

Calcolo del codice fiscale

Stefano Reolon-Belluno

Spett.le MCmicrocomputer desidero presentare un programma per la determinazione del Codice Fiscale, secondo le disposizioni che si trovano nel Decreto Ministeriale 23/12/1976.

L'idea mi è venuta per provare le nuove possibilità offerte dal modulo di estensione di funzioni e memoria, nel trattare le stringhe ALPHA. E in particolare la funzione "ATOX", che dà, di ogni carattere, il corrispondente valore numerico secondo lo standard ASCII. Prima di esaminare il programma, diciamo com'è fatto il Codice Fiscale: è costituito da 16 caratteri alfanumerici, dei quali 15 derivano dai dati anagrafici e il sedicesimo ha funzione di controllo.

I primi 3 caratteri alfabetici derivano dal cognome; il secondo gruppo di 3 caratteri deriva dal nome. Ci sono poi 5 caratteri che hanno origine dalla data di nascita e dal sesso. Seguono poi i 4 caratteri indicativi del comune o stato estero di nascita (rilevabili dai volumi: "Codice dei Comuni d'Italia" e

"Codice degli Stati Esteri", redatti a cura del Catasto e dei Servizi Tecnici Erariali). Il sedicesimo carattere è di controllo e si ottiene assegnando opportuni valori numerici ai precedenti caratteri.

Per una spiegazione più dettagliata, si rimanda al già citato Decreto Ministeriale.

Esaminiamo ora il programma. Nella prima parte (fino alla linea 39) avviene l'inizializzazione, poi la richiesta e relativa memorizzazione dei dati necessari. Viene creato un File ASCII (di nome "W") per memorizzare il cognome ed il nome, e acceso il Flag 00, se il soggetto è femminile.

C'è poi la parte di programma che estrae dal cognome e dal nome le consonanti o le vocali, come richiesto dal procedimento di codifica, e che comprende i passi 40-65 con le relative subroutine ai passi 159-223.

Per riconoscere in modo abbastanza rapido se una lettera è una consonante o una vocale, il programma opera in questo modo: ricava il valore numerico della lettera in questione con "ATOX", accende il Flag 25 (per ignorare un'operazione impropria) e poi ricerca, con "GTO IND X", una "LBL" con quel valore: se trova l'etichetta (sono le

LBL 65 = A, LBL 69 = E, ecc) vuol dire che la lettera in questione è una vocale; in caso contrario il programma prosegue con l'istruzione seguente a "GTO IND X", senza fermarsi con l'indicazione "NONEXISTENT", essendo acceso il flag 25. In questa fase sono usati anche i Flag dall'uno al quattro.

Ricavate le lettere di codificazione del cognome e del nome, i rispettivi codici ASCII sono memorizzati nei primi 6 registri di memoria. Quindi il programma trasforma la data di nascita e memorizza i valori corrispondenti nei registri da R 06 a R 10 (aggiungendo 40 al giorno di nascita, se il soggetto è femminile). Nei registri da R 11 a R 14 sono memorizzati i valori ASCII relativi al codice del luogo di nascita.

Il carattere di controllo è calcolato in questo modo: si dà un valore numerico ai precedenti caratteri, a seconda che questi occupino posizione pari o dispari (ci sono delle apposite tabelle nel Decreto Ministeriale); si esegue la somma di questi valori e si divide per 26; il resto della divisione (funzione "MOD") dà il carattere in base ad un'altra tabella.

Calcolo del codice fiscale

```

01*LBL "CODFISC" 46 ISG 16          91 *                136*LBL 17          181 STO 02          226 XEQ 01
02 0             47 GTO 05          92 XEQ 13          137 ISG 16          182 GTO 07          227 FS? 03
03 X<Y?         48*LBL 07          93 STO 06          138 GTO 15          183*LBL 04          228 XEQ 02
04 SIZE?       49 SF 03           94 ATOX            139 RCL 15          184 ATOX            229 RCL 16
05 17           50 XEQ 02           95 STO 07          140 26              185 STO 03          230 4
06 XYY?       51 XEQ 03           96 0               141 MOD            186 ATOX            231 XYY?
07 PSIZE       52 3,005         97 STO 15          142 65              187 STO 04          232 GTO 11
08*LBL 20      53 STO 16          98 1,01302         143 +               188 00              233 FS? 04
09 *M*         54 XEQ 06           99 STO 16          144 STO 15          189 STO 05          234 GTO 11
10 5           55 ISG 16          100*LBL 14         145 CLA             190 GTO 08          235 SF 04
11 CRFLAS     56 FC? 04          101 17             146 ,002            191*LBL 06          236 CF 01
12 *COGNOME*   57 SF 01            102 RCL IND 16     147 XEQ 18          192 0               237 GTO 07
13 AOH        58 XEQ 06           103 65             148 3,005           193 ATOX            238*LBL 11
14 PROMPT     59 ISG 16          104 -               149 XEQ 18          194 X=Y?            239 XEQ 09
15 APPREC     60 XEQ 06           105 X<0?          150 6,01            195 GTO 10          240*LBL 12
16 *NOME*     61*LBL 08           106 +               151 XEQ 18          196 32              241 ISG 16
17 PROMPT     62 CF 03            107 ST+ 15        152 11,015          197 X=Y?            242 GTO 11
18 APPREC     63 CF 04            108 ISG 16        153 XEQ 18          198 GTO 06          243 FC? 03
19 70         64 *M*             109 GTO 14         154 ADV             199 X<Y?            244 GTO 07
20 *SESSO M-F* 65 PURFL           110 ,01502        155 AVIEW           200 SF 25           245 FS? 03
21 PROMPT     66 RCL 06           111 STO 16         156 ADV             201 GTO IND X       246 GTO 08
22 ATOX       67 FRC             112*LBL 15        157 STOP            202 FS?C 01         247*LBL 13
23 X=Y?       68 1 E2             113 17             158 GTO 20          203 GTO 06          248 10
24 SF 00      69 *             114 RCL IND 16    159*LBL 01          204 STO IND 16     249 X<Y?            250 GTO 19
25 OFF        70 INT             115 65             160 SF 05           205 RTN              251 48
26 *DATA NASCITA* 71 *TABCODEHLMPRS* 116 -              161 0               206*LBL 65          252 XTOA
27 PROMPT     72 AROT           117 X<0?          162*LBL 02          207*LBL 69          253 PBN
28 STO 06     73 ATOX           118 +              163 FC?C 05         208*LBL 73          254*LBL 19
29 *COD. COMUNE* 74 STO 08          119 1              164 1               209*LBL 79          255 X<Y?
30 AOH        75 40             120 X<Y?          165 *M*            210*LBL 05          256 ARCL X
31 PROMPT     76 CLA             121 X=0?          166 SEEKPTR         211 FC?C 02         257 ATOX
32 OFF        77 RCL 06           122 GTO 16        167 GETREC          212 GTO 06          258 RTN
33 11,014     78 INT             123 X=Y?          168 RTN             213 STO IND 16     259*LBL 18
34*LBL 00     79 FS?C 00           124 GTO 17        169*LBL 03          214 GTO 12          260 RCL IND X
35 ATOX       80 *             125 *YXFHJMPRTVCESUL* 170 ALENG           215*LBL 09          261 XTOA
36 STO IND Y  81 XEQ 13           126 *FDGIMQAKWZ* 171 2               216 SF 02           262 X<Y?
37 X<Y?       82 STO 09           127 AROT           172 X=Y?            217 32              263 ISG X
38 ISG X      83 ATOX           128 ATOX           173 RTN             218 ATOX            264 GTO 10
39 GTO 00     84 STO 10           129 65            174 FS? 07          219 X=Y?            265 32
40 XEQ 01     85 CLA             130 -              175 GTO 04          220 GTO 09          266 XTOA
41 XEQ 03     86 RCL 06           131 ST+ 15        176 ATOX            221 SF 25           267 END
42 ,002       87 1 E4             132 GTO 17        177 STO 00          222 GTO IND X
43 STO 16     88 *             133*LBL 16        178 ATOX            223 GTO 09          224*LBL 10
44*LBL 05     89 FRC             134 X<Y?          179 STO 01          225 FC? 03
45 XEQ 06     90 1 E2             135 ST+ 15
    
```

Esempio

```

XEQ "CODFISC"
COGNOME
ROSSI          RUN
NOME
GIUSEPPE      RUN
SESSO: M-F
M              RUN
DATA NASCITA
4,101933      RUN
COD. COMUNE
A757          RUN
RSS GPP 33R04 A757Y
    
```

Per concludere, il Codice Fiscale è costruito nel registro ALPHA tramite la funzione "XTOA" e l'inserimento di alcuni spazi opportuni.

Come usare il programma:

a) controllare, per prima cosa, di avere almeno 5 registri liberi nell'estensione di memoria per la creazione del File ASCII e 17 registri nella memoria principale (l'allocation è automatica);

b) avviare il programma con XEQ "CODFISC";

c) impostare il cognome ed il nome alle rispettive richieste;

d) per il sesso, premere "F", se femminile, "M" o qualsiasi altro carattere, se maschile;

e) alla richiesta della data di nascita, impostarla nella forma GG,MMaaaa (giorno, mese-anno);

f) infine il codice del luogo di nascita (in modalità ALPHA);

g) il Codice Fiscale è visualizzato nel reg. ALPHA;

h) per un altro caso, battere R/S e si torna al punto c).

L'elaborazione è un po' lunga, mediamente da 1' a 1'30"; purtroppo il programma deve eseguire molti confronti, cicli e chiamate di subroutine, inoltre la funzione "ATOX" (usata parecchie volte) ha una durata di esecuzione notevole; in media 0,14 sec. contro i 0,04 sec. di "XTOA"; i tempi aumentano se è collegata la stampante.

Occorrerebbe riesaminare il programma per vedere se è possibile renderlo un po' più spedito, magari eliminando alcune subroutine a scapito però dell'occupazione di memoria. Purtroppo è nato abbastanza in fretta per esigenze di lavoro, e non mi è stato possibile migliorarlo.

Un complimento per l'ottima rivista, che dà spazio anche a queste "utilitarie" nel vasto campo dell'elaborazione dati.

Il programma funziona bene: ho ricavato il codice fiscale di molte persone ottenendo sempre lo stesso codice già fornito loro dal Ministero delle Finanze; del resto con il codice fiscale non si scherza, e quindi un funzionamento impeccabile del programma è il minimo che si possa pretendere. Unico appunto può essere fatto alla forma usata per l'output quando la stampante non è collegata; infatti la 41C fornisce la risposta, lunga più di dodici caratteri, "scrollando" sul display dopo una fugace apparizione, i primi sei caratteri del codice fiscale; per poter rileggere di nuovo la stringa è necessario premere il tasto "ALPHA", in modo da far scorrere ancora i caratteri per intero. Non credo ci sia da aggiungere altro, se non un "complimenti!" diretto al bravo Stefano Reolon.

Conversione numeri romani - numeri arabi e viceversa

Marco Merendi - Faenza (RG)

Spettabile redazione di MCmicrocomputer, vi invio due programmi per HP-41C sperando che possano essere utili a qualche utilizzatore. Il primo riguarda la conversione dei numeri in cifre arabe nei loro equivalenti in cifre romane, mentre il secondo svolge il compito inverso.

Il programma "R-A" (da arabo a romano), converte numeri compresi tra 1 e 3999; il pregio del programma è indubbiamente la sua velocità (da un secondo a sei secondi per i numeri più lunghi).

Il programma "A-R" (da romano ad arabo) è rivolto ai possessori del modulo "extended functions" e fa uso della funzione "ATOX".

Sperando di non avervi fatto perdere tempo e congratulandomi per l'eccellente qualità della rivista sotto ogni aspetto, vi auguro buon lavoro e vi invio i miei più distinti saluti.

Conversione numeri romani - numeri arabi

01*LBL "A-R"	32 XEQ IND X	63 RTN	94*LBL 12	125 *CCC*	01*LBL "R-A"	32*LBL 73
02*LBL 50	33 ASTO 03	64*LBL 01	95 *XX*	126 RTN	02 *WAIT..*	33 I E
03 *N?*	34 CLA	65 *I*	96 RTN	127*LBL 24	03 ASTO 02	34 RTN
04 PROMPT	35 RCL 00	66 RTN	97*LBL 13	128 *C*	04*LBL 03	35*LBL 86
05 CLA	36 X=0?	67*LBL 02	98 *XXX*	129 RTN	05 *N?*	36 5
06 4 E3	37 GTO 42	68 *II*	99 RTN	130*LBL 25	06 AOH	37 RTN
07 X<Y?	38 30	69 RTN	100*LBL 14	131 *D*	07 STOP	38*LBL 88
08 GTO 50	39 +	70*LBL 03	101 *XL*	132 RTN	08 AOFF	39 I E1
09 X<Y	40 XEQ IND X	71 *III*	102 RTN	133*LBL 26	09 VIEW 02	40 RTN
10 STO 00	41*LBL 42	72 RTN	103*LBL 15	134 *BC*	10 ATOX	41*LBL 76
11 X=0?	42 ARCL 03	73*LBL 04	104 *L*	135 RTN	11 XEQ IND X	42 50
12 GTO 50	43*LBL 41	74 *IV*	105 RTN	136*LBL 27	12 STO 01	43 RTN
13 XEQ A	44 ARCL 02	75 RTN	106*LBL 16	137 *DCC*	13 STO 00	44*LBL 67
14 XEQ IND X	45*LBL 40	76*LBL 05	107 *LX*	138 RTN	14*LBL 01	45 I E2
15 ASTO 01	46 ARCL 01	77 *V*	108 RTN	139*LBL 28	15 ATOX	46 RTN
16 CLA	47 PROMPT	78 RTN	109*LBL 17	140 *DCCC*	16 XEQ IND X	47*LBL 68
17 RCL 00	48 GTO 50	79*LBL 06	110 *LXX*	141 RTN	17 RCL 01	48 5 E2
18 X=0?	49*LBL A	80 *VI*	111 RTN	142*LBL 29	18 X<Y?	49 RTN
19 GTO 40	50 10	81 RTN	112*LBL 18	143 *CM*	19 GTO 02	50*LBL 77
20 XEQ A	51 /	82*LBL 07	113 *LXXX*	144 RTN	20 PDN	51 I E3
21 10	52 INT	83 *VII*	114 RTN	145*LBL 31	21 ST+ 00	52 RTN
22 +	53 STO 00	84 RTN	115*LBL 19	146 *M*	22 STO 01	53*LBL 00
23 XEQ IND X	54 LASTX	85*LBL 08	116 *XC*	147 RTN	23 GTO 01	54 *N=*
24 ASTO 02	55 FRC	86 *VIII*	117 RTN	148*LBL 32	24*LBL 02	55 FIX 0
25 CLA	56 10	87 RTN	118*LBL 21	149 *MM*	25 ST- 00	56 CF 29
26 RCL 00	57 *	88*LBL 09	119 *C*	150 RTN	26 PDN	57 ARCL 00
27 X=0?	58 RTN	89 *IX*	120 RTN	151*LBL 33	27 STO 01	58 PROMPT
28 GTO 41	59*LBL 00	90 RTN	121*LBL 22	152 *MMM*	28 R/	59 GTO 03
29 XEQ A	60*LBL 10	91*LBL 11	122 *CC*	153 END	29 -	60 END
30 20	61*LBL 20	92 *X*	123 RTN		30 ST+ 00	
31 +	62*LBL 30	93 RTN	124*LBL 23		31 GTO 01	

Esempio

```

N?          XEQ "A-R"
            1983  RUN
            MCMLXXXIII
            XEQ "R-A"
            MMXLI  RUN
            "WAIT.."
            N=2041
  
```

Esistono programmi per convertire numeri decimali in ottali, esadecimali e binari, essi consentono di lavorare, nel mondo dei calcolatori, in un linguaggio ad essi più adatto; forse questi due programmi servono per lavorare con i calcolatori del tempo di Giulio Cesare...

Scherzi a parte, i due programmi sono senz'altro interessanti; vediamo di esaminarne brevemente il funzionamento.

Il primo (A-R) scompone il numero nelle sue quattro cifre e costruisce il corrispondente in simboli romani, assemblando semplicemente i vari "pezzetti" corrispondenti a ciascuna cifra decimale, disponibili ai passi da 59 a 153 (le LBL 00, 10, 20 e 30 corrispondono a cifre nulle). Il secondo (R-A) anche se più breve del primo, è forse più elegante. Il programma, facendo uso dell'istruzione "ATOX" (del modulo XFUNCTIONS), estrae di volta in volta il carattere più a sinistra del numero romano e richiama alle LBL 73, 86, 88, 76, 67, 68 e 77 il valore decimale corrispondente a ciascuno di essi; ogni volta che il programma esegue questa operazione, confronta il valore del carattere estratto con quello del precedente, se quest'ultimo è superiore, il

valore del carattere estratto viene accumulato nel registro R00, altrimenti viene sottratto da tale registro il valore del precedente e sommato, sempre in R00, la differenza tra i due valori.

Il valore del primo carattere estratto dopo l'introduzione del numero da convertire, è sommato inizialmente in R00 (passo 15) dopo che questo è stato azzerato (passi 05 e 06), volendo si possono eliminare i passi 05 e 06 dopo aver sostituito il passo 15 ST+00 con 15 STO 00.

Per chi non conosce la funzione "ATOX", essa, ogni volta che viene eseguita, fornisce sul registro X un valore numerico intero corrispondente, secondo le norme ASCII, al carattere più a sinistra della stringa ALPHA eliminandolo automaticamente; l'autore fa uso di tali numeri per richiamare poi indirettamente delle routine che forniscono in X il valore decimale corrispondente al carattere romano estratto da ALPHA; l'esecuzione termina quando il programma estrae da alpha un carattere nullo che causerà il trasferimento dell'esecuzione al passo 34 LBL 00 e quindi alla routine di output LBL 03. Volendo semplificare un po' le cose eliminando questo "doppio salto", è possibile sostituire i seguenti passi:

```

63 GTO 00 con 63 GTO 03
04 LBL 00 con 04 LBL 03
57 LBL 03 con 57 LBL 00
e quindi eliminare i passi 34 LBL 00 e 35
GTO 03. Per usare i due programmi, pre-
mere XEQ "R-A" oppure XEQ "A-R", a
seconda della conversione che si desidera,
alla domanda "N?" impostare il numero
da convertire e premere R/S; per converti-
re un altro numero si può premere di nuo-
vo R/S e la macchina tornerà a chiedere
"N?"
  
```