

Questo mese presentiamo due programmi dei lettori, uno molto "serio" per il calcolo delle reti elettriche in corrente continua, che interesserà sicuramente molti studenti degli Istituti Tecnici, e uno un po' più ameno che, seppure con un sottofondo di scientifica verità, ci permetterà di stupire gli amici mostrando loro le doti di intelligenza e, perché no, di divinazione del nostro computer.

Il gioco del si/no ovvero i fondamenti dell'Informatica Psicologica

di Francesco Lentini
Reggio Calabria

Su che cosa si basa

Il modo migliore per dimostrare "l'intelligenza" di un computer è quello di fargli delle domande facendo in modo di ottenere delle risposte plausibili.

Si tratta di un metodo che ha successo nella maggior parte dei casi perché fa appello a qualcosa che uno studioso di psicanalisi chiamerebbe ARCHETIPO.

Se siete un venditore di bacchette magiche, e riuscite (non importa in quale modo) a convincere la gente che funzionano, state certi che ne venderete una per ogni abitante della terra.

Gli ARCHETIPI infatti non riguardano un singolo paese o nazione, ma sono universali.

Se offrite a qualcuno una macchina in grado di risolvere tutti i suoi problemi con la semplice pressione di un tasto, è molto probabile che abbiate successo perché fate appello a un'esigenza molto sentita da chiunque, che è quella di un essere onnipotente e onnisciente (non importa se uomo o dio) a cui affidarsi completamente.

Una macchina del genere ovviamente non esiste, ma il desiderio che esista è così radicato in noi che può essere facile crearne un'efficiente imitazione.

Avete mai visto quelle esibizioni (secondo me molto penose) di animali addestrati che riescono a rispondere a semplici domande, a contare e a fare perfino le quattro operazioni?

Si chiama Trucco degli Animali Intelligenti e si basa essenzialmente su due cose: 1) l'addestramento lungo e paziente dell'animale (sarebbe più giusto parlare di "condizionamento");

2) la possibilità di comunicare con l'animale senza che gli spettatori se ne accorgano.

Ad esempio per ottenere da un asino un conteggio da uno a dieci è sufficiente addestrarlo a battere ripetutamente a terra la zampa destra, e arrestare il conteggio alla decima battuta mediante un segnale cono-

sciuto solo dall'animale e dal suo addestratore.

Ecco come sfruttare l'Archetipo del Computer e il Trucco degli Animali Intelligenti per un gioco di sicuro effetto.

Come funziona

Stando tranquillamente seduti davanti al computer si lancia il programma e si chiede ai presenti di formulare delle do-

mande che implicino soltanto risposte di tipo SI/NO.

Domande di altri tipi ("perché", "che cosa", ecc.) non sono ammesse.

Naturalmente prima di cominciare si precisa di aver memorizzato un vocabolario di 5000-10000 parole e di aver istruito il computer su svariati argomenti (lavoro, amore, tempo libero, ecc.) in modo da ottenere risposte plausibili nell'80% dei casi

```

0 REM THE SI/NO GAME
1 REM (C) LENTINI 1982
2 REM
10 HOME : NC = 255 : CG# = CHR# (7) : POKE 214,128
20 INVERSE : PRINT SPC( 13) : *THE SI/NO GAME* : SPC( 13) : NORMAL
30 PRINT CG#
40 FOR PSE = 1 TO 500 : NEXT
50 VTAB 22 : HTAB 1
60 FLASH : PRINT SPC( 12) : *PUT YOUR QUESTION* : SPC( 11) : NORMAL
70 VTAB 5 : PRINT ">": CALL - 868
80 GET C#
90 IF ASC (C#) = 27 THEN GOSUB 60000
100 IF ASC (C#) < 45 OR ASC (C#) > 90 THEN GOSUB 1000 : GOTO 50
110 PRINT C#
120 FOR PSE = 1 TO 35
130 X = PEEK ( - 16384)
140 NEXT
150 POKE - 16384,0
160 IF X > 127 THEN PRINT CHR# (X) : R# = *SI* : GOTO 180
170 R# = *NO*
180 FOR C = 1 TO NC
190 GET C#
200 IF C# = "?" GOTO 235
210 IF ASC (C#) = 13 THEN GOSUB 1000 : GOTO 50
220 PRINT C#
230 NEXT
235 PRINT C#
240 VTAB 22 : HTAB 1 : CALL - 868
245 FOR PSE = 1 TO 3000 : NEXT
250 VTAB 14 : HTAB 19
260 PRINT CHR# (7)
270 INVERSE : PRINT R# : NORMAL
280 FOR PSE = 1 TO 2000 : NEXT
290 RUN
1000 REM INVALID KEY
1005 IK = IK + 1
1006 IF IK > = 3 THEN GOSUB 3000
1010 VTAB 22 : HTAB 1 : CALL - 868
1012 INVERSE : PRINT SPC( 14) : *INVALID KEY* : SPC( 14) : NORMAL
1014 PRINT CG#
1015 FOR PSE = 1 TO 1000 : NEXT
1020 RETURN
3000 REM INTELLIGENT GAME
3010 VTAB 22 : HTAB 1 : CALL - 868
3015 FOR CG = 1 TO 20 : PRINT CG# : NEXT
3017 FOR PSE = 1 TO 500 : NEXT
3020 INVERSE : PRINT *MR. OR MADAME, THIS IS AN* : NORMAL : PRINT * INTELLI
      GENT * : INVERSE : PRINT *GAME* : NORMAL
3025 FOR PSE = 1 TO 3000 : NEXT
3035 IF IK > = 6 THEN HOME : PRINT *GO HOME!!!* : NEW
3050 RETURN
60000 REM USCITA
60005 HOME : SPEED= 100
60010 PRINT *THAT'S ALL FOLKS...* : CG# : CG#
60015 SPEED= 255
60020 POKE 214,127 : NEW
    
```

(questa percentuale è molto importante, come vedremo).

Si parte allora con degli esempi di tipo assolutamente generico che implicino una serie di risposte assortite ("...Pippo è un bravo ragazzo?", e cose del genere).

Per ottenere la risposta "NO" si digita la prima lettera della prima parola seguita dopo almeno un secondo dalla seconda lettera. Per ottenere la risposta "SI" si digita la seconda lettera immediatamente dopo la prima. Le lettere successive possono essere digitate in qualsiasi modo. È indispensabile concludere la domanda col punto interrogativo.

Ecco quanto accade all'interno del programma.

Battuto il primo tasto parte un loop che legge la tastiera mediante un test sulla locazione speciale -16384. Se durante il loop viene premuto un qualsiasi altro tasto, il codice ASCII del valore contenuto in questa locazione diventa maggiore di 127. Con un semplice test successivo alla pressione del secondo tasto si va a stabilire se i tasti sono stati premuti a distanza l'uno dall'altro o in rapida successione.

Dopo aver "riflettuto" per qualche secondo, il computer fornirà la sua risposta e sarà pronto ad accettare la prossima domanda.

Dopo tre o quattro domande il più "sveglio" dei nostri spettatori dirà di aver capito tutto e che le risposte sono semplicemente casuali, cioè alla lunga prive di senso. Si ripropongono allora le domande già fatte facendo in modo di ottenere le stesse risposte: questo boccia definitivamente la tesi della casualità.

Sentendosi preso in giro, il nostro amico pretenderà di digitare lui stesso le domande, e noi lo accontenteremo senza esitare. Se non lo facesse lui saremmo noi ad invitarlo dopo una decina di domande.

A questo punto le possibilità sono due:

1) il nostro amico non ha molta pratica di computer e/o di tastiere: difficilmente riuscirà a trovare il secondo tasto immediatamente dopo il primo, e la risposta sarà ripetutamente NO (che è sempre la risposta migliore da dare quando non si è capita la domanda. Inoltre un diniego è sempre frustrante, soprattutto se dato da un computer).

2) Il nostro amico ha pratica della tastiera oppure è un impulsivo oppure ha fretta di trovare la soluzione: in questo caso si avranno alcuni SI e molti NO con un certo numero di risposte sbagliate.

In entrambi i casi si supera l'inconveniente con un pretesto di forma: il computer comprende le domande solo se poste in un determinato modo. Con questa scusa si riacquista la conduzione del gioco e si ripropongono in un'altra forma le domande che a dire del nostro interlocutore avevano prodotto una risposta sbagliata. È ovvio che questa volta la risposta sarà opposta.

La buona riuscita di questo gioco è ovviamente basata sull'abilità di chi lo conduce di conoscere e/o prevedere le reazioni dello spettatore (come insegnano i vari maghi, prestigiatori, telepati e veggenti).

Ad esempio sarà cura del conduttore di porre domande la cui risposta sia nota alla persona interessata ma anche a lui stesso (magari all'insaputa di tutti). Se qualcuno dei presenti insiste nel fare domande personali, l'esperienza insegna che queste domande ricadono o nella sfera futura (e allora si fa presente che il futuro non può essere conosciuto neanche dal computer) o nella sfera affettiva (e allora non resta che affidarsi alla conoscenza che si ha della persona o al proprio intuito personale).

Comunque vadano le cose ci sarà sempre un certo numero di risposte completamente sbagliate.

Si dirà allora che il vocabolario è ancora limitato, e che il computer impara dalle nostre stesse domande. Agli irriducibili si rammenterà che, dopotutto, la percentuale promessa era dell'80% di risposte esatte, e si lascerà intendere che allo stato attuale della scienza questo è un risultato più che accettabile.

Verso la fine del gioco può verificarsi l'eventualità che il nostro amico perda la pazienza e cominci a battere dei tasti a caso o tenti di listare il programma. Ecco l'effetto dei vari tasti:

- Il tasto RESET arresta il programma ma il comando LIST susseguente ottiene il solo scopo di farlo partire daccapo. Questo serve a rendere invisibile l'inesistente vocabolario.

- Come primo tasto di una domanda è ammessa soltanto una lettera. Il tasto ESC

provoca l'uscita normale del programma. Qualsiasi altro tasto provoca la comparsa di un messaggio di errore. Dopo un certo numero di errori apparirà un messaggio di rimprovero e dopo ripetuti rimproveri si avrà l'uscita forzata dal programma (con autodistruzione).

- Il tasto RETURN premuto in qualsiasi momento equivale a un errore. Questo serve a dare l'impressione di qualcosa di strano a chi ha già usato il computer.

Sfruttando a fondo i principi esposti si possono raggiungere risultati notevoli. Provate infatti ad immaginare cosa potrebbe fare un prestigiatore che riuscisse a mettere a punto un codice di comunicazione uomo-computer sufficientemente complesso (per esempio basato su intervalli brevi e lunghi tra le lettere). La gamma di risposte ottenibili sarebbe molto vasta, e il sospetto che il "canale" di comunicazione sia la tastiera sarebbe tanto minore quanto maggiore la complessità del codice.

E poi ottenere una risposta alle proprie domande (e quindi una soluzione di propri problemi) è proprio quello che la stragrande maggioranza della gente si aspetta dall'affascinante mondo dei computer.

Bibliografia

Pierre Daco
"Che cos'è la psicanalisi"
Enciclopedie Pratiche Sansoni

Circuiti elettrici in corrente continua

di Renzo Del Fabbro
Pieve Di Cadore (BL)

Nato come programma dimostrativo per un corso introduttivo di calcolo numerico in un istituto secondario superiore, il programma si propone la risoluzione completa di una rete elettrica in corrente continua e in regime stazionario.

Inviare i vostri programmi

Alcuni lettori ci chiedono, nelle loro lettere, come sottoporre i loro programmi a MC. È semplicissimo: registrate i vostri lavori su cassetta o disco (se il programma è proprio molto corto può bastare il semplice listato; certo, la cassetta non guasta mai...), corredati dell'opportuna documentazione e spedite il tutto alla redazione, indicando magari sulla busta la rubrica interessata.

Tutti i programmi che arrivano sono esaminati ed i migliori pubblicati.

Purtroppo non possiamo restituire, per ragioni organizzative, il materiale che ci viene inviato, anche in caso di mancata pubblicazione.

Ricordatevi che migliore è la documentazione, maggiore è la possibilità che il vostro lavoro venga pubblicato: spiegate quindi chiaramente il funzionamento del programma ed accludete tutto quello che pensate possa essere utile (elenco variabili e via dicendo). Soprattutto non dimenticate di indicare il vostro nome ed indirizzo (qualche volta succede!) e, se possibile, il numero telefonico.

Ah, quasi dimenticavamo: naturalmente è previsto un compenso, che varia normalmente tra le 30 e le 100.000 lire, a seconda della qualità del lavoro inviato.

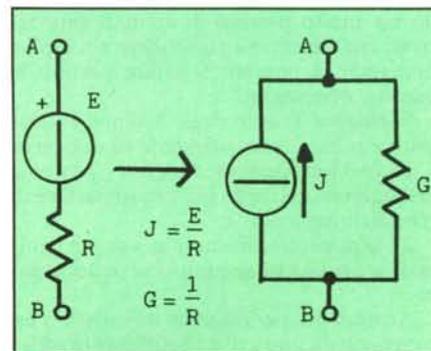


Figura 1 - Esempio di trasformazione di un ramo con una resistenza in serie ad un generatore di tensione nell'equivalente formato da una conduttanza in parallelo ad un generatore di corrente.

Nell'affrontare il problema la prima idea che viene in mente è quella di risolvere il circuito assumendo come incognite le intensità di corrente nei vari rami e applicando poi i due principi di Kirchoff per ricavare un sistema lineare di l equazioni in l incognite I_1, I_2, \dots, I_l (Il I° principio afferma che ad ogni istante la somma delle correnti entranti in un nodo è nulla $i_1 + i_2 + \dots + i_k = 0$ con rif. al nodo 1; il secondo principio afferma che — detto $A_1, A_2, \dots, A_m, A_1$ — un insieme ciclicamente ordinato di nodi della rete — in ogni istante $V_{A_1A_2} + V_{A_2A_3} + \dots + V_{A_mA_1} = \mathcal{E}$).

In questo modo però ci si scontra subito con un ostacolo particolarmente difficile da superare: e cioè quello di trovare tutte le maglie elementari o comunque indipen-

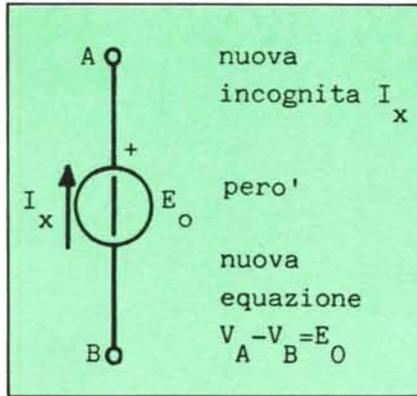


Figura 2 - Nel caso di rami con resistenza uguale a zero occorre aggiungere al sistema una nuova incognita ed una nuova equazione.

denti esistenti nella rete. Questo problema si presenta anche nella risoluzione con carta e penna quando la rete viene presentata in una configurazione spaziale e, se intendiamo per piana una rete che può essere distesa su un piano senza che si intersechino i lati che ne fanno parte, si può dimostrare che solo le reti con non più di 4 nodi sono sicuramente piane.

Non resta quindi che aggirare l'ostacolo del secondo principio di Kirchoff utilizzando quello che viene comunemente detto

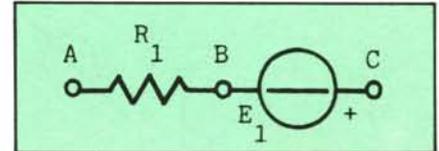


Figura 3 - È possibile calcolare il potenziale di punti del circuito che non sono dei nodi reali.

“metodo dei potenziali ai nodi”. Si assumono come incogniti i potenziali di $n-1$ nodi della rete scelto che sia l' n -mo come nodo di riferimento a potenziale \emptyset . Il sistema di $n-1$ equazioni viene ricavato utilizzando il solo I° princ. previa trasformazione di ogni ramo contenente un generatore di tensione in serie ad una resistenza nel corrispondente generatore di corrente in parallelo ad una conduttanza come in figura 1. Il metodo non risulta applicabile

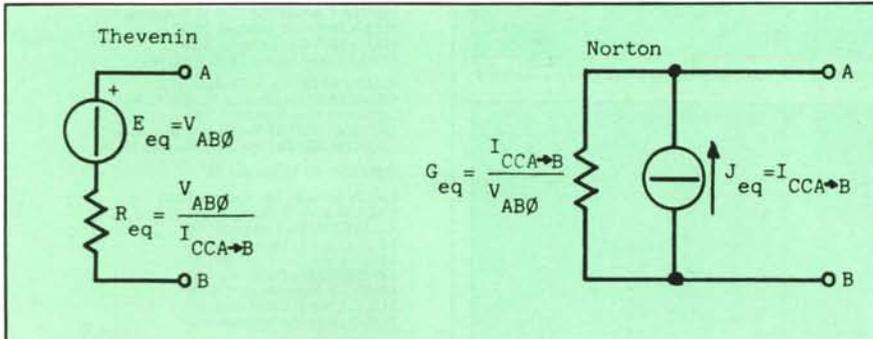


Figura 4 - Rami equivalenti secondo Thevenin e secondo Norton, e relative equazioni.

```

100 REM *****
110 REM **
120 REM ** CIRCUITI ELETTRICI
130 REM **
140 REM ** IN
150 REM **
160 REM ** CORRENTE CONTINUA
170 REM **
180 REM ** RENZO DEL FABBRO
190 REM **
200 REM ** PIEVE DI CADORE
210 REM **
220 REM ** 19 / 09 / 1983
230 REM **
240 REM *****
250 DATA 4,7,10,15,20,25
260 HOME
270 :
280 REM ** INIZIALIZZAZIONI
290 :
300 T = 1:LA = 30:
310 INPUT "NUMERO DI PUNTI BASE =" :NN
320 PRINT
330 DIM A(NN + 10, NN + 11), C(NN + 11), R(NN + 10), E(NN + 10, NN + 11)
340 DIM RE(LA), FLA), X(LA), S(LA), I(LA), G(LA), P(LA, 2)
350 IF T ( ) 1 THEN 390
360 PRINT " N A B E R " :
370 PRINT " J G I A B " :
380 PRINT "*****"
390 CALL - 95B: VTAB T + 4: HTAB ( T ) 9: + 2 * NOT ( T ) 9: PRINT T
400 :
410 REM ** INPUT DATI RAMO
420 :
430 FOR R = 1 TO 61 READ D: VTAB T + 4: HTAB D: INPUT D(R): NEXT
440 IF D(1) = D(2) THEN RESTORE : GOTO 390
450 P(T, 1) = D(1):P(T, 2) = D(2):F(T) = D(3):RE(T) = D(4):I(T) = D(5):G(T) =
D(6): GET A#: IF A# ( ) "S" AND A# ( ) "C" THEN T = T + 1: RESTORE
: GOTO 390
460 IF A# = "C" THEN RESTORE : GOTO 430
470 : RESTORE
480 FOR R = 1 TO T: FOR E = 1 TO 61 READ D: VTAB D + 5: HTAB D: PRINT " "
: NEXT : RESTORE : G = G + 1: NEXT
490 :
500 REM ** FORMAZIONE MATRICE
510 :
520 S = 0: FOR N = 1 TO T: S(N) = ((F(N) ( ) 0) * (RE(N) = 0) * (G(N) = 0)
+ (F(N) = 0) * (RE(N) = 0) * (G(N) = 0) * (I(N) = 0)) * N
530 IF S(N) ( ) 0 THEN S = S + 1
540 NEXT :NO = NN + S
550 H = 0
560 FOR N = 1 TO T: W = P(N, 1): V = P(N, 2)
570 IF S(N) ( ) 0 THEN 600
580 IF G(N) ( ) 0 THEN RE(N) = 1 / G(N)
590 IF RE(N) ( ) 0 THEN G(N) = 1 / RE(N)
600 IF F(N) ( ) 0 THEN I(N) = F(N) * G(N)
610 IF I(N) ( ) 0 THEN F(N) = I(N) * RE(N)
620 G = G(N)
630 E(W, W) = E(W, W) + G
640 E(W, V) = E(W, V) - G: E(V, W) = E(V, W)
650 E(V, V) = E(V, V) + G
660 E(V, NO) = E(V, NO) + I(N): E(W, NO) = E(W, NO) - I(N)
670 GOTO 710
680 REM
690 H = H + 1: Z = NN + H - 1
700 E(W, Z) = 1: E(V, Z) = - 1: E(Z, W) = - 1: E(Z, V) = 1: E(Z, NO) = F(N)
710 NEXT
720 IF H = 0 THEN Z = NN - 1
730 FOR I = 1 TO Z: FOR J = 1 TO NO: A(I, J) = E(I, J): NEXT : NEXT
740 DOSUB 910
750 DE = 0
760 FOR N = 1 TO T: W = P(N, 1): V = P(N, 2)
770 IF S(N) ( ) 0 THEN X(N) = V(NN + DE): DE = DE + 1: GOTO 810
780 IF I(N) ( ) 0 AND RE(N) = 0 THEN X(N) = I(N): GOTO 810
790 X(N) = - (V(V) - V(W) - F(N)) * G(N)
800 VTAB N + 4: HTAB 30: PRINT "
810 VTAB N + 4: ZX = INT (X(N) * 10000) / 10000: LN = LEN ( STR# ( INT ( ABS
(ZX))) ) + (ZX ( ) - (ZX) - 1 AND ZX ( 1): HTAB 33 - LN: PRINT ZX
820 NEXT
830 CALL - 95B
840 FOR U = 1 TO NN - 1: Y = T + 6 + U: XT = 1: IF T + 6 + U > 23 THEN Y =
Y - (23 - T - 6): XT = 20
850 VTAB Y: HTAB XT: PRINT " " : HTAB XT: PRINT " " : V(U): NEXT
860 VTAB 24: HTAB 1: PRINT " * VUOI INSERIRE UN NUOVO RAMO ?(1/0): " : GET C
: IF C# = "1" THEN T = T + 1
870 IF Y = 23 AND XT = 20 THEN VTAB 24: HTAB 1: INVERSE : PRINT "*****
NON CI STA " : NORMAL : GET A#: END
880 GOTO 390
890 END
900 :
910 REM ** CALCOLO SISTEMA *****
920 :
930 N = Z
940 FOR K = 1 TO N
950 P = 0
960 FOR X = 1 TO N
970 IF R(X) ( ) 0 THEN 1030
980 FOR Y = 1 TO N
990 IF C(Y) ( ) 0 THEN 1020
1000 IF ABS (A(X, Y)) ( P THEN 1030
1010 P = ABS (A(X, Y)): IP = X: JP = Y
1020 NEXT Y
1030 NEXT X
1040 IF P < .000001 THEN PRINT : PRINT : PRINT : PRINT : PRINT : PRINT :
PRINT : PRINT "*****" : STOP
INDETERMINATO " : STOP
1050 C(JP) = IP: R(IP) = JP
1060 FOR J = 1 TO N + 1
1070 IF C(J) ( ) 0 THEN 1090
1080 A(IP, J) = A(IP, J) / A(IP, JP)
1090 NEXT J
1100 FOR I = 1 TO N: IF I = IP THEN 1160
1110 FOR J = 1 TO N + 1
1120 IF C(J) ( ) 0 THEN 1140
1130 A(I, J) = A(I, J) - A(I, JP) * A(IP, J)
1140 S = S
1150 NEXT J
1160 NEXT I
1170 NEXT K
1180 FOR W = 1 TO N:
1190 V(W) = A(C(W), N + 1): NEXT
1200 FOR GH = 1 TO N + 1: R(GH) = 0: C(GH) = 0: NEXT
1210 IP = 0: JP = 0: G = 0
1220 FOR I = 1 TO N: FOR J = 1 TO N + 1: E(I, J) = 0: NEXT : NEXT
1230 RETURN
1240 END
    
```

La più grande catena di computer in Europa.

AGRATE BRIANZA Via G. Matteotti, 99
 ALBA Via Paruzza, 2
 ALESSANDRIA Via Savonarola, 13
 ANCONA Via De Gasperi, 40
 AOSTA Av. Conseil Des Commis, 16
 BELLANO Via Martiri della Libertà, 14
 BASSANO DEL GRAPPA Via Jacopo Da Ponte, 51
 BERGAMO Via S. F. D'Assisi, 5
 BIELLA Via Italia, 50A
 BOLOGNA Via Brugnoli, 1
 BRESCIA Via B. Croce, 11/13/15
 BUSTO ARSIZIO Via Gavinana, 17

CAGLIARI Via Zagabria, 47
 CALTANISSETTA Via R. Settimo, 10
 CAMPOBASSO Via Mons. Il Bologna, 10
 CATANIA Via Muscatello, 6
 CATANZARO Via XX Settembre, 62 A/B/C
 CESANO MADERNO Via Ferrini, 6
 CESENA Via Elli Spazzoli, 239
 CINISELLO BALSAMO V.le Matteotti, 66
 COLICO P.za Cavour, 24
 COMO Via L. Sacco, 3
 CONEGLIANO V.le Italia, 128
 CREMA Via IV Novembre, 56/58
 CUNEO C.so Nizza, 16

EMPOLI Via Masini, 32
 FAVRIA CANAVESE C.so G. Matteotti, 13
 FIRENZE Via G. Milanese, 28/30
 FIRENZE Via Centostelle, 5/B
 FORLÌ P.zza Melozzo Degli Ambrogi, 6

GALLARATE Via A. Da Brescia, 2
 GENOVA Via Domenico Fiasella, 51/R
 GENOVA C.so Gastaldi, 77/R
 GENOVA-SESTRI Via Chiaravagna, 10/R
 GENOVA-SESTRI Via Ciro Menotti, 136/R
 IMPERIA Via Delbecchi, 32

LATINA Via E. Toti (Galleria Cisa)
 L'AQUILA Via Strinella, 20/A
 LA SPEZIA Via Lumigniana, 481
 LEGGICO Via L. Da Vinci, 7
 LEGNANO C.so Garibaldi, 82
 LIVORNO Via Paoli, 32
 LODI V.le Rimembranze, 36/B
 LUCCA Via S. Concordio, 160
 LUGO (RA) Via Magnapassi, 26

MACERATA Via Spalato, 126
 MANTOVA Via Cavour, 69
 MESSINA Via Del Vespro, 71
 MILANO Via Altavanguardia, 2
 MILANO Via G. Cantoni, 7
 MILANO Via E. Petrella, 6
 MILANO Galleria Manzoni, 40
 MIRANO-VENEZIA Via Gramsci, 40/54
 MODENA Via Fonteraso, 18
 MONZA Via Azzone Visconti, 39
 MORBEGNO Via Fabiani, 31

NAPOLI Via Luigia Sanfelice, 7/A
 NAPOLI C.so Vittorio Emanuele, 54
 NAPOLI Via Luca Giordano, 40/42
 NOVARA Via Perazzi, 23/B

PADOVA Via Fistomba, 8 (Stanga)
 PADOVA Via Piovese, 37
 PALERMO Via Libertà, 191
 PALERMO Via Notarbartolo, 23 B/C
 PARMA Via Imbrani, 41
 PAVIA Via C. Battisti, 4/A
 PERUGIA Via R. D'Andreotto, 49/55
 PESCARA Via Conte di Ruvo, 134
 PESCARA Via Trieste, 73
 PIACENZA Via IV Novembre, 60
 PISA Via Emilia, 36

PISA Via XXIV Maggio, 101
 PISTOIA V.le Adua, 350
 POMEZIA Via Roma, 39
 POTENZA Via G. Mazzini, 72
 POZZUOLI Via G.B. Pergolesi, 13
 PRATO Via E. Boni, 76/78
 RECCO Via B. Assereto, 78
 REGGIO CALABRIA Via S. Marco, 8/B
 RIMINI Via Bertola, 75
 ROMA P.zza San Donà di Piave, 14
 ROMA Via Cerreto Da Spoleto, 23
 ROMA Via G. Villani, 24-26

S. DONÀ DI PIAVE P.zza Rizzo, 61
 SALERNO C.so Garibaldi, 56
 SANREMO Via S. Pietro Agosti, 54/56
 SASSUOLO P.zza Martiri Partigiani, 31
 SESTO CALENDE Via Matteotti, 38
 SENIGALLIA Via Maierini, 10
 SIRACUSA Viale Scala Greca, 339/9
 SONDRIO Via N. Sauro, 28

TERMOLI Via Martiri della Resistenza, 88
 TORINO C.so Grosseto, 209
 TORINO Via Tripoli, 179
 TORINO Via Nizza, 91
 TRENTO Via Sighele, 7/1
 TREVISO Via IV Novembre, 13A
 TRIESTE Via Fabio Severo, 138
 TRIESTE Via Torrebianca, 18
 TRIESTE Via Paolo Reti, 6
 UDINE Via Tavagnacco, 89/91

VARESE Via Carrobbio, 13
 VENEZIA Cannaregio, 5898
 VERCELLI Via Dionisotti, 18
 VIAREGGIO Via A. Volta, 79
 VICENZA Via del Progresso, 7/9
 VIGEVANO C.so V. Emanuele, 82
 VOGHERA P.zza G. Carducci, 11

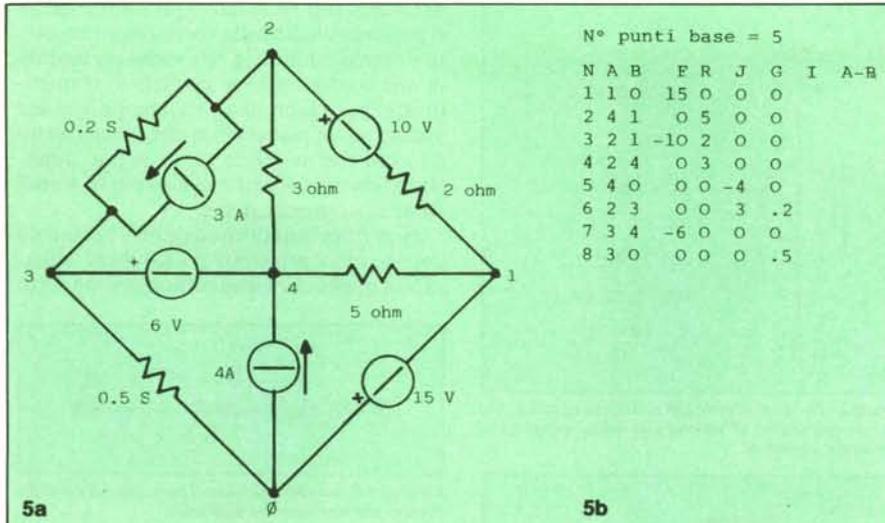


Figura 5 - Esempio di circuito in esame ed elenco dei nodi. I punti base sono stati numerati arbitrariamente da 0 a 4.

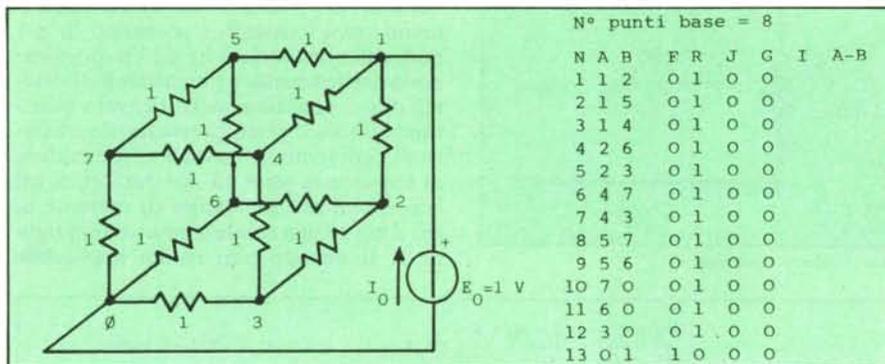


Figura 6 - Il classico problema del cubo di resistenze da un ohm si può risolvere facilmente una volta posto un generatore di tensione da 1 volt tra due spigoli opposti. La corrente risultante equivale all'inverso della resistenza totale.

quando $R=0$ e in questo caso bisogna introdurre una nuova incognita e cioè la corrente in quel ramo (I_x) e una nuova equazione $V_A - V_B = E_0$ nel sistema figura 2 (pag. 109).

Tutte queste operazioni vengono eseguite dal programma che inizia con la richiesta del numero di punti base. Questo numero può essere uguale o maggiore del numero dei nodi in quanto è possibile calcolare il potenziale di punti del circuito che non sono dei veri e propri nodi (vedi ad es. B in fig. 3, pag. 109).

Il programma prosegue con la richiesta dei dati relativi ad ogni ramo componente la rete e precisamente:

- A = n° punto base da cui parte
- B = n° punto base su cui arriva
- E = f.e.m. del generatore di tensione (positiva se con il + verso B)
- R = resistenza del ramo
- J = intensità del generatore di corrente (positivo se verso B)
- G = conduttanza del ramo

Bisogna battere RETURN fra un dato e l'altro, \emptyset se non esiste la grandezza richiesta. In caso di errore completare la riga e, una volta a capo, battere C. Esauriti tutti i rami si batte S.

Dopo un tempo dipendente dal numero di nodi compariranno sullo schermo, a destra, le correnti nei rami con verso conven-

zionale da A a B e sulla parte inferiore i potenziali dei nodi. Al termine si possono inserire uno o più nuovi rami nella rete originaria rispondendo affermativamente alla richiesta del programma.

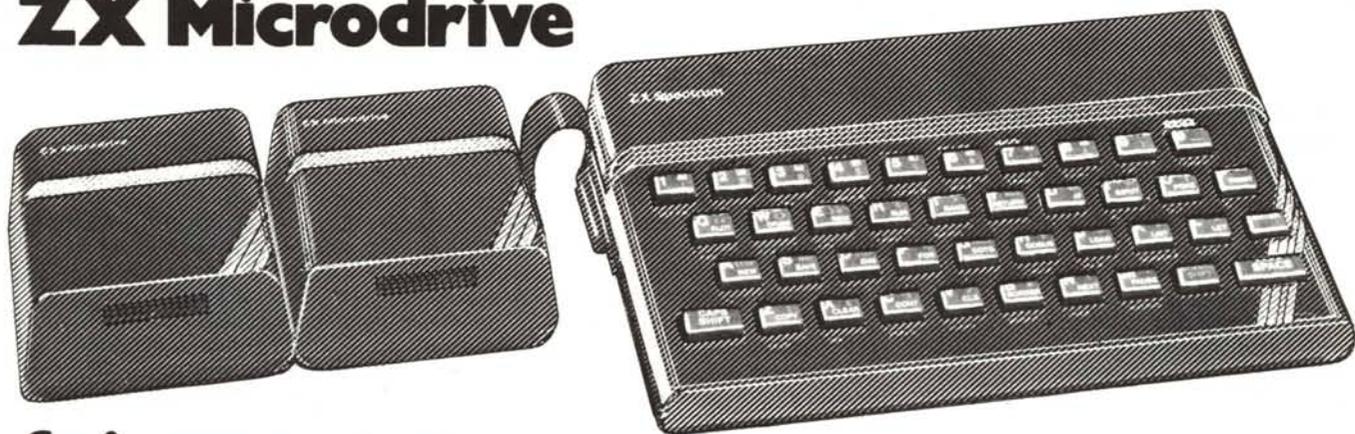
Questa opzione può risultare utile allorché si desidera calcolare il generatore equivalente fra due qualsivoglia punti (ad esempio A e B) della rete.

Ricavando infatti la differenza di potenziale a vuoto fra i punti in oggetto ed inserendo successivamente un ramo di corto circuito fra gli stessi, si viene in possesso delle due grandezze fondamentali $V_{AB\emptyset}$ e $I_{CCA \rightarrow B}$ e di conseguenza del generatore equivalente secondo Thevenin o secondo Norton (fig. 4, pag. 109).

Il circuito viene inserito senza alcuna trasformazione o calcolo preventivo così come lo si vede dal suo schema elettrico. Per chiarimento in figura 5b è riportata la tabella relativa all'inserimento dei dati del circuito di figura 5a così come si presenta sul monitor.

Come ulteriore applicazione si può infine calcolare la resistenza vista fra due spigoli opposti di un cubo di resistenze tutte uguali da 1 ohm; a tale scopo si inserisce un generatore ideale di tensione fra detti spigoli e si esegue il rapporto con la corrente da esso erogata fornita dal programma.

ZX Microdrive



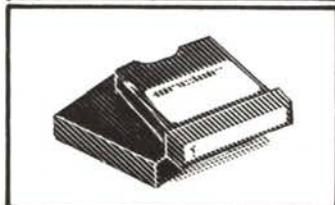
**fai crescere
il tuo Sinclair - Spectrum
con le sue eccezionali periferiche!**



ZX MICRODRIVE

Amplia le possibilità dello ZX Spectrum in quei settori come la didattica e le piccole applicazioni gestionali, dove è necessaria una ricerca veloce delle informazioni. Ogni cartuccia può contenere:

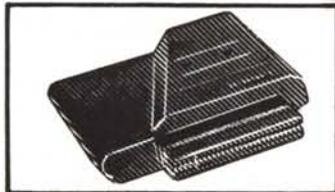
85 kbyte / 95 kbyte



CARTRIDGE

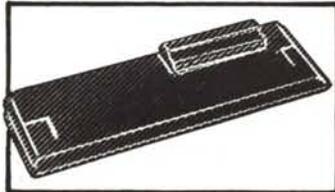
Per ZX Microdrive. Capacità: 85 kbyte / 95 kbyte. Confezione da 2 pezzi.

La coppia



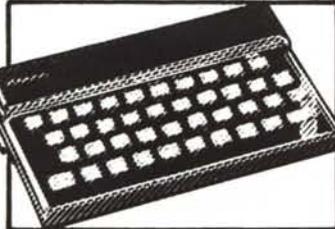
ZX INTERFACE 2

Permette di utilizzare le nuovissime ZX ROM, cartucce software e il collegamento per 2 joystick.



ZX INTERFACE 1

Indispensabile per il collegamento dello ZX Microdrive. Inoltre permette il collegamento fra lo ZX Spectrum e una ampia gamma di periferiche e di altri Sinclair in rete locale.



COMPUTER ZX SPECTRUM

A colori, collegabile ad un televisore a colori o in b/n e ad un normale registratore a cassetta. 32x24 caratteri. RAM di base: 16 k - 48 k. 256x192 punti. 8 colori - 2 luminosità.



L. 199.500

L. 45.000

L. 95.500

L. 199.500

16 K
L. 398.000

48 k
L. 499.000

I prezzi sono comprensivi di I.V.A.

**competenza
in COMPUTER
sinclair**