MC software RPN

a cura di Paolo Galassetti

Totocalcio

di Mauro Orlandini - Ferrara

Ecco qui un programmino per fare la schedina che non ha la pretesa dei grossi programmi per sistemisti ma che è certamente meglio della classica trottolina.

Il suo funzionamento è semplicissimo: alla domanda della calcolatrice bisogna immettere una valutazione sulle caratteristiche delle squadre (fattori CAMPO, TECNICO, FISICO, PSICOLOGICO) tramite un numero tra 1 e 4. Questa operazione va ripetuta per entrambe le squadre (OSPITANTE ed OSPITATA) e subito dopo otteniamo il "responso dell'oracolo HP": 1,X,2. Nella sua valutazione la calcolatrice utilizza dei "pesi", cioè dei coefficienti che elevano ad esponente i valori immessi dall'utilizzatore (righe 65, 72, 79, 86) ed inoltre inserisce un fattore di casualità (infatti introducendo gli stessi input per entrambe le squadre non è detto che si ottenga un X). Fatto il calcolo per le due squadre, se il totale è maggiore per la prima o la seconda squadra si ottiene rispettivamente un 1 o un 2 mentre per il pareggio (essendo quasi impossibile ottenere due numeri uguali a causa del fattore di casualità) si prende la "tolleranza" per la loro differenza uguale a 1 (riga 29).

Volendo si possono aggiungere un numero arbitrario di "fattori" con delle routine del tipo:

```
LBL 09          da 09 in poi, in sequenza
"FATTORE"
XEQ 02          routine di controllo dell'input
.5              "peso" dell'input
Yx
ST*IND 08
RTN
```

e modificando la linea 39 con un numero della forma 5.0NL con NL numero dell'ultima label usata.

A questo punto mi si dirà che un programma di questo tipo non serve perché è lo stesso che fare una schedina tradizionalmente. In realtà si ottengono molte sorprese quando si vanno a leggere le risposte date dalla HP (non pretendo di avere inventato il metodo per diventare milionario; a quest'ora vi scriverei dalle Isole Hawaii piuttosto che dal mio umido studio), comunque giocando sui "pesi" dei vari fattori si potrebbe ottenere un programma molto interessante.

* * *

Utilizzare un calcolatore elettronico per prevedere con certezza quale sarà la schedina vincente, è il sogno di molti programmatori. Chi più chi meno, tutti siamo stati accarezzati dall'idea di "buttar giù" un programma adatto a tale scopo. Alcuni scelgono soluzioni di tipo statistico, consistenti per esempio nell'escludere da tutte le colonne possibili, (3^{13} colonne), quelle con troppi "2", con troppi pareggi e così via; altri prendono la strada "tecnica" e studiano la classifica, le vicende interne della squadra, e mille altri fattori determinanti.

Il nostro programmino, senza troppe pretese, chiede all'operatore dei giudizi sulla partita e trae le conclusioni aggiungendo un "pizzico di suo" costituito da una variabile casuale introdotta nei calcoli effettuati. Purtroppo l'autore del programma ha esagerato nel peso da attribuire a quel "pizzico di suo" che la calcolatrice aggiunge, dato che questa, nel programma originale, moltiplica il risultato ottenuto dai conti precedenti per un numero casuale che va da 0 a 1; in pratica, il pronostico finale risulta, più che calcolato, "tirato a caso". Per far sì che il fattore casuale non assuma troppa rilevanza rispetto alle informazioni fornite dall'utente, occorre limitare il numero estratto ad un intervallo tanto più stretto intorno all'unità quanto meno deve influire l'elemento casuale. Per far ciò è possibile considerare quale elemento casuale, non il numero estratto (da 0 a 1), ma la sua radice ennesima che, pur essendo ancora un numero variabile da 0 a 1, presenta valori vicini allo zero tanto più raramente quanto più alto risulta n. Un valore ragionevole di n può essere 8, e per far ciò è stato sufficiente inserire dopo il passo 95 STO 00, tre istruzioni 96 SQRT, 97 SQRT e 98 SQRT. Un altro sistema consiste nel sommare, anziché moltiplicare, il numero casuale da 0 a 1 al valore ottenuto dai giudizi espressi dall'utilizzatore.

Ultima precisazione, il calcolatore non favorisce la squadra che gioca in casa, poiché si assume che tale fattore sia considerato nell'introdurre una variabile "Fattore Campo" normalmente più alta per la squadra di casa.

MC

Registri occupati

00	Seme casualità (meglio introdurlo all'inizio)
01	Totale squadra ospitante
02	Totale squadra ospite
03	Segno della schedina: 1, 2 o X
04 } 09 }	Nome squadra ospitante
05 } 10 }	Nome squadra ospite
06	Contatore loop
07	SIGN del totale squadra 1 meno totale squadra 2
08	Contiene 1 per la squadra 1 e 2 per quella 2
Size	011

```
01•LBL "TOT"
02 "OSPITANTE"
03 AON
04 PROMPT
05 AOFF
06 ASTO 04
07 ASHF
08 ASTO 09
09 1
10 STO 08
11 XEQ 03
12 "OSPITE"
13 AON
14 PROMPT
15 AOFF
16 ASTO 05
17 ASHF
18 ASTO 10
19 2
20 STO 08
21 XEQ 03
22 RCL 01
23 RCL 02
24 -
25 SIGN
26 STO 07
27 LASTX
28 ABS
29 1
30 X>Y?
31 GTO 00
32 RCL 07
33 X>0?
34 GTO 01
35 " 2"
36 ASTO 03
37 GTO 04
38•LBL 03
39 5.008
40 STO 06
41•LBL 10
42 XEQ IND 06
43 ISG 06
44 GTO 10
45 RTN
46•LBL 00
47 " X"
48 ASTO 03
49 GTO 04
50•LBL 01
51 " 1"
52 ASTO 03
53•LBL 04
54 CLA
55 ARCL 04
56 ARCL 09
57 "├/"
58 ARCL 05
59 ARCL 10
60 ARCL 03
61 PROMPT
62•LBL 05
63 "F. CAMPO"
64 XEQ 02
65 .6
66 Y↑X
67 STO IND 08
68 RTN
69•LBL 06
70 "F. TECNICO"
71 XEQ 02
72 .85
73 Y↑X
74 ST* IND 08
75 RTN
76•LBL 07
77 "F. FISICO"
78 XEQ 02
79 .8
80 Y↑X
81 ST* IND 08
82 RTN
83•LBL 08
84 "F. PSICOL."
85 XEQ 02
86 .65
87 Y↑X
88 ST* IND 08
89 RCL 00
90 9821
91 *
92 .211327
93 +
94 FRC
95 STO 00
96 SQRT
97 SQRT
98 SQRT
99 ST* IND 08
100 RTN
101•LBL 02
102 PROMPT
103 4
104 X<Y?
105 GTO 02
106 X<>Y
107 X<=0?
108 GTO 02
109 INT
110 END
```

Test di primalità

di Marco Panareo - Lecce

L'argomento forse più affascinante dell'aritmetica è costituito dai numeri primi; nel 1640 Fermat affermava: "Se potessi comprendere una volta per tutte la ragione sostanziale per cui 2,3,5,7,..... sono numeri primi, credo che scoprirei cose bellissime".

A tutt'oggi non esistono formule per costruire numeri primi, vi sono invece dei metodi più o meno raffinati tramite i quali si è in grado di stabilire se un numero è primo.

Questi procedimenti, grazie all'avvento dei calcolatori elettronici, ci consentono di asserire che numeri di migliaia di cifre sono primi, cosa pressocché impossibile impiegando unicamente carta e matita.

Il metodo impiegato nel programma per verificare la primalità di un dato numero, non prevede la fattorizzazione, ma sfrutta il cosiddetto "piccolo teorema di Fermat" che stabilisce: se "n" è primo e "b" è un intero, allora b^n-b è multiplo di "n". Ad esempio per $n = 13$ e $b = 2$ si ha che $2^{13}-2 = 8190$ è multiplo di 13: infatti 8190 è pari a 638×13.

Quindi se b^n-b non è multiplo di n, allora n non è primo, cioè se il rapporto $(b^n-b)/n$ dà resto diverso da zero possiamo concludere, in modo indiretto, che n non è un numero primo. Per $n = 6$ e $b = 2$ si ha, infatti, che $2^6-2 = 62$, che ha come resto 2 se diviso per 6.

Questo teorema, per quanto potente, presenta delle difficoltà nella sua applicazione: infatti se n è grande, per quanto piccolo sia b (al minimo 2), si rischia facilmente di mandare in overflow il calcolatore quando si calcola b^n.

Per nostra fortuna questo ostacolo è facilmente aggirabile utilizzando un metodo inventato il secolo scorso da Gauss: tale metodo prende il nome di aritmetica modulare.

Dati tre interi "a", "b", ed "m", positivi, negativi o nulli in quanto in genere "m" è un intero positivo maggiore di 1, se a-b è un multiplo di "m", diciamo che "a" è congruo a "b" modulo "m": in simboli $a \equiv b$ (*mod m*).

Ad esempio $7 \equiv 2$ (mod 5) oppure $10 \equiv 1$ (mod 3), dal momento che 2 e 1 sono rispettivamente i resti delle divisioni 7/5 e 10/3. Impiegando questa notazione possiamo allora scrivere che, se "n" è primo allora

$$b^n - b \equiv 0 \pmod n$$

Possiamo subito comprendere il vantaggio apportato dall'aritmetica modulare; è infatti possibile calcolare il resto della divisione $(b^n-b)/n$ senza eseguire direttamente questa operazione. Vogliamo per esempio calcolare 5^{523} (mod 523).

Ora, secondo un metodo ben noto a chi è pratico di computer, $523 = 2^9+2^3+2^1+2^0 = 512+8+2+1$ e perciò 5^{523} diventa $5^{512} \times 5^8 \times 5^2 \times 5^1$.

Quindi 5^{523} (mod 523) è congruo a 5^{512} (mod 523) $\times 5^8$ (mod 523) $\times 5^2$ (mod 523) $\times 5^1$ (mod 523).

Effettuando tutti i calcoli otteniamo che 5^{523} è congruo a 5 modulo 523, per cui $5^{523}-5 \equiv 0$ (mod 523), quindi 523 è primo. Non ne siete convinti?! Strano, a noi il discorso è sembrato.... congruente!

Dopo questa introduzione siamo in grado di discutere il programma. La prima parte calcola i termini

b^1 (mod n), b^2 (mod n), b^4 (mod n), b^8 (mod n), ..., b^q (mod n) dove "q" è il più piccolo numero della forma 2^s con $q > n$.

In tutto ciò si sfrutta abilmente il fatto che il resto del quadrato di un numero è (prendete fiato prima di continuare!) congruo al quadrato del resto del numero relativamente ad uno stesso modulo.

I termini di cui sopra sono memorizzati dal registro R07 in poi. Terminata questa fase, il programma calcola b^n come

$b^n = b^{(2^{S_1}+2^{S_2}+2^{S_3}+...+2^{S_m})}$ quindi esegue i prodotti

$b^{2^{S_1}}$ (mod n) $\times$ $b^{2^{S_2}}$ (mod n) $\times ... \times$ $b^{2^{S_m}}$ (mod n)

il cui risultato è b^n (mod n). Sottraendo b si ha b^n-b (mod n) che è la quantità che ci interessa.

Per far girare il programma basta inserire "n" e premere A, inserire "b" e premere B e calcolare il resto tramite C: se il resto è 0 allora il numero è primo.

Ma attenzione! Quanto sopra non è purtroppo sempre vero: ad esempio $2^{341}-2 \equiv 0$ (mod 341) pur essendo 341 composto, come pure $3^{91}-3 \equiv 0$ (mod 91).

I numeri per i quali accade ciò, relativamente ad un dato valore di "b", si dicono "pseudoprimi in base b".

Tuttavia questi sono piuttosto scarsi: si calcola infatti che la probabilità di commettere un errore nell'eseguire il test di Fermat in base 2 per tutti i numeri minori di 2^{10} è circa 10^{-6}. Esistono però metodi più raffinati, sempre basati sul test di Fermat, che riducono la percentuale di errore: questi procedimenti necessitano però di calcolatori molto più veloci della nostra TI (58 o 59).

Volete un esempio dei tempi di elaborazione ottenuti sulla TI 59? Per testare la primalità di 499 sono occorsi circa 40 secondi, mentre per il numero 2147483647 (che Eulero nel 1772 dimostrò essere primo) occorrono circa tre minuti.

MC

000	76	LBL	033	75	-	066	44	SUM	099	00	00	132	45	Y×
001	11	A	034	43	RCL	067	05	05	100	78	78	133	43	RCL
002	42	STO	035	01	01	068	02	2	101	43	RCL	134	05	05
003	01	01	036	65	×	069	45	Y×	102	05	05	135	85	+
004	91	R/S	037	53	(	070	43	RCL	103	85	+	136	43	RCL
005	76	LBL	038	43	RCL	071	05	05	104	07	7	137	03	03
006	12	B	039	03	03	072	95	=	105	95	=	138	95	=
007	42	STO	040	55	÷	073	42	STO	106	42	STO	139	42	STO
008	02	02	041	43	RCL	074	03	03	107	04	04	140	03	03
009	42	STO	042	01	01	075	67	EQ	108	73	RC*	141	52	EE
010	03	03	043	54	)	076	01	01	109	04	04	142	22	INV
011	91	R/S	044	59	INT	077	47	47	110	65	×	143	52	EE
012	76	LBL	045	95	=	078	69	OP	111	43	RCL	144	61	GTO
013	13	C	046	72	ST*	079	35	35	112	06	06	145	00	00
014	07	7	047	05	05	080	02	2	113	75	-	146	75	75
015	42	STO	048	33	X²	081	45	Y×	114	43	RCL	147	43	RCL
016	05	05	049	42	STO	082	43	RCL	115	01	01	148	06	06
017	25	CLR	050	03	03	083	05	05	116	65	×	149	75	-
018	42	STO	051	69	OP	084	85	+	117	53	(	150	43	RCL
019	04	04	052	24	24	085	43	RCL	118	73	RC*	151	02	02
020	43	RCL	053	69	OP	086	03	03	119	04	04	152	95	=
021	01	01	054	25	25	087	95	=	120	65	×	153	91	R/S
022	32	X:T	055	61	GTO	088	52	EE	121	43	RCL	154	00	0
023	02	2	056	00	00	089	22	INV	122	06	06	155	00	0
024	45	Y×	057	23	23	090	52	EE	123	55	÷	156	00	0
025	43	RCL	058	69	OP	091	67	EQ	124	43	RCL	157	00	0
026	04	04	059	35	35	092	01	01	125	01	01			
027	95	=	060	73	RC*	093	01	01	126	54	)			
028	77	GE	061	05	05	094	22	INV	127	59	INT			
029	00	00	062	42	STO	095	77	GE	128	95	=	001	11	A
030	58	58	063	06	06	096	01	01	129	42	STO	006	12	B
031	43	RCL	064	07	7	097	01	01	130	06	06	013	13	C
032	03	03	065	22	INV	098	61	GTO	131	02	2			