Figura 1

resistenza di riferimento è minore di RD (questo vuol dire che non esiste una combinazione ottimale); a questo punto la calcolatrice emette un tono grave e sul display appare la tolleranza relativa alla migliore combinazione trovata; premendo poi R/S due volte si visualizzano i valori di questa combinazione; se invece durante il calcolo la combinazione con resistenza equivalente entro la tolleranza viene trovata, si sente un BEEP e sul display si legge (in %) l'errore ottenuto, premendo poi R/S due volte si

```
01*LBL "RES"
02 CF 29
03 SF 21
04 "RESISTENZA ?"
05 PROMPT
06*LBL 01
```

Figura 2

Resist 41C

| | | | | | |
|-----------|---------------|-------------|----------------|-------------|--------------------|
| 01*LBL 01 | 36 RCL 11 | 71 RCL 04 | 106 RCL 06 | 141 * | 01*LBL "IN" |
| 02 STO 00 | 37 INT | 72 X>Y? | 107 RCL 07 | 142 RTN | 02 *TOLLER. IN ??" |
| 03 STO 01 | 38 10IX | 73 GTO 07 | 108 X=Y? | 143*LBL 12 | 03 PROMPT |
| 04 1 | 39 RCL IND 10 | 74 ABS | 109 GTO 14 | 144 RCL 11 | 04 1 E2 |
| 05 STO 03 | 40 * | 75 X<Y? | 110 GTO 11 | 145 INT | 05 / |
| 06 XEQ 00 | 41 STO 02 | 76 GTO 13 | 111*LBL 00 | 146 RCL 07 | 06 STO 05 |
| 07 RCL 06 | 42 1/X | 77*LBL 00 | 112 RCL 01 | 147 X>Y? | 07 1 |
| 08 STO 07 | 43 RCL 01 | 78 ISG 10 | 113 LOG | 148 GTO 00 | 08 STO 14 |
| 09 ENTER↑ | 44 1/X | 79 GTO 04 | 114 INT | 149 RTN | 09 1.2 |
| 10 ENTER↑ | 45 + | 80 14.026 | 115 STO 06 | 150*LBL 13 | 10 STO 15 |
| 11 2 | 46 1/X | 81 STO 10 | 116 14.026 | 151 BEEP | 11 1.5 |
| 12 + | 47 RCL 00 | 82 ISG 11 | 117 STO 09 | 152 RCL 03 | 12 STO 16 |
| 13 1 E3 | 48 - | 83 GTO 04 | 118*LBL 03 | 153 1 E2 | 13 1.8 |
| 14 / | 49 LASTX | 84*LBL 07 | 119 XEQ 02 | 154 * | 14 STO 17 |
| 15 + | 50 / | 85 1 | 120 ENTER↑ | 155 STOP | 15 2.2 |
| 16 STO 11 | 51 STO 04 | 86 31 - 0.2 | 121 ENTER↑ | 156 RCL 01 | 16 STO 18 |
| 17 RCL 09 | 52 ABS | 87 RCL 09 | 122 RCL 00 | 157 STOP | 17 2.7 |
| 18 INT | 53 RCL 03 | 88 INT | 123 - | 158 RCL 02 | 18 STO 19 |
| 19 STO 08 | 54 X<Y? | 89 14 | 124 LASTX | 159 STOP | 19 3.3 |
| 20 RCL 00 | 55 GTO 06 | 90 - | 125 / | 160*LBL 14 | 20 STO 20 |
| 21 2 | 56 X<Y? | 91 CHS | 126 ABS | 161 TONE 0- | 21 3.9 |
| 22 * | 57 STO 03 | 92 X<=0? | 127 RCL 05 | 162 RCL 03 | 22 STO 21 |
| 23 STO 01 | 58 RCL 06 | 93 GTO 09 | 128 X<Y? | 163 1 E2 | 23 4.7 |
| 24 XEQ 00 | 59 10IX | 94 25.026 | 129 X<Y? | 164 * | 24 STO 22 |
| 25*LBL 11 | 60 RCL IND 09 | 95 STO 09 | 130 GTO 15 | 165 STOP | 25 5.6 |
| 26 XEQ 02 | 61 * | 96 1 | 131 RCL 2 | 166 RCL 12 | 26 STO 23 |
| 27 STO 01 | 62 STO 12 | 97 ST- 06 | 132 RCL 01 | 167 STOP | 27 6.8 |
| 28 14.026 | 63 RCL 11 | 98*LBL 09 | 133 X<Y? | 168 RCL 13 | 28 STO 24 |
| 29 STO 10 | 64 INT | 99 RCL 09 | 134 RTN | 169 STOP | 29 8.2 |
| 30*LBL 04 | 65 10IX | 100 INT | 135 ISG 09 | 170*LBL 15 | 30 STO 25 |
| 31 RCL 10 | 66 RCL IND 10 | 101 RCL 00 | 136 GTO 03 | 171 TONE 9 | 31 10 |
| 32 INT | 67 * | 102 X>Y? | 137*LBL 02 | 172 RCL 2 | 32 STO 26 |
| 33 RCL 08 | 68 STO 13 | 103 GTO 10 | 138 RCL 06 | 173 STOP | 33 GTO *PES* |
| 34 X>Y? | 69*LBL 06 | 104 GTO 11 | 139 10IX | 174 END | 34 END |
| 35 XEQ 12 | 70 RCL 05 | 105*LBL 10 | 140 RCL IND 09 | | |

hanno i valori cercati.

Avendo a disposizione un modulo di memoria si possono modificare le procedure di INPUT e OUTPUT in modo da renderle più comode, automatizzando, per esempio, il caricamento delle resistenze standard; volendo si può sostituire la serie E12 con la E24 modificando opportunamente le linee 28, 80, 116, 94 e 88.

Sperando di essere stato abbastanza chiaro, saluto cordialmente.

* * *

Fa comodo, a chi progetta circuiti elettronici, avere la possibilità di disporre di una vasta gamma di valori resistivi, semplicemente accoppiando due resistori della reperibilissima serie E12. Il programma del Signor Gullino consente ciò ai possessori di una 41C/CV.

Forse per mancanza di spazio, l'input output dei dati è un po' "grezzo" e, senza alcun messaggio alfanumerico, ricorda un po' l'uso della HP67, ma chi dispone di moduli di espansione di memoria, può senz'altro dare quel "tocco finale" che manca al programma. Questioni estetiche a parte, l'unico inconveniente serio causato da questa mancanza di spazio, è la necessità di dover rimemorizzare a mano tutti i dati della serie E12, ogni volta che per qualsiasi ragione viene modificato il contenuto dei registri da R14 a R26. Per questo propongo di caricare in memoria (chi ha spazio) anche la routine "IN", che serve a introdurre nei registri dati tutti i valori della serie E12, nonché la percentuale di approssimazione desiderata, direttamente in unità percento. Sempre chi ha spazio in me-

moria, può modificare l'input-output come segue:

1) Sostituire tutta la sequenza dal passo 150 LBL 13 fino alla fine, con i passi indicati nel list riportato in figura 1.

2) Aggiungere prima della LBL 01 i passi (01-05) elencati nel list di figura 2.

Per usare il programma così modificato, premere XEQ "RES" o il tasto al quale è stata assegnata la LBL "RES", la calcolatrice chiederà "RESISTENZA?", impostare il valore della resistenza desiderata e premere R/S, la calcolatrice risponderà in uno dei tre modi possibili:

1) Coppia di resistenze soddisfacenti le condizioni desiderate: suono medio e visualizzazione dell'errore ottenuto, poi, premendo R/S, in un'unica stringa Alpha visualizzazione dei valori delle due resistenze da collegare in parallelo.

2) Coppia di resistenze non soddisfacenti le condizioni di approssimazione desiderate: suono grave, visualizzazione del minimo errore ottenuto poi, premendo R/S, visualizzazione della coppia di resistenze.

3) Valore cercato già compreso nella serie E12: suono acuto e visualizzazione del valore E12.

Con queste modifiche il programma è perfettamente compatibile nell'uso con la stampante.

Unico neo del procedimento, il programma non controlla i dati impostati, per cui chiedendo ad esempio un valore di 600 miliardi di ohm, la 41C risponde innocentemente di collegarne in parallelo due da 1200 miliardi!

MC

