

# Verifica della correttezza del codice fiscale italiano

Claudio Beccari

GUI<sup>IT</sup><sub>meeting</sub> 2012

Scrivendo documenti fiscali, legali e simili è spesso necessario trascrivere i codici fiscali di persone fisiche.

La possibilità di commettere errori è elevata, quindi quando si scrive un documento con  $\text{\LaTeX}$  è comodo disporre di un programmino  $\text{\LaTeX}$  che esegua la verifica della correttezza del codice immesso o da immettere nel documento.

# Uso del pacchetto di verifica

Il pacchettino che presento qui esegue questa verifica;  
basta caricarlo con il solito comando:

```
\usepackage{codicefiscaleitaliano}
```

Si scriverà poi nel documento:

```
...  
Mario Rossi, codice fiscale  
\codicefiscaleitaliano{RSSMRA60D30L219K},  
...
```

o, meglio ancora:

```
...  
Mario Rossi, codice fiscale  
\codicefiscaleitaliano*{RSS MRA 60D30  
L219K},  
...
```

Se il codice fiscale è corretto viene scritto nel file `.log` che esso è “corretto”; se è sbagliato, un vistoso messaggio d'errore appare sullo schermo e il programma aspetta istruzioni dall'utente.

Se l'utente usa la variante asteriscata e il codice fiscale è “corretto”, il codice fiscale viene anche inserito nel documento in lavorazione; se non passa la verifica vengono prodotti dei messaggi d'errore, che possono riguardare il numero di caratteri nella stringa (senza contare gli spazi) oppure una inconsistenza fra la lettera di controllo e il resto del codice.

# Che cosa vuol dire “corretto”

- Ho messo fra virgolette la parola “corretto” perché questa correttezza à basata sulla consistenza della lettera di controllo (K nell’esempio precedente) non sulla consistenza dell’intero codice con i dati anagrafici della persona fisica.

# Che cosa vuol dire “corretto”

- Ho messo fra virgolette la parola “corretto” perché questa correttezza è basata sulla consistenza della lettera di controllo (K nell’esempio precedente) non sulla consistenza dell’intero codice con i dati anagrafici della persona fisica.
- Se si sbaglia ad introdurre il codice scambiando fra di loro due caratteri qualunque che occupino entrambi posizioni pari oppure dispari, questa verifica risulta “corretta” ma il codice introdotto è sbagliato.

# Che cosa vuol dire “corretto”

- Ho messo fra virgolette la parola “corretto” perché questa correttezza è basata sulla consistenza della lettera di controllo (K nell’esempio precedente) non sulla consistenza dell’intero codice con i dati anagrafici della persona fisica.
- Se si sbaglia ad introdurre il codice scambiando fra di loro due caratteri qualunque che occupino entrambi posizioni pari oppure dispari, questa verifica risulta “corretta” ma il codice introdotto è sbagliato.
- La probabilità di commettere errori di digitazione che si compensano è molto bassa; quindi la parola “corretto” ha un’alta probabilità di essere giusta.

Il codice del pacchetto è semplice, nel senso che bastano alcune decine di righe di programmazione.

Il codice è complicato perché bisogna ricorrere a **sporchi trucchi** che normalmente non sono descritti da nessuna parte, oppure sono adeguatamente nascosti alla vista di un lettore frettoloso.



Il codice fiscale è formato da una stringa alfanumerica strettamente ASCII di 16 caratteri di cui il sedicesimo rappresenta la lettera di controllo.

La lettera di controllo si determina attraverso un algoritmo che associa ad ogni lettera e ad ogni cifra del codice fiscale un valore numerico che dipende anche dalla posizione pari o dispari che quel carattere occupa nelle prime quindici posizioni.

Si sommano questi valori e si calcola il valore della somma **modulo 26** (cioè si calcola il resto della divisione intera fra la somma e 26). Questo resto (compreso fra 0 e 25) serve per ritrovare la lettera di controllo nella lista delle 26 lettere dell'alfabeto.

# I valori dei caratteri

Valori dei caratteri di posto pari							
0	0	A	0	K	10	U	20
1	1	B	1	L	11	V	21
2	2	C	2	M	12	W	22
3	3	D	3	N	13	X	23
4	4	E	4	O	14	Y	24
5	5	F	5	P	15	Z	25
6	6	G	6	Q	16		
7	7	H	7	R	17		
8	8	I	8	S	18		
9	9	J	9	T	19		

# I valori dei caratteri

Valori dei caratteri di posto dispari							
0	1	A	1	K	2	U	16
1	0	B	0	L	4	V	10
2	5	C	5	M	18	W	22
3	7	D	7	N	20	X	25
4	9	E	9	O	11	Y	24
5	13	F	13	P	3	Z	23
6	15	G	15	Q	6		
7	17	H	17	R	8		
8	19	I	19	S	12		
9	21	J	21	T	14		

Non è difficile scandire il codice fiscale da sinistra a destra, staccando via via una lettera; infatti basta usare i comandi ad argomenti delimitati e chiamare la macro suddetta con:

```
\expandafter\GetCFLetter\CFisc!
```

dove la stringa contenuta nel codice fiscale `\CFisc`, sviluppata prima dell'uso grazie a `\expandafter` è delimitata a destra dal punto esclamativo. Il primo token della stringa, cioè il primo carattere, viene staccato, e tutto il resto della stringa riassegnato a `\CFisc`.

In realtà ci vuole una macro che scarti i token che corrispondano ad uno spazio; questo si ottiene chiedendo alla macro di richiamante se stessa se il primo token è uno spazio:

```
\def\getCFletter#1#2!{%  
\ifx#1\space\getCFletter#2!\else  
\Letter='#1\def\CFisc{#2}\fi}
```

Lo sporco trucco consiste nel premettere all'argomento **#1** della macro il segno del backtick ' che serve per prelevare il codice interno di qualunque oggetto maneggiabile da T<sub>E</sub>X.

```
\Letter='#1
```

**\Letter** è il nome di un contatore (T<sub>E</sub>X); gli viene assegnato un numero; questo numero è il codice interno del carattere rappresentato da **#1**; siccome T<sub>E</sub>X al suo interno usa la codifica ASCII, quel codice interno non è altro che l'**indirizzo** di quel carattere nella tabella ASCII.

# Somma dei valori numerici

Per determinare la lettera di controllo bisogna sommare i valori numerici dei vari caratteri, valori diversi a seconda che il carattere occupi una posizione pari o dispari nella stringa del codice.

```
\@whilenum\cifra<16\do{%  
\advance\cifra\@ne  
\expandafter\getCFletter\CFisc!\relax  
\ifodd\cifra  
    \getOddValore%  
\else  
    \getEvenValore%  
\fi  
\advance\somma\valore}
```

# Assegnazione dei valori numerici ai caratteri

Le tabelle dei valori da assegnare ai vari caratteri secondo le norme ministeriali richiedono una procedura semplicissima per le cifre di posto pari:

```
\def\getEvenValore{%  
\ifnum\Letter<\A % È un numero?  
  \valore=\numexpr\Letter-\zero\relax % Sì  
\else  
  \valore=\numexpr\Letter-\A\relax % È una lettera  
\fi}
```

`\A` è il nome di un contatore  $\text{T}_{\text{E}}\text{X}$  contenente il codice ASCII di 'A'; `\zero` è il nome di un contatore  $\text{T}_{\text{E}}\text{X}$  contenente il codice ASCII della cifra '0'.



# Assegnazione dei valori numerici ai caratteri

Per i caratteri di posto dispari l'assegnazione del valore è decisamente più complessa perché i valori da assegnare ai vari caratteri non hanno nessuna relazione visibile con gli indirizzi ASCII:

```
\def\getOddValore{%  
\ifnum\Letter<\A  
\valore=  
  \expandafter\ifcase\numexpr\Letter-\zero\relax  
  1\or0\or5\or7\or9\or13\or15\or17\or19\or21\fi  
\else  
\valore=\expandafter\ifcase\numexpr\Letter-\A\relax  
  1\or0\or5\or7\or9\or13\or15\or17\or19\or21  
  \or2\or4\or18\or20\or11\or3\or6\or8\or12\or14  
  \or16\or10\or22\or25\or24\or23\fi  
\fi}
```

Il ciclo `\@whilenum` ci ha lasciato nel contatore `\valore` della lettera di controllo da verificare (essendo di posto pari non ci sono problemi con il suo valore e l'indirizzo ASCII); visto che la somma è stata eseguita su tutte le 16 cifre, bisogna sottrarre dalla `\somma` il `\valore` dell'ultima lettera che tra l'altro ci serve per la verifica.

```
\advance\somma-\valore
```

poi possiamo procedere alla verifica.

# \somma modulo 26

Ora eseguiamo il calcolo modulare e verifichiamo se la somma modulo 26 corrisponde esattamente al valore della cifra di controllo immessa:

```
\Letter\somma \divide\Letter by 26\relax
\somma=\numexpr\somma-26*\Letter\relax
\ifnum\valore=\somma
  \PackageInfo{codicefiscaleitaliano}{%
    \MessageBreak Codice fiscale OK}
\else
  \PackageError{codicefiscaleitaliano}{%
    \MessageBreak%
    *****\MessageBreak
    Codice fiscale #1 errato\MessageBreak
    *****}{...}
\fi
```

Dopo che è apparso su CTAN il mio pacchetto, Enrico, molto opportunamente ha prodotto un software analogo, apparso in un filone di discussione sul forum  $\text{\texttt{GUT}}$ , dove gli stessi calcoli e le stesse verifiche venivano eseguiti con le nuove macro “sperimentali” di  $\text{\texttt{LATEX}}$  3.

Lo scopo di Enrico era quello di mostrare la potenza di queste macro che non hanno bisogno di ricorrere a nessuno **sporco trucco** per raggiungere il risultato.

Che siano le mie macro, ormai già parte integrante di T<sub>E</sub>X Live 2012, o quelle di Enrico, reperibili nel forum GuIT, spero che questo esercizio di programmazione non sia stato eseguito solo per divertimento, ma sia utile a coloro che hanno bisogno di scrivere codici fiscali, avvocati, commercialisti, e altri professionisti utenti di L<sup>A</sup>T<sub>E</sub>X.