

# Quando i computer giocano a scacchi

*In questo speciale di DEV tratteremo un argomento molto affascinante ovvero la programmazione di un motore scacchistico e i principali concetti fondamentali da apprendere prima di cimentarsi nella realizzazione di questo tipo di software.*

di **Andrea Lanza** > [alanza@infomedia.it](mailto:alanza@infomedia.it)

**I**n Italia si disputa ogni anno il CIPS (Campionato Italiano per Programmi Scacchistici) che nel 2005 è arrivato alla sua quinta edizione.

Partecipano a questa manifestazione i principali motori italiani (che attualmente sono più di 30) sviluppati nei più disparati linguaggi di programmazione (C/C++, Delphi, Visual Basic e Java).

Generalmente, le partite si disputano con un'ora a testa di riflessione (60 minuti) più 10 secondi di bonus per mossa (modalità Fisher).

I contendenti sono uno di fronte all'altro, ciascuno con il proprio computer (la scelta della piattaforma hardware è in genere libera) e interposta tra i due vi è una scacchiera "reale" che riflette le mosse dei due giocatori.

I turni di gioco sono generalmente organizzati secondo il sistema svizzero cioè, un sistema che permette di disputare un torneo, quando si è alla presenza di parecchi giocatori e con poco tempo a disposizione (in genere un paio di giornate). Chi vince passa virtualmente il turno e si incontra con un altro giocatore vincente; chi perde invece continua a giocare e può rifarsi nella partita successiva. Al turno successivo inoltre, chi ha giocato con il bianco giocherà con il nero e viceversa.

Il numero dei turni deve essere scelto in maniera tale da permettere la cosiddetta forma a piramide dove al vertice dovrà esserci il giocatore che ha realizzato più punti degli altri.

La formula da utilizzare deve essere questa: se N è il numero di giocatori e T il numero di turni, allora N deve essere inferiore o uguale a 2 elevato a T.

Il numero di turni può così essere inferiore al numero di giocatori con grande risparmio di tempo.

Il sistema svizzero prevede di accoppiare tra loro i giocatori di pari punteggio, quando i partecipanti sono in numero pari, altrimenti, in caso di numero dispari uno di loro denominato "Floater" dovrà essere necessariamente abbinato ad un



**FIGURA 1**

Logo di Matilde

giocatore con punteggio inferiore.

Sempre nel 2005 si è svolta a Bologna la CCC (Computer Chess Cup), una manifestazione preolimpica che ha visto la partecipazione di numerosi motori di gioco italiani, in vista dell'evento del 2006, le olimpiadi scacchistiche di Torino, una manifestazione che comprende tra l'altro il famoso World Computer Chess Championship (WCCC) arrivata quest'anno alla 14° edizione e che vede la partecipazione dei "guru" di fama mondiale riguardo alla programmazione scacchistica.

Le partite del 14° WCCC sono suddivise su 11 turni secondo il sistema Svizzero e il tempo a disposizione per ciascun giocatore è 60 mosse in 120 minuti (X mosse in X minuti).

Anche in questo caso i due contendenti utilizzeranno una scacchiera per giocare la mossa elaborata dal proprio programma.

Ogni partita sarà seguita da un arbitro che verificherà la correttezza delle mosse giocate ed eventuali reclami (richiesta di patta per Threefold Repetition, illegal move ecc...).

Se un operatore inserisce per errore una mossa sbagliata è possibile, dietro consenso dell'arbitro del torneo, fare un "takeback" della mossa (se il proprio programma lo consente) e giocare la mossa corretta.

In questo caso gli orologi dei due contendenti verranno anch'essi aggiornati e sincronizzati.

Se due partecipanti arrivano a contendersi il primo posto ecco che scattano i “play-off”, due partite supplementari con un tempo a disposizione di 60 minuti a testa per ciascun giocatore.

Esistono numerosi motori scacchistici che possiamo reperire in rete, alcuni commerciali, altri open source ma con licenze restrittive per l'utilizzo dei sorgenti.

Un'ottima base per cominciare a studiare, sono i sorgenti del TSCP – Tom Kerrigan Simple Chess Program <http://home.comcast.net/~tckerrigan/> giunto alla versione 1.81, un programma scritto in C e molto bene commentato). I sorgenti resi pubblici comunque, non possono in alcun modo essere modificati previa esplicita autorizzazione dell'autore.

Un altro sito molto interessante e da consigliare per i principianti è senz'altro quello di Bruce Moreland (<http://www.brucemo.com/compchess/programming/index.htm>).

Vi sono inoltre delle competizioni che si svolgono su Internet; la WOCT - Winboard Olympic Chess Tournament (<http://vigo.altervista.org/>) arrivata quest'anno alla 4° edizione; WBEC Ridderkerk è un altro sito dove si organizzano tornei tra motori scacchistici e poi, vi sono siti alcuni gratuiti come FICS – Free Internet Chess Server (<http://www.freechess.org>) e altri a pagamento come Internet Chess Club (<http://www.chessclub.com>) dove, dal proprio PC di casa, si possono sfidare con il proprio motore avversari di qualunque parte del mondo.

La classifica italiana vede al primo posto ormai già da diversi anni Delfi T (<http://www.msbsoftware.it/delfi/>), un motore scritto in Delphi e con alcune parti di codice ottimizzate in assembler per renderlo più veloce.

Un bravo programmatore dunque, fa senz'altro la differenza e non necessariamente deve essere un bravo giocatore di scacchi.

Egli dovrà avere una buona conoscenza delle regole fondamentali del gioco e del valore da attribuire ai pezzi presenti sulla scacchiera (il cosiddetto materiale), ma soprattutto dovrà avere un'ottima conoscenza delle comuni tecniche di programmazione di un motore scacchistico.

Le differenze sostanziali tra i numerosi motori di gioco presenti oggi possono essere numerose;

si comincia dalla rappresentazione interna della scacchiera che può essere ad esempio una 8x8, oppure una 12x10.

Motori più forti possono utilizzare rappresentazioni 0x88 e bitboard e, ultimamente comincia a prendere piede anche la rappresentazione 16x16.

L'algoritmo di ricerca può essere una PVS (Principal Variation Search, anche se diversi motori sfruttano la possibilità di avere a disposizione più processori (ricerca parallela).

All'interno di questa ricerca possono essere introdotte dei controlli di verifica in base alla posizione del Re (è sotto scacco?), oppure se una mossa porta ad una cattura; in questo caso si procede ad estendere la nostra ricerca per cercare di sfruttare la situazione, anche se, bisogna essere cauti nell'utilizzo di queste “estensioni” perché si rischia di rallentare il nostro algoritmo di ricerca.

Un altro sistema per accelerare e non di poco la nostra funzione di ricerca è quella di utilizzare la cosiddetta

“Transposition Table”; siccome le mosse analizzate dal motore sono ripetute migliaia e migliaia di volte non ha senso tutte le volte doverle rielaborare; è molto più efficace memorizzare le mosse già viste in una grande “cache” in modo da poterle recuperare molto più velocemente.

La funzione di valutazione infine definisce il criterio per il quale sono scelte le mosse considerate migliori dal proprio motore.

Si comincia con la valutazione base del materiale (cioè la differenza tra i pezzi amici e quelli nemici) assegnando dei valori che vanno dal più basso (Pedone) al più alto (la Regina); il Re in genere non è tenuto in considerazione per il calcolo del materiale.

Seguono altri aspetti molto importanti riguardo la realizzazione della funzione di valutazione:

la mobilità, la struttura dei pedoni (pedoni passati, pedoni isolati, pedoni doppi ecc...), la sicurezza del Re... tutte considerazioni che, se ben applicate possono fare la differenza tra un motore ed un altro.

Un buon motore deve possedere un libro di aperture (un file che risiede su disco fisso in cui sono memorizzate tutte le posizioni iniziali di una partita di scacchi) in modo che perlomeno le prime mosse siano giocate senza perdita di tempo e con la certezza di giocare una buona mossa.

Inoltre molti motori dispongono anche delle EGTB (End Game Table Base), dei file che memorizzano il minor numero di mosse possibili per arrivare al matto a partire da 6, 5, 4 e 3 pezzi presenti sulla scacchiera.

Ad esempio se nel finale di partita sulla scacchiera rimangono solamente i due Re più una nostra Torre, il programma andrà ad attingere nel file KRK (Re, Torre e Re avversario) le mosse migliori da giocare per dare scacco matto con il minor numero possibile di mosse.

Nel server FTP troverete l'eseguibile, i sorgenti e i libri di aperture (Bianco e Nero) di Matilde, il motore sviluppato dal sottoscritto in Visual Basic 6 e Access 2003.

La versione è la 2.5.5, la stessa che ha partecipato al CCC di Bologna nel 2005 piazzandosi al 10° posto.

Se siete interessati alla nuova versione, la potete scaricare collegandovi al sito [www.cds.pc.it/Matilde](http://www.cds.pc.it/Matilde).

Con la versione 2.6.1 sono stati apportati notevoli miglioramenti; rappresentazione della scacchiera 12x10, algoritmo di ricerca Alpha-Beta con “Aspiration Window” e un miglior ordinamento delle mosse, estensione della ricerca in caso di scacco al Re (check extension), nuova funzione di valutazione per la struttura dei Pedoni (Pedoni passati, Pedoni doppi, Pedoni isolati ecc...) più un bonus per i Pedoni in settima traversa per facilitarne la promozione.

Ad oggi non mi risulta che sono disponibili motori scacchistici così intelligenti da riconoscere i propri errori ed evitare di ripeterli nelle partite successive, così come non esistono motori che reagiscono in maniera differente in base alla tipologia di avversario che sta loro di fronte.

L'attenzione di un motore di scacchi è rivolta solamente alla scacchiera (materiale, mobilità, posizione, sicurezza del Re, struttura dei Pedoni) e chissà, forse in un futuro ci saranno software che agiranno sull'aspetto psicologico ed emotivo, ma per adesso accontentiamoci che lo facciano i giocatori umani!