

COMPUTER CHESS TACTICS AND STRATEGY

by

ALEXANDER SZABO

B. Sc., The University of British Columbia, 1978

M. Sc., The University of British Columbia, 1980

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE

in

THE FACULTY OF GRADUATE STUDIES
DEPARTMENT OF COMPUTER SCIENCE

We accept this thesis as conforming
to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

July, 1984

© Alexander Szabo, 1984

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

Department of Computer Science

The University of British Columbia
1956 Main Mall
Vancouver, Canada
V6T 1Y3

Date August 8, 1984

Abstract

The design for a chess program is presented. The notions of tactics and strategy are used to define complementary ways of understanding any position. An association is made between tactics and brute force tree searching, and strategy and knowledge encoding. It is claimed that knowledge is best applied at the top of the tree as exemplified by TECH's positional presort rather than at the terminal nodes using complex evaluation functions.

The tactics part of the design is implemented by the program TECH3 which is described in terms of refinements to the minimax algorithm. The refinements are the α - β algorithm, the quiescence search, the transposition table, α - β move ordering, the search depth metric, the depth iterative α - β technique, and aspiration searching or windowing. The simple evaluation function used can also be considered as a refinement to minimax.

TECH3's performance on the problems in Reinfeld (1958) is 274/300 which, modulo machine power, compares favourably with BELLE's. A comparison is also made with the knowledge based tactics program, PARADISE.

Finally, the technology curve is developed as a tool for measuring the effectiveness of knowledge encoding (or strategy). In this respect NUCHESS is identified as the current best chess program.

Supervisor: Dr. R.S. Rosenberg

Table of Contents

	Page
Abstract	ii
List Of Tables	iv
List Of Figures	v
Acknowledgements	vi
 1. INTRODUCTION	 1
2. PRELIMINARY DISCUSSION	3
3. TACTICS AND STRATEGY	8
3.1 The Basic Idea	8
3.2 How To Solve Tactics	10
3.3 How To Solve Strategy	11
4. TECH3: A BRUTE FORCE TACTICS PROGRAM	14
4.1 Overview	14
4.2 The α - β Algorithm	15
4.3 The Quiescence Search	17
4.4 The Transposition Table	18
4.5 The Remaining Refinements	20
5. PERFORMANCE EVALUATION OF TECH3	25
6. THE TECHNOLOGY CURVE	31
7. CONCLUSIONS	38
 Bibliography And References	 40
Appendix 1: Source Code For TECH3	43
Appendix 2: Test Problems And Solutions	72
Appendix 3: Results Of TECH3 On Test Problems	84

List of Tables

Table	Page
I - Inaccuracies In The Book Solutions	29
II - The Problems Not Solved By TECH3 And BELLE	30

List of Figures

Figure	Page
1 - A Clear Example Of Tactics	13
2 - Human Strategy Versus Computer Tactics	13
3 - The Idea Behind The α - β Algorithm	24
4 - The Transposition Table	24
5 - The Elo Rating System	36
6 - The Technology Curve	37
7 - Fundamental Division Of The Problem	39

Acknowledgements

I thank my thesis supervisor, Dr. R.S. Rosenberg, for his suggestions and patient support over the three year period of this project. I am especially grateful for being allowed the freedom to follow my nose, for better or for worse.

I thank the Computing Center for supporting this project by making available a 0.2 rate factor account.

I thank my fellow graduate student, Barry Brachman, for the many discussions which helped shape the ideas presented here.

I thank Dr. Hans Berliner for providing a corrections list to the Reinfeld problems, and for his comments on the 1st edition of this thesis.

I thank Ken Thompson for providing the output of BELLE on the Reinfeld problems.

I thank Professor Marsland for his suggestions and comments on the 1st edition of this thesis.

Finally, I thank my parents for their encouragement and support.

1. INTRODUCTION

Computer chess is a fruitful testing ground for the ideas developed in the study of Artificial Intelligence (AI). For example, tree searching techniques have benefited from extensive use in chess-playing programs. Indeed, the claim that the state of the art in two-person tree searching is exemplified by the modern chess program is justified. Knowledge engineering, another notion of importance to AI, has also been applied to computer chess. Although significant, the results to date have been short of expectations. The goal of creating a master strength program, using tools other than brute force computational power, has eluded researchers.

The problem addressed by this thesis is simply stated: how to write a better chess program. How can a given amount of computational power be most effectively utilised for the purposes of chess-playing? This is the central question every chess-program writer grapples with.

In the following chapters, the plan for writing a better chess program based on thinking of chess in terms of tactics and strategy is presented. Chapter 4 describes TECH3, a program intended to solve chess tactics problems using the techniques of brute force tree searching. In Chapter 5, the performance of TECH3 is evaluated on the 300 problems in Reinfeld (1958) and compared with other brute force and knowledge based tactics programs, in particular BELLE and PARADISE. The technology curve, defined as the relation between chess strength and machine power for a TECH3-like program, is developed in Chapter 6 as a tool for measuring the effectiveness of knowledge

encoding. The final chapter summarises the major points along with providing directions for further work.

The Bibliography and References section can be used for further orientation. Readers unfamiliar with chess are directed to "Lasker's Manual of Chess" (Lasker 1947). You are warned that Lasker defines chess as an exercise in independent thinking. The ground work for computer chess was set by Shannon (1950). The history of the field since then can be obtained from Newborn (1975, 1979). A theoretical treatment of the α - β algorithm can be obtained from Pearl (1980).

2. PRELIMINARY DISCUSSION

The relevant history of computer chess begins with the contribution of Shannon in 1950. This is not to say that earlier developments were not important. In particular Zermelo (1912) showed that chess, and as a consequence other similar games, have a simple structure. His results are central to Game Theory, although much more work was required before this became apparent (Von Neumann and Morgenstern, 1944). Shannon's contribution lies in bridging theory and practice.

Game Theory classifies chess as a finite, two-person, zero-sum game of perfect information and no chance. The theory is that chess is solved by the minimax algorithm. The difficulty is that the computations take a long time. Shannon's solution was to terminate the algorithm prematurely by applying an evaluation function which guessed at the result of the truncated computation. Some discussion of terminology may be helpful at this point.

A concept called the game tree is a useful way of understanding the structure of games like chess. Each board position is represented by a node. Different board positions arise from the moves made by the players in the course of a game. Thus the board positions are logically connected by moves which are represented by directed arcs connecting the nodes. The essence of the game is contained in the game tree.

Consider some of the properties of the game tree. First notice that many nodes are not connected to the bulk of the tree, which arises from the node representing the starting board position. They form islands which are inaccessible to the

players. From this point on we will consider only the main island containing the starting node. Second notice that some nodes have no arcs leaving from them. These are called leaf nodes, and they represent mate positions. There are three types of leaf nodes corresponding to the three possible outcomes of chess, namely, white is checkmated and black wins, black is checkmated and white wins, or white or black are stalemated and the game is a draw. The three outcomes are referred to as the game theoretic values. Third notice that there are cycles in the game tree, whereby it is possible to repeatedly return to the same node. In fact some nodes form groups from which leaf nodes are inaccessible. Finally notice that in general there are many paths from one node to another, especially if the shortest path is long.

All nodes can be labeled with a unique game theoretic value. If from a node white can, no matter how black plays, arrive at a leaf node where black is checkmated, then the value of that node is, of course, that white wins. Conversely, if from a node black can, no matter how white plays, arrive at a leaf node where white is checkmated, then the game theoretic value of that node is that black wins. If neither white nor black can force a win then the node is labeled as a draw.

A systematic procedure for determining the game theoretic value of a node is the minimax algorithm which in negamax form can be described as follows:

```

Minimax( node )
{  if LEAF( node )
    then return( GAMETHEORETICVALUE( node ) )
  if CYCLE( node ) then return( 0 )
  a := -∞
  loop:
    {  node' := NEXTSUCCESSOR( node )
      if node'=nil then return( a )
      a' := -Minimax( node' )
      if a'>a then a := a'
      goto loop
    }
}

```

The input is a chess position (node). The top level input is referred to as the root position or the root node. The output, a , is the game theoretic value of the position. In the negamax formalism, if white is to move and white wins or black is to move and black wins then $a=+63$ (say), while if white is to move but black wins or black is to move but white wins then $a=-63$. If the game theoretic value of the position is a draw then $a=0$. The negamax formalism removes the importance of side-to-move, except at the top level. The LEAF predicate is true if its input is a leaf node. The CYCLE predicate is true if its input has occurred before on this variation. (A variation is a path from the root node towards a leaf node.) The GAMETHEORETICVALUE function returns 0 (=draw) if its input leaf node is a stalemate, and -63 if its input leaf node is a checkmate. The NEXTSUCCESSOR generator embodies the rules of chess. It successively generates successor nodes to its input node. The special value "nil" is returned after all successors have been generated.

In practice this is not effective since most positions require an unreasonable amount of computation. Shannon's

solution involved modifying the LEAF predicate so that many more nodes are "end points". These "end points" or pseudo-leaf nodes are called terminal nodes and the modified LEAF predicate I will refer to as the TERMINAL predicate. The GAMETHEORETICVALUE function must also be modified to return approximate values for non-leaf nodes. I will refer to the new function as the EVALUATION function.

Another important contribution made by Shannon was recognizing the problem of quiescence. Many positions cannot be evaluated (locally) by any function of less complexity than "minimax". Consequently it is simplest to choose TERMINAL such that the search continues through these nodes, if possible.

Shannon's ideas form the bases of every successful chess program to this date. Possibilities for the choice of TERMINAL and EVALUATION are far from being exhausted, nor do we have any good idea of what the limits of their potential might be.

Computer chess did not end with Shannon, far from it. Progress was however, slow. It took two decades, till the 1970's, before the state of the art had advanced to the point where tournaments between chess-playing programs became worthwhile. Enthusiasts rushed to code the most complex TERMINAL-EVALUATION combinations they could. What can hardly be considered surprising, the early experience was that all things being equal, simpler was better. Programs not based on the minimax paradigm did worst of all.

Three programs of note emerged during this period. The first was CHESS 4.5, a creation of Slate and Atkin (1977) at Northwestern University. This program (and its ancestors or

desendants) won virtually every computer chess championship in the 70's. Through their program, Slate and Atkin refined and developed the techniques of efficient tree searching (see Chapter 4). They made the minimax algorithm go.

The second program to emerge during this period was BELLE, a creation of Condon and Thompson (1982) at Bell Labs. Taking the best searching techniques available, they constructed special-purpose hardware of at least an order of magnitude more chess-specific computational power than any previous machine. Both CHESS 4.5 and BELLE are completely in the Shannon mold. Except for technique (α - β , transposition tables, etc) everything is according to Shannon.

In my view, the most important program to emerge (and submerge) during this period was TECH, a creation of Gillogly at Carnegie-Mellon University. TECH has significance because it hints at an extension to the Shannon paradigm. The extension (Gillogly's positional presort) is that a few computations done at or near the root node may have as much significance as the multitudinous computations done at the terminal nodes. The next chapter, after making clear the central thesis of this paper (computations for tactics vs computations for strategy), will hopefully enlighten the reader as to the significance of terminal node vs root node computations.

3. TACTICS AND STRATEGY

3.1 The Basic Idea

The basic idea is to think of chess in terms of tactics and strategy. This provides two complementary, yet complete, points of view. From the perspective of tactics, the construction of any program will focus on the local, direct consequences of the rules of chess, while the complementary perspective of strategy will necessitate a global focus. The two areas are approached with different devices. A clear separation of these different aims may provide the basis for the construction of superior programs. Other researchers including Moussouris, Holloway, and Greenblatt (1979) have expressed similar opinions.

Chess is often thought of in terms of tactics and strategy by human chess players. For them tactics is synonymous with combinations or calculating variations while strategy is more often referred to as position play or planning. The following three examples are provided to clarify this. (1) An example of tactics is illustrated by Figure 1 where black forces mate with the combination¹ 1... Bh2+ 2. Kh1 Bg3+ 3. Kg1 Rh1+ 4. Kxh1 Qh4+

¹ Chess moves are given by specifying the piece type which is to move and the square the piece is to move to. The piece types are P=pawn, K=king, N=knight, B=bishop, R=rook, and Q=queen. The squares are labelled by their column and row on the board. The 8 columns (files) are called, left to right (from the white side of the board), a, b, c, d, e, f, g, and h. The 8 rows (ranks) are called, bottom to top (from the white side), 1, 2, 3, 4, 5, 6, 7, and 8. Thus the bottom right corner square is h1. An "x" indicates a capture. A "+" indicates that the move gives check, while "++" indicates checkmate. In the figures black pieces are prefixed with "-".

5. Kg1 Qh2++. (2) An example of strategy can be gleaned from the endgame king and rook versus king. The side with king and rook plans to manoeuvre the lone king to an edge of the board. (3) The position in Figure 2 was used by Berliner (1973) to show the need for strategy in computer chess. A human chess player would realise with little trouble that the only thing to do here is Kd2-c2-b2-a3-b4-c5 etc (or Kd2-c1-b2 etc). This is strategically obvious. The same conclusion, with considerable effort, can be derived tactically. In fact it takes TECH3 less than two seconds to show that white forces the win of at least a pawn with Kd2-c1.

Tactics is a direct consequence of the rules of chess. Strategy, on the other hand, is what you do when nothing can be calculated. In practice this distinction is blurred. Material is a strategic concept which, because of its utility, has been transferred into the domain of tactics. Chess tactics problems are problems where the game theoretic values - win, lose, and draw - and the material balance play the decisive role. All other strategic considerations are ignored.

Strategy means forming a plan based on an assessment of the position. The plan does not arise from the creative fancy of a player, but rather the plan is based on the position. The problem of strategy is to formulate a mapping between positions and plans. This may not be difficult if the number of plans is not large. Strategy is an easier way of doing deep tactics. Strategy is what complicated tactics becomes. In principle either one is sufficient to solve chess, given enough time or space.

The nature of chess is such that to win, a background of near perfect shallow tactics is required. The prime reason for the lack of success of many chess programming efforts has been due to a deficiency on this point, or perhaps it is more accurate to say that the success of chess programs (like CHESS 4.5) has been due to the fact that they play tactically sound chess.

3.2 How to Solve Tactics

I propose that the most effective solution to tactics is provided by the brute force approach. This approach is illustrated and developed by the programs TECH (Gilloply 1972, 1978), CHESS 4.5 (Slate and Atkin 1977), and BELLE (Condon and Thompson 1982, 1983). The central core of the brute force approach is a direct, head-on attack to the problem. This consists of the well developed techniques of tree searching. These techniques are fully explained in Chapter 4 where TECH3 is described.

Another approach to solving tactics has been tried with some success. This approach is exemplified by CAPS II (Berliner 1974, 1977), Pitrat's program (Pitrat 1977, 1980), and PARADISE (Wilkins 1980, 1982). They have the common feature of utilising knowledge to guide and control the search. These efforts are of great value to the understanding of the problems of knowledge encoding. However, although in principle a knowledge based program could be constructed to perform with the accuracy of an exhaustive brute force program, such labor is misdirected. Knowledge is properly used by that which requires it (ie

strategy).

3.3 How to Solve Strategy

I propose that the most effective solution to strategy is provided by the knowledge approach. A major part of current efforts in AI are directed towards understanding (and defining) this approach. Chess programs will continue to play stronger as faster hardware is introduced. In this way they will inevitably surpass the best humans (see Chapter 6). To obtain a better chess program, independent of hardware, the use of human chess knowledge is essential. The problem of course is how.

One way of using knowledge in chess-playing programs is by applying evaluation functions to the terminal nodes of the search tree. An evaluation function accomplishes its task of encoding knowledge to solve strategy by assigning a single numerical measure of goodness to a position. In addition to material, an evaluation function will have terms dealing with pawn structure, king safety, piece activity and other concepts from human chess knowledge. Most programs, including BELLE, use evaluation functions. Slate and Atkin (1977) have described the evaluation function used by CHESS 4.5. The major drawback with this method of encoding knowledge is that the evaluation function must remain computationally simple if the brute force component is to remain effective.

The positional presort method of TECH provides a way of using knowledge and avoiding the terminal node computational bottleneck. This method can be viewed in two different ways. (1) An evaluation function is applied to the successors of the root

node. This defines the strategic value of the moves available at the root node. In the likely event that more than one move is available with largest tactical value, the strategic value acts as a tie breaker. (2) The positional presort defines a mapping between positions and plans. The appropriate plan guides the move selection among tactically equivalent moves. (A plan is a method or a rule or a prescription for what to do or how to handle a certain type of position.)

There are many other ways of using knowledge to solve strategy. This is an unexplored area for research.

8	.	.	.	-Q	.	-R	-K	.
7	-P	-P	-P	.
6	.	.	-P	-B	.	.	.	-P
5	.	.	.	-P
4	P	-R
3	.	P	N	Q
2	P	.	P	B	.	P	P	.
1	R	R	K	.
	a	b	c	d	e	f	g	h

Black to move

Figure 1 - A Clear Example of Tactics

8
7	.	.	.	-K
6	-P	P	-P	.
5	.	.	.	-P	P	.	P	.
4	.	.	-P	P
3	.	.	P
2	.	.	.	K
1
	a	b	c	d	e	f	g	h

White to move

Figure 2 - Human Strategy Versus Computer Tactics

4. TECH3: A BRUTE FORCE TACTICS PROGRAM

4.1 Overview

The algorithm described here has much in common with the program TECH (Gillooly 1972, 1978). For this reason the algorithm has been named TECH3. TECH3 is a brute force program. Its task is accomplished with almost no chess specific knowledge.

TECH3's I/O requirements are as follows. The input (I) consists of a legal chess position and a CPU time resource constraint. The output (O) consists of a legal chess move and a score. Naturally, the output move is legal (and best!?) in the input position. The score is just the 'minimax' value of the EVALUATION function applied to the terminal nodes of the tree defined by the search algorithm (TECH3) and the input (I).

The EVALUATION function used is simple. If the terminal node is game theoretic (ie a leaf node) then the game theoretic value is returned. Otherwise the material balance is returned. The game theoretic values are -63, 0, and 63 for side-to-move loses, draws, and wins, respectively. The material balance is given by the sum of the values of all the pieces on the board. The side-to-move piece values are 1, 18, 3, 3, 5, and 9 for pawn, king, knight, bishop, rook, and queen, respectively. The side-not-to-move piece values are the negatives of the side-to-move piece values.

TECH3 can be viewed as a number of refinements of the minimax algorithm. The refinements are the α - β algorithm, the

quiescence search, the transposition table, α - β move ordering, the search depth metric, the depth iterative α - β technique, and aspiration searching or windowing. The following sections will focus on each of these contributions in detail.

4.2 The α - β Algorithm

The central idea behind the α - β algorithm is that only one refutation per position is necessary. To understand this consider the following (see Figure 3). Let A be a position, Aa and Ab be successor positions to A , and AbA be a successor position to Ab . Suppose that the EVALUATION function applied to Aa yields $-x$. This implies (minimax) that the score of A is $\geq x$. (In the negamax formalism the sign changes with every level, thus $-x$ at Aa goes to x at A . The score of A is $\geq x$ because the player, always trying to maximize his score, has other choices (Ab) which may yield a larger score.) Suppose that the EVALUATION function applied to AbA yields y . This implies that the score of Ab is $\geq -y$ (by the same reasoning as before). The worst case condition for the player at Ab is that the score of Ab equals $-y$ in which case the score of A is $\geq y$. Finally, suppose that $x \geq y$. Clearly, examining other successors to Ab will not change the score of A . (Why? Because successors to Ab can only improve Ab 's score which at A makes y smaller still than x .) AbA has refuted Ab (relative to Aa). The subtree with root Ab can be cut off.

As a modification to the minimax algorithm the α - β algorithm can be described as follows:

```

aβ( a, β, node )
{
  if TERMINAL( node )
    then return( EVALUATION( node ) )
  loop:
  {
    node' := NEXTSUCCESSOR( node )
    if node'=nil then return( a )
    a' := -aβ( -β, -a, node' )
    if a'>a then a := a'
    if a≥β then return( a )
    goto loop
  }
}

```

The two parameters a and β are used to keep track of the best score for the current node and the score necessary to refute the current node, respectively. If a becomes $\geq \beta$ then the current node has been refuted and a is returned to the next higher level. This is referred to as a β -cutoff. If the score of the current node is not greater than the original a then a is returned anyway. This is referred to as an a -cutoff even though none of the tree is cut off at this level. An a -cutoff guarantees a β -cutoff at the next higher level. At an a -cutoff the score returned is $\leq a$. At a β -cutoff the score returned is $\geq \beta$. If neither cutoff occurs then a is less than the score returned which is less than β .

The predicate CYCLE has been removed since its purpose will be fulfilled by the transposition table. The functions Minimax(node) and $a\beta(-\infty, +\infty, \text{node})$ have identical I/O behavior (modulo CYCLE, TERMINAL, and EVALUATION).

4.3 The Quiescence Search

From a practical point of view the most important thing to evaluate accurately is the material balance. The quiescence search (Slate and Atkin 1977) is an efficient way of doing this. Essentially an α - β search is done on the tree, extending from terminal nodes, of all moves which immediately change the material balance. The quiescence search can be described as follows:

```

Qsearch(  $\alpha$ ,  $\beta$ , node )
{
   $\alpha'$  := EVALUATION( node )
  loop:
    {
      if  $\alpha' > \alpha$  then  $\alpha$  :=  $\alpha'$ 
      if  $\alpha \geq \beta$  then return(  $\alpha$  )
      node' := NEXTCAPTURE( node )
      if node' = nil then return(  $\alpha$  )
       $\alpha'$  := -Qsearch(  $-\beta$ ,  $-\alpha$ , node' )
      goto loop
    }
}

```

The NEXTCAPTURE generator only generates successors which have a different material balance than their parent node. Otherwise it is identical to the NEXTSUCCESSOR generator.

The order of the steps has been changed to reflect the possibility of early termination (via the null move). If at any point the side-to-move is satisfied ($\alpha \geq \beta$) with the immediate EVALUATION then it is assumed that a successor (the null move) exists which at worst preserves this EVALUATION. Otherwise the search continues until a node has no NEXTCAPTURE successors.

The quiescence search is integrated with the α - β algorithm by replacing EVALUATION(node) with Qsearch(α , β , node).

4.4 The Transposition Table

It is possible to arrive at the same chess position from many different move sequences. For example the move sequences 1. Ke2 Ke7 2. Ke3, 1. Kf2 Ke7 2. Ke3 and 1. Kd2 Ke7 2. Ke3 all lead to the same position. It is worthwhile supplementing the α - β algorithm with a mechanism which eliminates re-searching positions arrived at through a transposition of moves (Slate and Atkin 1977, Condon and Thompson 1982).

The basic problem here computationally is the detection of duplicate positions. This is solved elegantly by the following scheme. Assign 12 (64-bit) random numbers to each of the 64 squares on the chessboard. The 12 random numbers are associated, one to one, with the 12 types of pieces (white and black pawn, king, knight, bishop, rook and queen). The hash code of a position is the exclusive or (XOR) of the random numbers associated with the pieces on the board. What makes this computationally efficient is that once we have the hash code for a position, the hash code for successor positions can be easily determined by XORing just the pieces which change. For example, after the move Bc1-d2 the hash code for the new position is just the hash code for the old position XORed with the random number for Bc1 XORed with the random number for Bd2. To distinguish between positions which are different in side-to-move the hash code is further XORed each time the move phase changes (ie every move) with the move phase random number. Positions which are identical except for castling status or en passant status are not distinguished. The assumption is made that identical hash codes correspond to identical positions.

Part of the hash code (16 bits, maximum of 32 bits) is used as an array index into the transposition table while the rest (32 bits) is used as a hash code check. The table consists of (64-bit) entries divided into two parts. The first part (32 bits) of each table entry is the hash code check while the second part (32 bits) stores information relevant to the α - β algorithm. The information stored includes (1) (5 bits) the depth to which this position has been searched, (2) (6 bits) the best move to be played from this position, (3) (2 bits) a flag indicating the status of this node (α -cutoff, β -cutoff or no cutoff), and (4) (16 bits) the score for this position (see Figure 4).

Each time a node is entered in the search process a check is made to see if it is in the transposition table. If it is, and the search depth required is less than or equal to the search depth stored, then one of the following three conditions holds. (1) The node is an α -cutoff in which case the current β parameter is set to the minimum of the stored score and the current β parameter. (2) The node is a β -cutoff in which case the current α parameter is set to the maximum of the stored score and the current α parameter. In either case if $\alpha \geq \beta$ then further processing is not necessary and the score returned is the stored score. Otherwise processing continues normally. (3) The node is not a cutoff in which case the score returned is the stored score.

Each time a node is exited by the search process a check is made to see if it should be written in the table. This is done if the depth this node has been searched to is greater than or

equal to the depth stored in the table at the write location. Thus the table has a priority structure favouring positions close to the root over positions deep in the tree. Normally the table is 'zeroed' before each top level α - β invocation.

Notice that the table can be used to detect draws by repetition. This is done trivially by writing draw scores ($=0$) to the table for each node on the way down the tree. (This is only done if the search depth is greater than or equal to the depth stored in the table.) The node, if encountered lower in the tree, will necessarily cause the search to backup. The draw score is unfailingly overwritten with the correct score on backing up through the original node.

The table also maintains the principal variation by keeping track of best moves. From the point of view of Game Theory the transposition table encodes the best strategy for both sides.

The probability of obtaining false data from the table on a single probe is about one chance in 2^N , where N is the number of bits in the hash code check. This error rate, from practical experience, is acceptable even for small N ($N=14$).

4.5 The Remaining Refinements

(1) α - β move ordering:

The efficiency of the α - β algorithm is dependent on the order in which successor positions (or moves) are searched. In the worst case α - β examines exactly the same nodes as minimax and thus is of no benefit at all. The best case, however, allows α - β to examine only on the order of the square root of the

number of nodes examined by minimax and yet still yield the same result.

In practice it has been discovered that a simple ordering works well here. First, the best move from the transposition table entry for this node is searched. Second, the moves which change the material balance are searched in the order of largest changes first. Finally, the remaining moves are searched. The NEXTSUCCESSOR and NEXTCAPTURE generators have this responsibility.

(2) the search depth metric:

It should be clear that the TERMINAL predicate does an enormous job in defining the character of the search algorithm. TERMINAL defines the search depth metric. A common metric, known as full width d-ply searching, is to declare leaf nodes and all nodes $\geq d$ levels from the root node terminal. A variant of this which takes into account the tactical importance of moves which give check, is to not count moves which give check when counting the d levels from the root node (ie moves which give check count for zero while all other moves count for one). The version of TECH3 described in the 1st edition of this thesis employed this metric. BELLE uses a similar search depth metric, not counting replies to check instead of the checking moves themselves.

The metric used by the version of TECH3 described here is a variant of full width d-ply searching where checking moves count for zero, replies to check (if not checking moves themselves) count for one, and all other moves not of the above two types have a weight of one or two depending on a comparison with the

current α parameter. If the material balance after the move is played is greater than α then the move counts for one, otherwise ($\text{material balance} \leq \alpha$) the move counts for two. The justification for this new metric is that it works (see Chapter 5).

(3) the depth iterative α - β technique:

A critical problem for chess programs is that of time control. Chess is played in a finite time environment. The search algorithm must return with its results after a predictable time interval. Because of the highly variable nature of chess search trees, there is some difficulty in accomplishing this. The depth iterative α - β technique was introduced to deal with this problem (Slate and Atkin 1977). This technique can be described as follows:

```

Iterative $\alpha\beta$ ( node )
{
  d := 1
  loop:
  {
    d := d + 1
    score :=  $\alpha\beta$ (  $-\infty$ ,  $+\infty$ , node )
/* d is passed to TERMINAL */
    if NOMORETIME then return( score )
    goto loop
  }
}

```

NOMORETIME is a predicate which is true if a predefined time threshold is exceeded. The transposition table (with best move stored) generally allows the iterative version to complete a d-ply search in less time than the non-iterative version.

(4) aspiration searching or windowing:

Aspiration searching or windowing refers to the choice of values for α and β in the top level call of the α - β algorithm. The smaller the difference between α and β , the smaller the window. Generally, smaller windows lead to faster searches. There are a number of ways this choice can be made (Marsland and Campbell 1982). The following is reasonable:

```

Search( node )
{
  d := 1
  score := EVALUATION( node )
  loop:
    {
      d := d + 1
      score' :=  $\alpha\beta$ ( score-1, score+1, node )
/* d is passed to TERMINAL */
      if score'  $\neq$  score then
        score :=  $\alpha\beta$ (  $-\infty$ ,  $+\infty$ , node )
      if NOMORETIME then return( score )
      goto loop
    }
}

```

The window is made as small as possible around the score returned by the previous search iteration on the assumption that this iteration will return the same score. If it does not then the search is repeated with a wider window (ie $\alpha=-\infty$, $\beta=+\infty$). The transposition table assists in the repeated search.

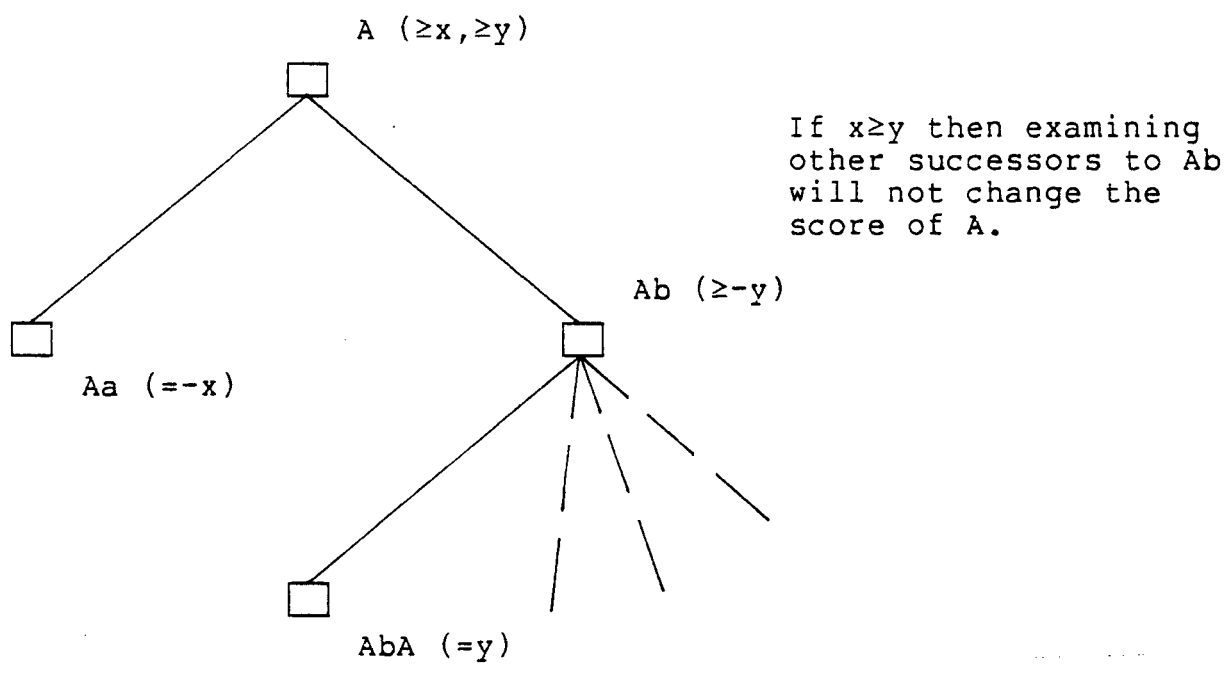


Figure 3 - The Idea behind the α - β Algorithm

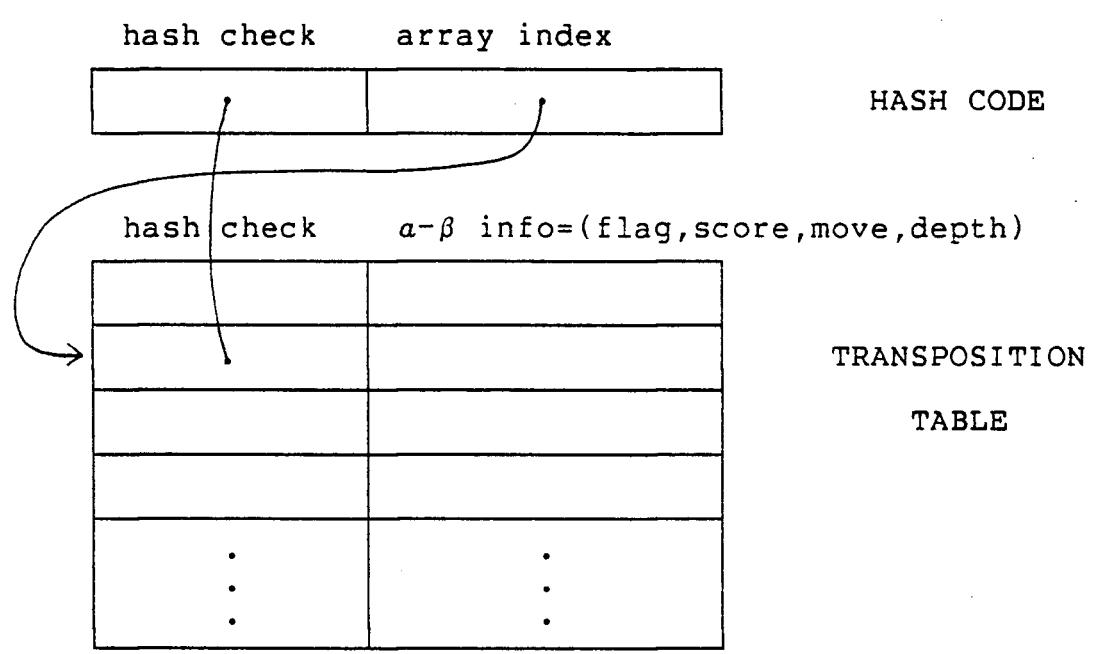


Figure 4 - The Transposition Table

5. PERFORMANCE EVALUATION OF TECH3

TECH3, running on an Amdahl 470 V/8, was tested on the positions in the book "Win at Chess" (Reinfeld 1958). This book contains 300 chess tactics problems and their solutions. Other programs, including PARADISE and BELLE, have been tested on these problems and thus it is possible to compare their performance with TECH3's. The complete Fortran and 370 Assembler source code for the version of TECH3 which was tested is given in Appendix 1.

TECH3 was given credit for solving a problem if (1) the CPU time used did not exceed 5 minutes and (2) the move returned was the same as one of the book solutions. In some cases more than one solution (move) was considered valid. No consideration was given for whether TECH3 understood¹ the reason behind the initial move. Credit was given for valid moves tendered for spurious reasons. This decision was made to make the credit scoring as objective (and automatable) as possible.

The book solutions were found with the aid of TECH3 and in communications with Berliner (1984) to have 12 inaccuracies. These are listed in Table I. The 300 positions from "Win at Chess" in pseudo-forsythe notation as they were input to TECH3 along with their accepted solutions are listed in Appendix 2.

Given these corrections TECH3 obtained a score of 274/300.

¹ If a program "sees" in its search tree all the relevant factors behind a move then it understands the problem. It is not always easy to see if the program sees. Would you give credit to TECH3 for #9 at D=3 (see Appendix 3) for seeing the draw by repetition?

The 26 problems not solved are listed in Table II. The unsolved problems were due to exceeding the 5 minute CPU time limit (or at least getting very close to it). More details can be obtained from Appendix 3 where the actual output of TECH3, in summarized form, is listed.

PARADISE was tested on 92 of these problems (Wilkins 1980, 1982). These were #1,...,#100 not including #2, #18, #33, #41, #43, #86, #87, and #100. On these 92 problems PARADISE got credit for 89. PARADISE failed to solve #31, #71, and #91. On these same 92 problems TECH3 got credit for 90, failing to solve #71 and #92. I would be interested to know how PARADISE performed on #92 since the solution given by Reinfeld is incorrect and Wilkins makes no mention of this in his papers.

Berliner (1981) gave BELLE credit for 280.5 on the full set of problems. I have examined the output of a more recent version of BELLE (Thompson 1984). Using the same scoring criterion outlined above for TECH3, BELLE obtained a score of 282/300. The 18 problems not solved by BELLE are listed in Table II where they can be compared with the 26 problems not solved by TECH3. It is clear that BELLE is superior, in real time, to TECH3 running on an Amdahl V/8.

A more appropriate measure of performance for a TECH3-like (or BELLE-like) program is to consider the basic computational step inherent in the design. This basic unit of computation I will call the "arc". An arc is the computation necessary to go from one node to its successor node and later, after possibly other arcs, return to the original node. The search tree contains only legal chess positions. Consequently arcs which

visit pseudo-legal positions (ie king left in check) are never counted. All extraneous computation is considered arc overhead.

To obtain some idea of the complexity of an arc consider the following. On an Amdahl 470 V/8, TECH3 proceeds at an average of 21,000 arcs per second. The V/8 hums along at 40 cycles per microsecond. The machine instructions forming TECH3 require, on average, 2.5 machine cycles per instruction to execute. Consequently each arc consists of the execution of about 760 machine instructions.

BELLE's measure of performance in this respect is 120,000 arcs per second (Thompson 1984). Are the algorithms equivalent in the sense that they accomplish identical results given the same number of arcs? How can two programs be compared algorithmically, or independent of the different hardware they may be using?

There are two things which must be defined (and measured). First is the difference in hardware, the difference in chess-specific computing power. Second is the real-time performance, the time taken to solve a chess problem. To define the chess-specific computing power assume a very simple brute force tree searching algorithm (like TECH3) is implemented on both machines. The difference in hardware is then the ratio of arcs per second performance of the same algorithm on the two machines. I estimate the difference between the chess-specific computing power of the BELLE hardware and an Amdahl V/8 as a factor of 8. Notice that this is not the ratio BELLE/TECH3 arcs per second performance since BELLE and TECH3 are not identical algorithms. Removing the slow evaluation component from BELLE,

for example, should increase BELLE's arcs per second performance and at the same time move the algorithms "closer".

The real-time performance of the two algorithms can be determined by comparing their performance on the "Win at Chess" problems. Since there is a high variance of performance on any single problem, the comparison is done by considering the amount of time required to solve N problems, where N can vary anywhere from 1 to 300. Put another way, we know that TECH3 solves 274 problems in less than 5 minutes per problem. Compare this number (5 minutes) with the amount of time BELLE requires to solve 274 problems. Repeat this comparison for 273 problems solved, then 272 problems solved, and so on. The average of the ratio of the time for TECH3 to solve N problems to the time for BELLE to solve N problems ($N=230$ to 268) is more than 2 but less than 3 ($2.5 \pm .5$).

Algorithmically, TECH3 is better than BELLE by a factor of $8/2.5$ or about 3 (± 1 all things considered).

Table I - Inaccuracies in the Book Solutions

# 33	Qe4-f4	(1)
# 92	Be6xg4	(2)
#123	Re7-c7	(1)
#130	Qg7-h8	(1)
#152	Nc3-e4	(1)
#157	Nd5-e7	(2)
#210	Rd1-h1	(2)
#224	e5-e4	(1)
#264	Ra8-b8	(1)
#277	Rg8xg2	(1)
#294	Bd6-f8	(1)
#296	Bc6-d7, Bc6xf3	(1)

Explanations:

(1) This move(s) is equivalent to the book solution. Count the move(s) as a valid solution.

(2) This move is better than the book solution. Count the move as a valid solution. The book solution is not valid.

Table II - The Problems not Solved by TECH3 and BELLE

# 2	# 2	#180	#180		#248
# 71	# 71		#196		#249
	# 86	#213			#262
# 87	# 87	#222	#222	#265	#265
# 92	# 92	#229	#229	#269	#269
#100	#100	#230	#230		#270
#141	#141	#235	#235	#275	#275
#150			#239		#293
	#155	#241		#297	#297
	#163		#243		

Explanation:

The first of each pair of columns lists the 18 problems not solved by BELLE, while the second lists the 26 problems not solved by TECH3. The problems are listed such that it is clear that 3 problems (#150, #213, and #241) are solved by TECH3 but not by BELLE, 11 problems are solved by BELLE but not by TECH3, and 15 problems are not solved by both programs.

6. THE TECHNOLOGY CURVE

The technology curve is a graph of chess strength versus machine power (see Figure 6). It delineates to what extent success is due to hardware.

The Y-axis represents the chess strength of a program in Elo rating points. The Elo rating system is described by Elo (1978). Figure 5 shows the correspondence between ratings and descriptive titles in this system. It is sufficient to know that as a linear approximation the rating difference between two players in a match is given by $400 \cdot (W-L)/N$ where W is the number of games won, L the number lost, and N the total number played.

The X-axis represents the machine power in terms of the speed at which an efficient implementation of a very simple brute force tree searching algorithm (like TECH3) is executed. The execution speed is measured by counting the number of arcs traversed in the search tree per unit time (one way only). Arcs are counted at all levels of the tree including the quiescence part. Arcs traversed more than once, because of iteration or windowing for example, are counted once for each traverse. This is identical to the arcs per second performance measure introduced in the last chapter. The machine power is expressed as the logarithm to the base 2 of the arcs per second performance measure. The machine power of the Amdahl 470 V/8 is

assumed to be $\log_2(16,000 \text{ arcs/second})$ which is about 14^1 .

There are a number of ways the X-axis can be interpreted. Since chess games are normally played under tournament conditions of 40 moves in 2 hours, the average number of nodes visited per move can be determined. Changing the rate at which play proceeds is equivalent to changing the machine power. Tournament chess proceeds at an average rate of 2^7 seconds per move while speed chess is played at an average rate of 2^2 seconds per move. A machine power difference of 5 can be simulated by changing from tournament to speed chess.

Another useful measure of machine power is the average maximum full width depth searched at tournament speeds. A factor of about $32=2^5$ additional effort is required to extend the search by 2-ply. To do a 7-ply search requires a machine power of about 15. A 9-ply search in the same time would require a machine power of 20.

The BELLE data in Figure 6 was computed from data presented in Thompson (1982) and Condon and Thompson (1983). A series of BELLE-like programs P3, P4, ... P9 restricted in their search depth were defined. Matches were played between the programs to determine their relative ratings. An absolute rating for the programs was obtained by assuming $BELLE = (17,2200)$ (ie the machine power of the BELLE hardware is 17 and the chess strength

¹ This is based on an earlier version of TECH3 described in the 1st edition of this thesis. Since the V/8 has not changed between editions, it seems reasonable that its machine power should remain at 14. My position is now that the V/8 (=14) is the standard and any piece of code which can be considered as typifying chess-specific computations can be used for comparison to the V/8.

of BELLE is 2200).

A similar experiment had been performed earlier by Gillogly (1979) using TECH. For comparison the TECH data is also plotted on Figure 6. The rating computations have been done assuming $TECH = (9, 1250)$ (machine power for a PDP-10 + KA10 (0.34 mips) is assumed to be 9).

Newborn (1979) has published his own version of the "technology curve". For comparison his results are shown in Figure 6 as well.

The solid line in Figure 6 is the technology curve. This line represents the playing strength of the tactics algorithm, TECH3, presented in Chapter 4. No independent data is given to support the claim that the solid line in fact represents this. The slope of the solid line is based on an average of the slopes of the 3 dashed lines. The offset of the solid line is based on the observation by Gillogly (1978) that the positional presort improves TECH by about 1-ply (=250 rating points) and my estimate that TECH3 is two times better than TECH at spending arcs². (This estimate is supported by comparing TECH's performance with TECH3's on problems #19, #29, #30, and #48.) Thus the solid line is fixed at $(9, 1250 - 250 + 100 = 1100)$. Until additional supporting data is provided the technology curve is only qualitatively defined.

Using the technology curve it is possible to measure the

² Again this refers to the 1st edition version of TECH3. The version in this thesis is estimated to be two times better again. The effect of this is that the technology curve should be raised by an additional 100 rating points.

success with which chess knowledge has been encoded in programs. This is done by comparing the established rating of programs like BELLE, CRAY BLITZ, NUCHESS, CHAOS, etc. with the rating given by the technology curve at the machine power for the machine used. For example BELLE has an established rating of 2200. The machine power of the BELLE hardware is estimated at 17. The technology curve at 17 yields a rating of 1900. Thus BELLE encodes about $2200 - 1900 = 300$ rating points worth of chess knowledge. As further examples, consider CRAY BLITZ, NUCHESS and CHAOS with established ratings of 2200, 2100 and 1900 respectively. CRAY BLITZ runs on a Cray-I XMP, NUCHESS established its rating on a CDC Cyber 175, while CHAOS uses an Amdahl 470 V/8. The machine power for these computers is estimated at 16, 14, and, of course, 14, respectively. The technology curve rating at these machine powers is 1800, 1600, and 1600, respectively. Thus CRAY BLITZ encodes $2200 - 1800 = 400$ rating points worth of chess knowledge, while NUCHESS encodes $2100 - 1600 = 500$ rating points worth, and CHAOS encodes $1900 - 1600 = 300$ rating points worth. The three sources of error in this measurement are the determination of the established rating, the determination of the machine power, and the determination of the technology rating.

The main points we can derive from the technology curve are as follows. (1) For every factor of two increase in machine power, brute force chess programs improve by 100 rating points. (2) The amount of chess knowledge encoded in competition programs ranges from 500 points for NUCHESS to 300 points for BELLE and CHAOS. All chess-playing programs, including the

current world champion CRAY BLITZ, and commercial chess computers³, probably fall within this narrow range. (3) Hardware improvements over the history of computer chess have increased computing power by a factor of 1000 ($=2^{10}$). This corresponds to a rating improvement of 1000 points. A further increase in computing power by a factor of 32 (500 rating points) is required before computers will be able to compete on par with humans for the world championship.

³ There is evidence to suggest that some commercial chess computers are pushing the top of this range. For example the Fidelity Elite based on the 6502 microprocessor (at 4MHz) has a rating of perhaps 1800. The machine power of the 6502 (at 4MHz) is optimistically 10. (It's hard to believe that an 8-bit microprocessor is 1/16 an Amdahl V/8.) Thus the Elite encodes $1800-1200=600$ points of knowledge.

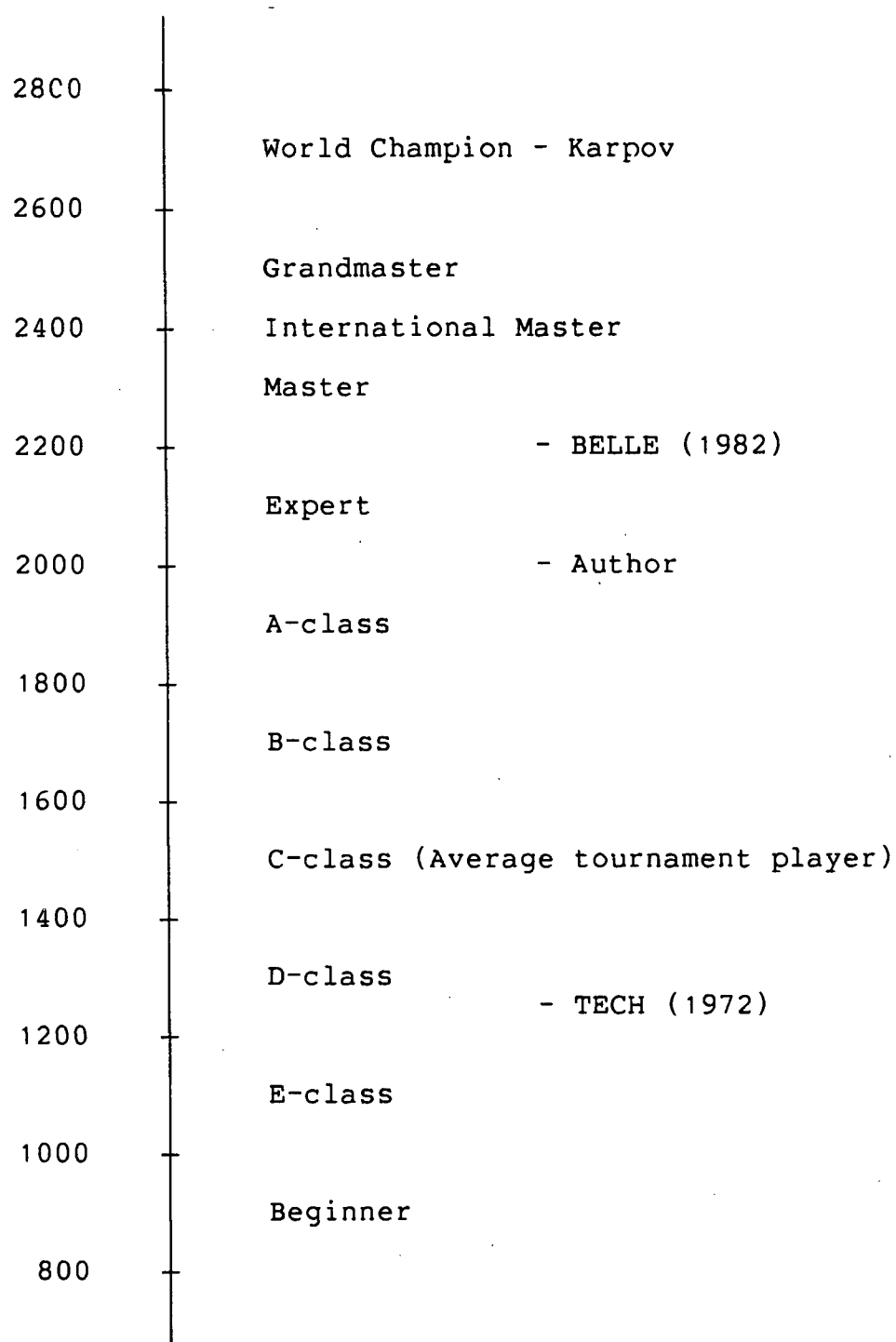


Figure 5 - The Elo rating system

THE TECHNOLOGY CURVE:

Chess Strength VS Machine Power

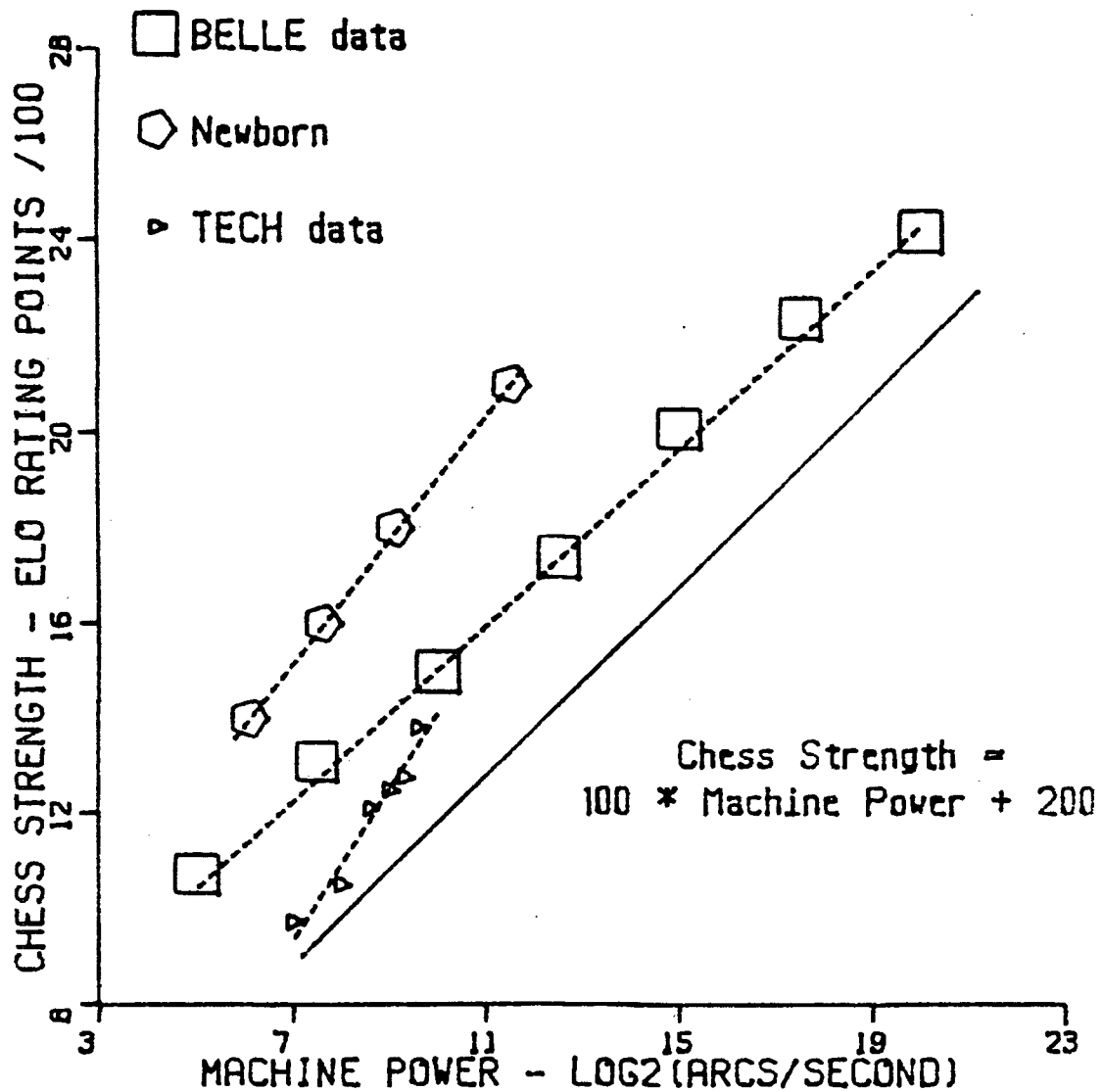


Figure 6 - The Technology Curve

7. CONCLUSIONS

A framework for understanding computer chess has been presented. The basic points are: (1) It is useful to illuminate computer chess from the point of view of tactics and strategy. Tactics is essential to playing chess well. (2) Brute force programs are an efficient method of dealing with chess tactics. They are available off the shelf. (3) The technology curve provides the means to measure the success with which knowledge has been encoded. Thus the effects of hardware and tree searching techniques are decoupled from computer strategy. (4) Efforts should now be directed towards discovering an effective computer chess strategy. The positional presort technique merits further investigation.

At the top level our problem is to construct a program which plays chess well. A fundamental step towards a solution is the division of the problem into two parts - strategy and tactics (see Figure 7). The strategy part is responsible for the global features of the game. This involves planning. The tactics part is responsible for giving good advice on local features of the game to the strategic part. This involves calculating variations. An attempt to proceed along these lines has been made.

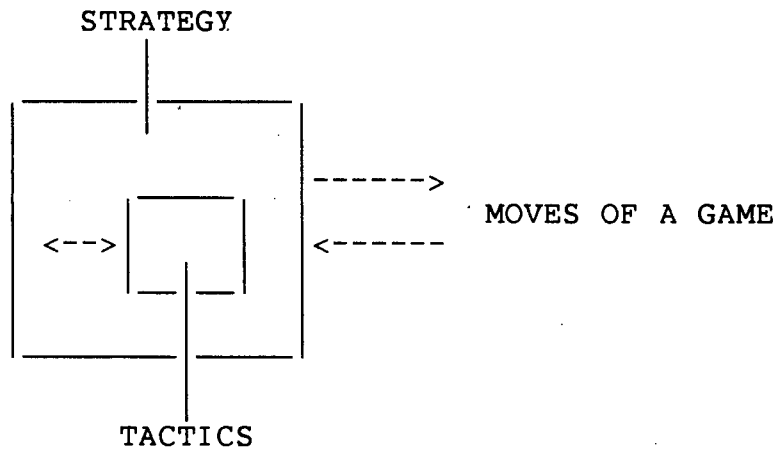


Figure 7 - Fundamental Division of the Problem

Bibliography and References

Adelson-Velskii, G.M., Arlazarov, V.L., Bitman, A.R., Zhivotovskii, A.A., and Uskov, A.V. (1969), "Programming a Computer to Play Chess", Mathematics Surveys 25, pp221-262.

Berliner, H.J. (1973), "Some Necessary Conditions for a Master Chess Program", Proceedings 3rd International Joint Conference on Artificial Intelligence, pp77-85.

Berliner, H.J. (1974), "Chess as Problem Solving: the Development of a Tactics Analyzer", Ph.D. Thesis, Carnegie-Mellon University.

Berliner, H.J. (1977), "A Representation and Some Mechanisms for a Problem-Solving Chess Program", Advances in Computer Chess 1, editor M.R.B Clarke, pp7-29, Edinburgh.

Berliner, H.J. (1981), "An Examination of Brute Force Intelligence", Proceedings 7th International Joint Conference on Artificial Intelligence, pp581-587.

Berliner, H.J. (1984), Private Communications.

Condon, J.H. and Thompson, K. (1982), "Belle Chess Hardware", Advances in Computer Chess 3, editor M.R.B. Clarke, pp45-54, Pergamon.

Condon, J.H. and Thompson, K. (1983), "Belle", Chess Skill in Man and Machine, 2nd edition, editor P.W. Frey, pp201-210, Springer-Verlag.

Elo, A.E. (1978), The Rating of Chessplayers, Batsford.

Frey, P. (1980), "Machine Problem Solving Part 3: The Alpha-Beta Procedure", BYTE 5 (November), pp244-264.

Gillogly, J.J. (1972), "The Technology Chess Program", Artificial Intelligence 3, pp145-163.

Gillogly, J.J. (1978), "Performance Analysis of the Technology Chess Program", Ph.D. Thesis, Carnegie-Mellon University.

Lasker, Emanuel (1947), Lasker's Manual of Chess, Dover.

Marsland, T.A. and Campbell, M. (1982), "Parallel Search of Strongly Ordered Game Trees", Computing Surveys 14, pp533-551.

Michie, D. (1982a), "Game-Playing Programs and the Conceptual Interface", Sigart Newsletter 80, pp64-70.

Michie, D. (1982b), "Computer Chess and the Humanization of Technology", Nature 299, pp391-394.

Moussouris, J., Holloway, J., and Greenblatt, R. (1979), "CHEOPS: A Chess-Oriented Processing System", Machine Intelligence 9, editors J.E. Hayes, D. Michie, and L.I. Mikulich, pp351-360, Wiley.

Newborn, M.M. (1975), Computer Chess, Academic Press.

Newborn, M.M. (1979), "Recent Progress in Computer Chess", Advances in Computers 18, editor M. Yovits, pp59-117, Academic Press.

Newell, A., Shaw, J.C. and Simon, H.A. (1963), "Chess-Playing Programs and the Problem of Complexity", Computers and Thought, editors E.A. Feigenbaum and J. Feldman, pp39-70, McGraw-Hill.

Pearl, J. (1980), "Asymptotic Properties of Minimax Trees and Game-Searching Procedures", Artificial Intelligence 14, pp113-138.

Pitrat, J. (1977), "A Chess Combination Program Which Uses Plans", Artificial Intelligence 8, pp275-321.

Pitrat, J. (1980), "The Behavior of a Chess Combinations Program Which Uses Plans", Advances in Computer Chess 2, editor M.R.B. Clarke, pp110-121, Edinburgh.

Reinfeld, F. (1958), Win at Chess, Dover.

Shannon, C.E. (1950), "Programming a Computer for Playing Chess", Philosophical Magazine 41, pp256-275.

Slate, D.J. and Atkin, L.R. (1977), "CHESS 4.5 - The Northwestern University Chess Program", Chess Skill in Man and Machine, editor P.W. Frey, pp82-118, Springer-Verlag.

Thompson, K. (1982), "Computer Chess Strength", Advances in Computer Chess 3, editor M.R.B. Clarke, pp55-56, Pergamon.

Thompson, K. (1984), Private Communication.

Von Neumann, J., and Morgenstern, O. (1944), Theory of Games and Economic Behavior, Princeton.

Wilkins, D.E. (1980), "Using Patterns and Plans in Chess", Artificial Intelligence 14, pp165-203.

Wilkins, D.E. (1982), "Using Knowledge to Control Tree Searching", Artificial Intelligence 18, pp1-51.

Zermelo, E. (1912), "Uber Eine Anwendung der Mengenlehre auf die Theorie des Schachspiels" (in German), 5th International Congress of Mathematics, pp501-504.

Appendix 1: Source Code for TECH3

Listed are 7 pages of Fortran code followed by 20 pages of 370 Assembler code. The Fortran code includes the routines MAIN, BLOCK DATA, TACTIC, and DISPMV. The subroutine "TECH3", called from TACTIC, is the 370 Assembler code. Collectively, all this code together is what is referred to as TECH3 in the main text.

TACTIC and "TECH3" communicate via the 4 common blocks SPACE0, SPACE1, SPACE2, and SPACE3. The transposition table (TABLE) occupies SPACE0, WEB occupies SPACE1, MIXED1 and MIXED2 occupy SPACE3, while all the remaining data structures along with LISTMV (an area where generated moves are saved) are collected together in SPACE2.

MIXED1 and MIXED2 are both arrays of 12x64 32-bit random integers used in the hash code calculations. It was generated by the following program:

```

      INTEGER*4 MIXED1(768),MIXED2(768)
      INTEGER*4 I,J,K,X/-11/,MULT/78125005/
      INTEGER*2 II(2),JJ(2)
      EQUIVALENCE (I,II(1)),(J,JJ(1))
C
      DO 200 K=1,768
        X=X*MULT
        I=X
        X=X*MULT
        J=X
        II(2)=JJ(1)
        MIXED1(K)=I
        X=X*MULT
        I=X
        X=X*MULT
        J=X
        II(2)=JJ(1)
        MIXED2(K)=I
      200 CONTINUE
C
      WRITE(1) MIXED1,MIXED2
      STOP
      END
```

WEB is an array of 64x64 8-bit integers representing the relationship of each square (x) to every other square (y). If WEB(x,y) equals one of (2, 4, 6, 8, 10, 12, 14, 16) then it is possible to go from x to y by making $D(WEB(x,y)/2)$ displacements (ie y is in x's queen's web). If it is possible to go from x to y in a single knight's move then $WEB(x,y) = 1$. In all other cases $WEB(x,y) = 0$.

TABLE is initialised to zero for every invocation of TACTIC. MIXED1, MIXED2, and WEB are read in from the file "ZABO:TECH3*DATA" once only. They are never changed. The remaining data structures (all in SPACE2) which require initialisation are set by BLOCK DATA.

Arrays will be exceeded if the search depth proceeds beyond 48-ply. This is unlikely to happen, but not as unlikely as you might think. LISTMV has space for 1025 moves. This sometimes overflows on a depth 10 search of a complex middle-game position. Both these events are tested for and an operation exception is created at the appropriate place in the code.

```

1 C*****
2 C
3 C   MAIN
4 C
5 C   =====
6 C
7 C   IMPLICIT INTEGER*4(A - Z)
8 C
9 C   INTEGER*4 BOARD(65), MAXLEN /128/
10 C
11 C   LOGICAL*1 A(128) /128*' '/, SPOT(2)
12 C   INTEGER*2 I1 /0/
13 C   EQUIVALENCE (I1,SPOT(1))
14 C
15 C   INTEGER*2 PLUS /78/, MINUS /96/
16 C   INTEGER*2 WPAWN /215/, WKING /210/, WWHITE /213/
17 C   INTEGER*2 WBSHP /194/, WROOK /217/, WQUEEN /216/
18 C   INTEGER*2 BPAWN /151/, BKING /146/, BNITE /149/
19 C   INTEGER*2 BBSHP /130/, BROOK /153/, BQUEEN /152/
20 C
21 C   =====
22 C
23 C check PAR= for TIME and DEPTH
24 C set default
25 C
26 C   TIMMAX = 500
27 C   DTHMAX = 31
28 C
29 C   CALL PARSTR(A, LEN, MAXLEN, 0, &1020)
30 C
31 C   CALL FINDST(A, LEN, 'TIME=', 5, 1, NF, &1010)
32 C   NA = 4
33 C   CALL DTB(A(NF + 5), TIME, NA, NB, ' ', &1010)
34 C   TIMMAX = TIME * 1000
35 C 1010 CONTINUE
36 C   CALL FINDST(A, LEN, 'DEPTH=', 6, 1, NF, &1020)
37 C   NA = 2
38 C   CALL DTB(A(NF + 6), DTHMAX, NA, NB, ' ', &1020)
39 C
40 C   =====
41 C
42 C input a chess position in forsythe notation
43 C
44 C 1020 CONTINUE
45 C   READ (5,1030,END=1100) (A(I),I=1,128)
46 C 1030 FORMAT (128A1)
47 C
48 C   =====
49 C
50 C translate to internal board representation
51 C
52 C   SQ = 0
53 C   I = 0
54 C 1040 I = I + 1
55 C   SPOT(2) = A(I)
56 C

```

```

57         IF (I1 .EQ. PLUS .OR. I1 .EQ. MINUS)
58         1 GO TO 1070
59         IF (I1 .GT. 240) GO TO 1050
60     C
61         SQ = SQ + 1
62         IF (I1 .EQ. WPAWN) BOARD(SQ) = 1
63         IF (I1 .EQ. WKING) BOARD(SQ) = 2
64         IF (I1 .EQ. WNITE) BOARD(SQ) = 3
65         IF (I1 .EQ. WBSHP) BOARD(SQ) = 4
66         IF (I1 .EQ. WROOK) BOARD(SQ) = 5
67         IF (I1 .EQ. WQUEN) BOARD(SQ) = 6
68     C
69         IF (I1 .EQ. BPAWN) BOARD(SQ) = -1
70         IF (I1 .EQ. BKING) BOARD(SQ) = -2
71         IF (I1 .EQ. BNITE) BOARD(SQ) = -3
72         IF (I1 .EQ. BBSHP) BOARD(SQ) = -4
73         IF (I1 .EQ. BROOK) BOARD(SQ) = -5
74         IF (I1 .EQ. BQUEN) BOARD(SQ) = -6
75     C
76         GO TO 1040
77     1050 CONTINUE
78         NUM = I1 - 240
79         DO 1060 K = 1, NUM
80             SQ = SQ + 1
81             BOARD(SQ) = 0
82     1060 CONTINUE
83     C
84         GO TO 1040
85     1070 CONTINUE
86         IF (I1 .EQ. PLUS) BOARD(65) = 1
87         IF (I1 .EQ. MINUS) BOARD(65) = -1
88         I = I + 1
89     C
90     C =====
91     C
92     C call TACTIC to solve problem
93     C
94         WRITE (6,1080) (A(J),J=I,128)
95     1080 FORMAT ('-----', 128A1)
96     C
97         CALL TACTIC(BOARD, TIMMAX, DTHMAX)
98     C
99         WRITE (6,1090)
100     1090 FORMAT ('-----')
101         GO TO 1020
102     C
103     C =====
104     C
105     1100 CONTINUE
106         RETURN
107         END
108 C*****
109 C
110         BLOCK DATA
111     C
112     C =====

```

```

113      C
114      IMPLICIT INTEGER*4(A - Z)
115      C
116      LOGICAL*1 WEB
117      INTEGER*2 FINE, BESTCD, CASTLE, DEPTH, INCHK,
118      1      LBSAV, LSAV, LSSAV, MATER, MOVECD, SAVE,
119      2      SCORE, B, SQ, QSP, CHK1, CHK2, CHK4, D,
120      3      ROW, WT, WT2, D1, D4, D5, D5P, D6,
121      4      KINGSQ, KSQ, NITE, PAWN, ROOK, PLYP,
122      5      PLYQ, WHOMOV, PASSE, FRESH, M64, M63,
123      6      M4, N1, N2, N4, N6, N8, N10, N14, N16,
124      7      N63, N80, N128, N160, N252, FREE
125      C
126      COMMON /SPACE2/ FINE(768), BESTCD(48), CASTLE(48),
127      1      DEPTH(48), HASH1(48), HASH2(48), INCHK(48),
128      2      LBSAV(48), LSAV(48), LSSAV(48), MATER(48),
129      3      MOVE(48), MOVECD(48), SAVE(48), SCORE(48),
130      4      B(120), SQ(78), QSP(64), CHK1(42),
131      5      CHK2(42), CHK4(8), D(16), ROW(13), WT(13),
132      6      WT2(4), D1(14), D4(2), D5(2), D5P(2),
133      7      D6(2), KINGSQ(2), KSQ(2), NITE(2), PAWN(2),
134      8      ROOK(2), PARCS, QARCS, MASK, PLYP, PLYQ,
135      9      WHOMOV, PASSE, NBIG, FRESH, M64, M63, M4,
136      *      N1, N2, N4, N6, N8, N10, N14, N16, N63,
137      1      N80, N128, N160, N252, FREE, LISTMV(1025),
138      2      WEB(4096), MIXED1(768), MIXED2(768)
139      C
140      C
141      C
142      DATA FINE /768*0/
143      DATA MASK /Z0000FFFF/
144      DATA D /-22, -2, 18, 20, 22, 2, -18, -20, -42,
145      1      -24, 16, 38, 42, 24, -16, -38/
146      DATA LSSAV(3) /0/, LBSAV(3) /4104/
147      DATA SQ /0, 1, 2, 3, 4, 5, 6, 7, 2*0, 8, 9, 10,
148      1      11, 12, 13, 14, 15, 2*0, 16, 17, 18, 19, 20,
149      2      21, 22, 23, 2*0, 24, 25, 26, 27, 28, 29, 30,
150      3      31, 2*0, 32, 33, 34, 35, 36, 37, 38, 39, 2*0,
151      4      40, 41, 42, 43, 44, 45, 46, 47, 2*0, 48, 49,
152      5      50, 51, 52, 53, 54, 55, 2*0, 56, 57, 58, 59,
153      6      60, 61, 62, 63/
154      DATA B /120*14/
155      DATA ROW /2816, 2560, 2304, 2048, 1792, 0, -1,
156      1      1536, 256, 512, 768, 1024, 1280/
157      DATA CHK1 /-1, -1, 0, 0, -1, 0, 0, 0, 1, 0, 0, 1,
158      1      1, 0, -1, 0, -1, 0, -1, -1, 0, 0, 1, 0, 1, 0,
159      2      1, 0, -1, 0, -1, 0, -1, 0, 0, 1, 1, 0, 1, 0,
160      3      1, 0/
161      DATA CHK2 /-1, -1, 9*0, 1, 1, 0, -1, 0, -1, 7*0,
162      1      1, 0, 1, 0, -1, 0, -1, 7*0, 1, 0, 1, 0/
163      DATA NITE /-6, 6/, CHK4 /40, 12, 68, 12, 68, 12,
164      1      40, 12/
165      DATA PAWN /-2, 2/, D5P /32, -32/
166      DATA ROOK /-10, 10/
167      DATA D1 /6*-1, 0, 6*1, 0/
168      DATA D4 /22, -22/, D5 /20, -20/, D6 /18, -18/

```

```

169      DATA KSQ /52, 192/
170      DATA WT /9, 5, 3, 3, 18, 1, 0, 1, 18, 3, 3, 5, 9/
171      DATA WT2 /2, 2, 4, 8/
172      DATA M64 /-64/, M63 /-63/, M4 /-4/, N1 /1/, N2 /2/
173      1      , N4 /4/, N6 /6/, N8 /8/, N10 /10/, N14 /14/,
174      2      N16 /16/, N63 /63/, N80 /80/, N128 /128/,
175      3      N160 /160/, N252 /252/, NBIG /Z12000000/
176      DATA QSP /198, 196, 194, 192, 190, 188, 186, 184,
177      1      164, 144, 124, 104, 84, 64, 44, 46, 48, 50,
178      2      52, 54, 56, 58, 78, 98, 118, 138, 158, 178,
179      3      176, 174, 172, 170, 168, 166, 146, 126, 106,
180      4      86, 66, 68, 70, 72, 74, 76, 96, 116, 136,
181      5      156, 154, 152, 150, 148, 128, 108, 88, 90,
182      6      92, 94, 114, 134, 132, 130, 110, 112/
183      C
184      C      =====
185      C      END
186      C*****
187      C
188      SUBROUTINE TACTIC(BOARD, TIMMAX, DTHMAX)
189      C
190      C      =====
191      C
192      IMPLICIT INTEGER*4(A - Z)
193      C
194      COMMON /SPACE0/ TABLE(131072)
195      C
196      LOGICAL*1 WEB
197      C*      COMMON /SPACE1/ WEB(4096)
198      C
199      C*      COMMON /SPACE3/ MIXED1(768), MIXED2(768)
200      C
201      INTEGER*2 FINE, BESTCD, CASTLE, DEPTH, INCHK,
202      1      LBSAV, LSAV, LSSAV, MATER, MOVECD, SAVE,
203      2      SCORE, B, SQ, QSP, CHK1, CHK2, CHK4, D,
204      3      ROW, WT, WT2, D1, D4, D5, D5P, D6,
205      4      KINGSQ, KSQ, NITE, PAWN, ROOK, PLYP,
206      5      PLYQ, WHOMOV, PASSE, FRESH, M64, M63,
207      6      M4, N1, N2, N4, N6, N8, N10, N14, N16,
208      7      N63, N80, N128, N160, N252, FREE
209      C
210      COMMON /SPACE2/ FINE(768), BESTCD(48), CASTLE(48),
211      1      DEPTH(48), HASH1(48), HASH2(48), INCHK(48),
212      2      LBSAV(48), LSAV(48), LSSAV(48), MATER(48),
213      3      MOVE(48), MOVECD(48), SAVE(48), SCORE(48),
214      4      B(120), SQ(78), QSP(64), CHK1(42),
215      5      CHK2(42), CHK4(8), D(16), ROW(13), WT(13),
216      6      WT2(4), D1(14), D4(2), D5(2), D5P(2),
217      7      D6(2), KINGSQ(2), KSQ(2), NITE(2), PAWN(2),
218      8      ROOK(2), PARCS, QARCS, MASK, PLYP, PLYQ,
219      9      WHOMOV, PASSE, NBIG, FRESH, M64, M63, M4,
220      *      N1, N2, N4, N6, N8, N10, N14, N16, N63,
221      1      N80, N128, N160, N252, FREE, LISTMV(1025),
222      2      WEB(4096), MIXED1(768), MIXED2(768)
223      C
224      C      -----

```

```

225 C
226     DIMENSION BOARD(65)
227     LOGICAL*1 CHAR(8), DELTA /Z26/, SECOND / .FALSE. /
228 C
229     QS(K) = K + 21 + (K - 1) / 8 * 2
230 C
231 C     =====
232 C
233     IF (SECOND) GO TO 1010
234     CALL FTNCMD('ASSIGN 1=AZPG:TECH3*DATA;')
235     READ (1) WEB, MIXED1, MIXED2
236     SECOND = .TRUE.
237 1010 CONTINUE
238 C
239 C     -----
240 C 'zero' transposition table
241     KMAX = MASK * 2 + 2
242     DO 1020 K = 1, KMAX
243 1020 TABLE(K) = 0
244 C translate position into common block representation
245     MATER(3) = 0
246     DO 1030 K = 1, 64
247     B(QS(K)) = BOARD(K) * 2
248     X = BOARD(K)
249     IF (X .EQ. 0) GO TO 1030
250     IF (X .EQ. 2) KINGSQ(2) = QS(K) * 2
251     IF (X .EQ. - 2) KINGSQ(1) = QS(K) * 2
252     IF (X .GT. 0) MATER(3) = MATER(3) + WT(X + 7)
253     IF (X .LT. 0) MATER(3) = MATER(3) - WT(X + 7)
254 1030 CONTINUE
255 C
256     WHOMOV = BOARD(65)
257     MATER(3) = MATER(3) * WHOMOV
258     CASTLE(2) = 3
259     CASTLE(3) = 3
260     PASSE = 0
261     PLYP = 6
262     HASH1(3) = MIXED1(2)
263     HASH2(3) = MIXED2(2)
264 C
265 C     =====
266 C initialize alpha-beta iteration loop
267 1040 CONTINUE
268     CALL TIME(1, 0, TIM0)
269     PARCS = 0
270     QARCS = 0
271     DEPTH(3) = 1
272     VALUE = MATER(3)
273 C
274 C     -----
275 C
276 1050 CONTINUE
277 C increment maximum search depth
278     DEPTH(3) = DEPTH(3) + 1
279 C set window
280     SCORE(1) = VALUE - 1

```



```

281         SCORE(2) = -VALUE - 1
282     C
283         CALL TECH3
284     C
285         IF (SCORE(3) .EQ. VALUE) GO TO 1060
286     C score was outside window
287     C open window and repeat search
288         SCORE(1) = -63
289         SCORE(2) = -63
290     C
291         CALL TECH3
292     C
293         VALUE = SCORE(3)
294     C
295     C -----
296     C
297     C end of iteration
298     C output statistics
299     1060 CONTINUE
300         L = LSSAV(4) / 4 - BESTCD(3)
301         IF (L .GT. LSSAV(3)/4) GO TO 1070
302         L = LBSAV(4) / 4 - BESTCD(3) + 63
303         IF (L .LT. LBSAV(3)/4) GO TO 1070
304         BESTMV = 0
305         GO TO 1080
306     1070 BESTMV = LISTMV(L)
307         IF (BESTMV .LT. 0) BESTMV = -BESTMV
308     1080 CONTINUE
309     C
310         CALL TIME(1, 0, TIM)
311         DSCORE = SCORE(3) - MATER(3)
312         DTIME = TIM - TIM0
313         FOM = LAND(SHFTR(BESTMV,8),255) / 2
314         TOM = LAND(BESTMV,255) / 2
315         FLAG = LAND(SHFTR(BESTMV,16),255) / 2
316     C
317         IF (DTIME .EQ. 0) DTIME = 1
318         SPEED = (PARCS + QARCS + DTIME/2) / DTIME
319         BDF = B(FOM) / 2
320         BDT = B(TOM) / 2
321         CALL DISPMV(FOM, TOM, FLAG, BDF, BDT, CHAR)
322     C
323         WRITE (6,1090) DEPTH(3), DELTA, DSCORE, DTIME,
324             1PARCS, QARCS, SPEED, (CHAR(I),I=1,8)
325     1090 FORMAT ('0D=', I2, A3, 'S=', I3, ' T=', I7,
326         1      ' N=', I8, 'p', I8, 'q' N/T=', I2,
327         2      ' MV=', 8A1)
328     C
329     C continue iterating as long as time and depth
330     C limits are not exceeded
331         IF (DEPTH(3) .LT. DTHMAX .AND. DTIME .LT. TIMMAX)
332         1 GO TO 1050
333     C
334     C =====
335     C
336     RETURN

```

```

337          END
338 C*****
339 C
340     SUBROUTINE DISPMV(FROM, TO, FLAG, BDF, BDT, CHAR)
341 C
342 C     =====
343 C
344     IMPLICIT INTEGER*4(A - Z)
345 C
346     LOGICAL*1 BIGO /'O'/, BLANK /' '/, CROSS /'x'/,
347 1           DASH /'-'/, EQUALS /'='/,
348 2           CPIECE(6) /' ','K','N','B','R','Q'/
349 3           , CFIL(8) /'a','b','c','d','e','f',
350 4           'g','h'/, CRANK(8) /'8','7','6',
351 5           '5','4','3','2','1'/, CHAR(8)
352 C
353 C     =====
354 C
355 C translate move to external representation
356 C
357     DO 1010 K = 1, 8
358 1010 CHAR(K) = BLANK
359     IF (FROM .EQ. 0) GO TO 1030
360     IF (FLAG .EQ. 7) GO TO 1020
361 C
362     CHAR(1) = CPIECE(IABS(BDF))
363     CHAR(2) = CFIL(FROM - FROM/10*10 - 1)
364     CHAR(3) = CRANK(FROM/10 - 1)
365     CHAR(4) = DASH
366     IF (FLAG .EQ. 2 .OR. BDT .NE. 0) CHAR(4) = CROSS
367     CHAR(5) = CFIL(TO - TO/10*10 - 1)
368     CHAR(6) = CRANK(TO/10 - 1)
369 C
370     IF (FLAG .LT. 3 .OR. FLAG .GT. 7) GO TO 1030
371     CHAR(7) = EQUALS
372     CHAR(8) = CPIECE(FLAG)
373     GO TO 1030
374 C
375 1020 CONTINUE
376     CHAR(1) = BIGO
377     CHAR(2) = DASH
378     CHAR(3) = BIGO
379     IF (FROM .LT. TO) GO TO 1030
380     CHAR(4) = DASH
381     CHAR(5) = BIGO
382 C
383 1030 CONTINUE
384     RETURN
385     END

```

1 * 370 Assembler code for TECH3
 2
 3 FINE EQU 0
 4
 5 BESTCD EQU 1534
 6 CASTLE EQU 1630
 7 DEPTH EQU 1726

8	HASH1	EQU	1820
9	HASH2	EQU	2012
10	INCHK	EQU	2206
11	LBSAV	EQU	2302
12	LSAV	EQU	2322
13	LSSAV	EQU	2494
14	MATER	EQU	2590
15	MOVE	EQU	2684
16	MOVECD	EQU	2878
17	SAVE	EQU	2974
18	SCORE	EQU	3070
19			
20	B	EQU	3166
21	SQ	EQU	3364
22	QSP	EQU	3562
23			
24	CHK1	EQU	3692
25	CHK2	EQU	3776
26	CHK4	EQU	3858
27	D	EQU	3874
28	ROW	EQU	3920
29	WT	EQU	3946
30	WT2	EQU	3954
31	D1	EQU	3980
32	D4	EQU	3997
33	D5	EQU	4001
34	D5P	EQU	4005
35	D6	EQU	4009
36	KINGSQ	EQU	4013
37	KSQ	EQU	4017
38	NITE	EQU	4021
39	PAWN	EQU	4025
40	ROOK	EQU	4029
41	PARCS	EQU	4032
42	QARCS	EQU	4036
43	MASK	EQU	4040
44	PLYP	EQU	4044
45	PLYQ	EQU	4046
46	WHOMOV	EQU	4048
47	PASSE	EQU	4050
48			
49	NBIG	EQU	4052
50	FRESH	EQU	4056
51	M64	EQU	4058
52	M63	EQU	4060
53	N1	EQU	4064
54	N2	EQU	4066
55	N4	EQU	4068
56	N6	EQU	4070
57	N8	EQU	4072
58	N10	EQU	4074
59	N14	EQU	4076
60	N63	EQU	4080
61	N80	EQU	4082
62	N160	EQU	4086
63			

```

64  LISTMV  EQU  4088      area for move lists
65  MIXED1  EQU  0        numbers for hash coding
66  MIXED2  EQU  3072
67  WEB     EQU  0        relation between squares
68  TABLE  EQU  0        transposition table
69
70  *        The following refers to registers
71
72  *        Register 15 = TECH3 = entry point
73  *        Register 14 = A(SPACE2)
74  *        Register 13 = A(SPACE0)
75  *        or offset to A(SPACE2)
76  *        or return address
77
78  PLY      EQU  12
79  WHO      EQU  11
80  PLY1     EQU  10      shft1(ply,1)
81  NWHO     EQU  9       not who
82
83  FOM      EQU  8
84  TOM      EQU  7
85  FLAG     EQU  6
86  DELTA    EQU  6
87
88  LB       EQU  5
89  C        EQU  5
90  L        EQU  5
91  FOX      EQU  5
92  INDEX    EQU  5
93  LS       EQU  4
94  LMAX     EQU  4
95  DEPTH0   EQU  4
96  VEC      EQU  4
97  PEACE    EQU  4
98  S        EQU  4
99  X        EQU  3
100 MAX      EQU  3
101 M        EQU  2
102 MOSS      EQU  2
103 MASS     EQU  1
104 PIECE    EQU  1
105
106 *ENTER
107 TECH3     CSECT
108           USING  *,15      reg 15 never changed
109           EXTRN  SPACE0
110  *        EXTRN  SPACE1
111           EXTRN  SPACE2
112  *        EXTRN  SPACE3
113           B      A1000
114           DC     XL1'07'
115           DC     CL7'TECH3
116 RETURN    DC     F'0'
117 ASpace0   DC     A(SPACE0)
118 ASpace1   DC     A(SPACE2+8192)
119 ASpace2   DC     A(SPACE2)

```

```

120  ASPACE3  DC      A(SPACE2+12288)
121  A1000    STM     14,12,12(13)  save registers
122          ST      13,RETURN
123
124          L        14,ASPACE2      reg 14 never changed
125
126          LH       PLY,PLYP(14)
127          LH       WHO,WHOMOV(14)
128          LR       PLY1,PLY
129          SLL      PLY1,1(0)
130          LCR      NWHO,WHO
131          XR       0,0              reg 0 (=0) never changed
132
133          LR       C,NWHO           root position in check?
134          LH       S,KINGSQ(WHO,14)
135          BAL      13,A2070
136          STH      C,INCHK(PLY,14)
137          B        A1120           initialization over
138  *                               goto initial entry point
139
140  *QSEARCH
141  A1020     LH      1,MATER(PLY,14)
142          STH      1,SCORE(PLY,14)
143
144          LCR      2,1
145          CH       2,SCORE-2(PLY,14)
146          BNH      A1580
147          LA       2,17(0,0)
148          AR       2,1
149          CH       2,SCORE-4(PLY,14)
150          BNH      A1580
151          STH      PLY,PLYQ(14)
152
153  A1030     CH      1,SCORE-4(PLY,14)
154          BNL      A1031
155          LH       2,SCORE-4(PLY,14)
156          STH      2,SCORE(PLY,14)
157  A1031     LA      PLY,2(0,PLY)
158          LA      1,96(0,0)
159          CR      1,PLY
160          BL      A1031+2
161          LA      PLY1,4(0,PLY1)
162          LCR      WHO,WHO
163          LCR      NWHO,NWHO
164
165          BAL      13,A1620
166          LH      LB,LBSAV-2(PLY,14)
167          STH      LB,LBSAV(PLY,14)
168  A1040     XR      MAX,MAX
169          XR      LMAX,LMAX
170          LH      L,LSSAV-2(PLY,14)
171  A1050     LA      L,4(0,L)
172          CH      L,LSSAV(PLY,14)
173          BH      A1060
174          C        MAX,LISTMV(L,14)
175          BNL      A1050

```

176		L	MAX,LISTMV(L,14)
177		LR	LMAX,L
178		B	A1050
179			
180	A1060	LTR	LMAX,LMAX
181		BZ	A1110
182		C	MAX,NBIG(14)
183		BL	A1070
184			
185		L	1,QARCS(14)
186		SH	1,N1(14)
187		ST	1,QARCS(14)
188		SH	PLY,N2(14)
189		SH	PLY1,N4(14)
190		LCR	WHO,WHO
191		LCR	NWHO,NWHO
192		BAL	13,A2010
193		B	A1040
194			
195	A1070	ST	MAX,MOVE(PLY1,14)
196		ST	0,LISTMV(LMAX,14)
197			
198		BAL	13,A1950
199		L	1,QARCS(14)
200		AH	1,N1(14)
201		ST	1,QARCS(14)
202		LH	1,MATER(PLY,14)
203		STH	1,SCORE(PLY,14)
204		LCR	2,1
205		CH	2,SCORE-2(PLY,14)
206		BH	A1030
207		BAL	13,A2010
208		B	A1110
209			
210	A1090	BAL	13,A2010
211		LH	1,SCORE(PLY,14)
212		LCR	2,1
213		CH	2,SCORE-2(PLY,14)
214		BNH	A1040
215		STH	2,SCORE-2(PLY,14)
216		CH	1,SCORE-4(PLY,14)
217		BH	A1040
218			
219	A1110	SH	PLY,N2(14)
220		SH	PLY1,N4(14)
221		LCR	WHO,WHO
222		LCR	NWHO,NWHO
223		CH	PLY,PLYQ(14)
224		BH	A1090
225		B	A1580
226			
227	*TABLE		
228	A1120	STH	WHO,FRESH(14)
229		L	INDEX,HASH1(PLY1,14)
230		N	INDEX,MASK(14)
231		SLL	INDEX,3(0)

232		L	13,ASPACE0
233			
234		L	1,HASH2(PLY1,14)
235		C	1,TABLE(INDEX,13)
236		BNE	A1150
237			
238		STH	0,FRESH(14)
239		L	2,TABLE+4(INDEX,13)
240		SRDA	2,6(0)
241		SRL	3,26(0)
242		LR	DEPTH0,3
243		SRDA	2,2(0)
244		SRL	3,30(0)
245		LR	FLAG,3
246		SRDA	2,8(0)
247		SRL	3,24(0)
248		STH	3,BESTCD(PLY,14)
249		STH	2,SCORE(PLY,14)
250			
251		CH	DEPTH0,DEPTH(PLY,14)
252		BL	A1160
253		CH	FLAG,N1(14)
254		BL	A1610
255		BE	A1140
256			
257		LH	1,SCORE(PLY,14)
258		LCR	2,1
259		CH	2,SCORE-2(PLY,14)
260		BNH	A1610
261		CH	1,SCORE-4(PLY,14)
262		BNL	A1170
263		LH	1,SCORE-4(PLY,14)
264		STH	1,SCORE(PLY,14)
265		B	A1170
266			
267	A1140	LH	1,SCORE(PLY,14)
268		CH	1,SCORE-4(PLY,14)
269		BNH	A1610
270		LCR	2,1
271		CH	2,SCORE-2(PLY,14)
272		BNH	A1160
273		STH	2,SCORE-2(PLY,14)
274		B	A1160
275			
276	A1150	STH	0,BESTCD(PLY,14)
277		LH	1,SCORE-4(PLY,14)
278		STH	1,SCORE(PLY,14)
279		LA	1,63(0,0)
280		N	1,TABLE+4(INDEX,13)
281		CH	1,DEPTH(PLY,14)
282		BH	A1180
283			
284	A1160	LH	1,SCORE-4(PLY,14)
285		STH	1,SCORE(PLY,14)
286			
287	A1170	L	1,HASH2(PLY1,14)

288		ST	1, TABLE(INDEX, 13)
289		LH	1, DEPTH(PLY, 14)
290		ST	1, TABLE+4(INDEX, 13)
291			
292	*PSEARCH		
293	A1180	LA	PLY, 2(0, PLY)
294		LA	1, 96(0, 0)
295		CR	1, PLY
296		BL	A1180+2
297		LA	PLY1, 4(0, PLY1)
298		LCR	WHO, WHO
299		LCR	NWHO, NWHO
300		LH	1, M64(14)
301		STH	1, SCORE(PLY, 14)
302			
303		BAL	13, A1620
304			
305		STH	0, LSAV(PLY, 14)
306		CH	0, FRESH(14)
307		BNE	A1220
308		LH	1, BESTCD-2(PLY, 14)
309		STH	1, MOVECD(PLY, 14)
310		SLL	1, 2(0)
311		LH	L, LSSAV(PLY, 14)
312		SR	L, 1
313		CH	L, LSSAV-2(PLY, 14)
314		BH	A1210
315			
316		LH	L, LBSAV(PLY, 14)
317		SR	L, 1
318		LA	L, 252(0, L)
319		CH	L, LBSAV-2(PLY, 14)
320		BNL	A1220
321			
322	A1210	L	1, LISTMV(L, 14)
323		ST	1, MOVE(PLY1, 14)
324		LCR	1, 1
325		ST	1, LISTMV(L, 14)
326		B	A1280
327			
328	A1220	CH	0, LSAV(PLY, 14)
329		BL	A1260
330		XR	MAX, MAX
331		XR	LMAX, LMAX
332		LH	L, LSSAV-2(PLY, 14)
333	A1230	LA	L, 4(0, L)
334		CH	L, LSSAV(PLY, 14)
335		BH	A1240
336		C	MAX, LISTMV(L, 14)
337		BNL	A1230
338		L	MAX, LISTMV(L, 14)
339		LR	LMAX, L
340		B	A1230
341			
342	A1240	LTR	LMAX, LMAX
343		BP	A1250

344		LH	1,LBSAV-2(PLY,14)
345		STH	1,LSAV(PLY,14)
346		B	A1260
347			
348	A1250	ST	MAX,MOVE(PLY1,14)
349		LCR	MAX,MAX
350		ST	MAX,LISTMV(LMAX,14)
351		LH	1,LSSAV(PLY,14)
352		SR	1,LMAX
353		SRL	1,2(0)
354		STH	1,MOVECD(PLY,14)
355		B	A1280
356			
357	A1260	LH	1,LSAV(PLY,14)
358		SH	1,N4(14)
359		CH	1,LBSAV(PLY,14)
360		BL	A1270
361		STH	1,LSAV(PLY,14)
362		C	0,LISTMV(1,14)
363		BH	A1260
364		L	2,LISTMV(1,14)
365		ST	2,MOVE(PLY1,14)
366		SH	1,LBSAV(PLY,14)
367		LCR	1,1
368		LA	1,252(0,1)
369		SRL	1,2(0)
370		STH	1,MOVECD(PLY,14)
371		B	A1280
372			
373	A1270	LH	1,SCORE-2(PLY,14)
374		CH	1,M63(14)
375		BNE	A1271
376		LA	2,31(0,0)
377		STH	2,DEPTH-2(PLY,14)
378	A1271	LH	2,SCORE(PLY,14)
379		CH	2,M64(14)
380		BNE	A1272
381		CH	0,INCHK-2(PLY,14)
382		BNE	A1272
383		STH	0,SCORE-2(PLY,14)
384	A1272	XR	FLAG,FLAG
385		LH	1,SCORE-2(PLY,14)
386		CH	1,SCORE-6(PLY,14)
387		BH	A1600
388		LA	FLAG,1(0,0)
389		B	A1600
390			
391	*LEGAL		
392	A1280	BAL	13,A1950
393			
394		L	13,ASPACE1
395		XR	VEC,VEC
396			
397		CH	FLAG,N4(14)
398		BE	A1490
399		CH	FLAG,N14(14)

400		BE	A1430
401		BH	A1330
402		LH	FOX,KINGSQ(NWHO,14)
403		CH	0,INCHK-2(PLY,14)
404		BNE	A1320
405			
406		LH	1,SQ(FOM,14)
407		SLL	1,6(0)
408		AH	1,SQ(FOX,14)
409		IC	VEC,WEB(1,13)
410			
411		CH	VEC,N1(14)
412		BNH	A1350
413			
414	A1300	AH	FOX,D(VEC,14)
415		CH	0,B(FOX,14)
416		BE	A1300
417		LH	1,CHK4(VEC,14)
418		AH	1,B(FOX,14)
419		CH	WHO,CHK2(1,14)
420		BNE	A1350
421		B	A1520
422			
423	A1320	LH	1,SQ(TOM,14)
424		SLL	1,6(0)
425		AH	1,SQ(FOX,14)
426		IC	VEC,WEB(1,13)
427			
428		LTR	VEC,VEC
429		BZ	A1520
430			
431	A1330	LR	C,WHO
432		LH	S,KINGSQ(NWHO,14)
433		BAL	13,A2070
434		LTR	C,C
435		BNZ	A1520
436		L	13,ASPACE1
437			
438	A1350	LH	FOX,KINGSQ(WHO,14)
439		LH	1,SQ(FOM,14)
440		SLL	1,6(0)
441		AH	1,SQ(FOX,14)
442		IC	VEC,WEB(1,13)
443			
444		CH	VEC,N1(14)
445		BNH	A1380
446			
447	A1360	AH	FOX,D(VEC,14)
448		CH	0,B(FOX,14)
449		BE	A1360
450		LH	1,CHK4(VEC,14)
451		AH	1,B(FOX,14)
452		CH	NWHO,CHK2(1,14)
453		BNE	A1380
454			
455	A1370	STH	WHO,INCHK(PLY,14)

456		B	A1530
457			
458	A1380	LH	FOX,KINGSQ(WHO,14)
459		LH	1,SQ(TOM,14)
460		SLL	1,6(0)
461		AH	1,SQ(FOX,14)
462		IC	VEC,WEB(1,13)
463			
464		CH	VEC,N1(14)
465		BH	A1400
466		BL	A1390
467			
468		LH	1,NITE(NWHO,14)
469		CH	1,B(TOM,14)
470		BE	A1370
471			
472	A1390	STH	0,INCHK(PLY,14)
473		B	A1530
474			
475	A1400	AH	FOX,D(VEC,14)
476		CH	0,B(FOX,14)
477		BE	A1420
478		LH	1,CHK4(VEC,14)
479		AH	1,B(FOX,14)
480		CH	NWHO,CHK1(1,14)
481		BNE	A1390
482		B	A1370
483			
484	A1420	AH	FOX,D(VEC,14)
485		CH	0,B(FOX,14)
486		BE	A1420
487		LH	1,CHK4(VEC,14)
488		AH	1,B(FOX,14)
489		CH	NWHO,CHK2(1,14)
490		BNE	A1390
491		B	A1370
492			
493	A1430	CH	0,INCHK-2(PLY,14)
494		BNE	A1520
495		LR	C,WHO
496		LR	S,TOM
497		BAL	13,A2070
498		LTR	C,C
499		BNZ	A1520
500		LR	C,WHO
501		CR	FOM,TOM
502		BH	A1431
503		SH	S,N2(14)
504		B	A1432
505	A1431	AH	S,N2(14)
506	A1432	BAL	13,A2070
507		LTR	C,C
508		BNZ	A1520
509		B	A1501
510			
511	A1490	LR	C,WHO

512		LH	S,KINGSQ(NWHO,14)
513		BAL	13,A2070
514		LTR	C,C
515		BNZ	A1520
516			
517	A1501	LR	C,NWHO
518		LH	S,KINGSQ(WHO,14)
519		BAL	13,A2070
520		STH	C,INCHK(PLY,14)
521		B	A1530
522			
523	A1520	BAL	13,A2010
524		B	A1220
525			
526	A1530	L	1,PARCS(14)
527		AH	1,N1(14)
528		ST	1,PARCS(14)
529			
530		LH	1,DEPTH-2(PLY,14)
531		CH	0,INCHK(PLY,14)
532		BNE	A1531
533			
534		CH	0,INCHK-2(PLY,14)
535		BNE	A15305
536		LH	2,MATER(PLY,14)
537		LCR	2,2
538		CH	2,SCORE-2(PLY,14)
539		BH	A15305
540		SH	1,N1(14)
541			
542	A15305	SH	1,N1(14)
543	A1531	STH	1,DEPTH(PLY,14)
544		LTR	1,1
545		BNP	A1020
546			
547	*ENCODE		
548		L	13,ASPACE3
549			
550		L	1,HASH1-4(PLY1,14)
551		L	2,HASH2-4(PLY1,14)
552		LH	FOM,SQ(FOM,14)
553		SLL	FOM,2(0)
554		LH	PEACE,B(TOM,14)
555		LH	TOM,SQ(TOM,14)
556		SLL	TOM,2(0)
557			
558		LH	X,ROW(PEACE,14)
559		AR	X,FOM
560		X	1,MIXED1(X,13)
561		X	2,MIXED2(X,13)
562			
563		LH	X,ROW(PEACE,14)
564		AR	X,TOM
565		X	1,MIXED1(X,13)
566		X	2,MIXED2(X,13)
567			

568		X	1,MIXED1(0,13)
569		X	2,MIXED2(0,13)
570			
571		CH	0,SAVE(PLY,14)
572		BE	A1532
573		LH	X,SAVE(PLY,14)
574		LH	X,ROW(X,14)
575		AR	X,TOM
576		X	1,MIXED1(X,13)
577		X	2,MIXED2(X,13)
578			
579	A1532	CH	FLAG,N2(14)
580		BNH	A1561
581		CH	FLAG,N14(14)
582		BE	A1550
583		BH	A1561
584		CH	FLAG,N4(14)
585		BH	A1540
586			
587		LH	X,PAWN(WHO,14)
588		LH	X,ROW(X,14)
589		AR	X,TOM
590		AH	X,D5P(WHO,14)
591		X	1,MIXED1(X,13)
592		X	2,MIXED2(X,13)
593		B	A1561
594			
595	A1540	LH	X,ROW(PEACE,14)
596		AR	X,FOM
597		X	1,MIXED1(X,13)
598		X	2,MIXED2(X,13)
599			
600		LH	X,PAWN(NWHO,14)
601		LH	X,ROW(X,14)
602		AR	X,FOM
603		X	1,MIXED1(X,13)
604		X	2,MIXED2(X,13)
605		B	A1561
606			
607	A1550	LH	PEACE,ROOK(NWHO,14)
608		LH	PEACE,ROW(PEACE,14)
609		AR	PEACE,TOM
610		CR	TOM,FOM
611		BL	A1560
612			
613		LR	X,PEACE
614		SH	X,N4(14)
615		X	1,MIXED1(X,13)
616		X	2,MIXED2(X,13)
617			
618		LR	X,PEACE
619		AH	X,N4(14)
620		X	1,MIXED1(X,13)
621		X	2,MIXED2(X,13)
622		B	A1561
623			

```

624      A1560      LR      X,PEACE
625                      AH      X,N4(14)
626                      X      1,MIXED1(X,13)
627                      X      2,MIXED2(X,13)
628
629                      LR      X,PEACE
630                      SH      X,N8(14)
631                      X      1,MIXED1(X,13)
632                      X      2,MIXED2(X,13)
633
634      A1561      ST      1,HASH1(PLY1,14)
635                      ST      2,HASH2(PLY1,14)
636                      B      A1120
637
638      *PSEARCH continued
639      A1580      BAL      13,A2010
640                      LH      1,SCORE(PLY,14)
641                      LCR      1,1
642                      CH      1,SCORE-2(PLY,14)
643                      BNH      A1220
644
645                      STH      1,SCORE-2(PLY,14)
646                      LH      2,MOVECD(PLY,14)
647                      STH      2,BESTCD-2(PLY,14)
648
649                      LCR      2,1
650                      CH      2,SCORE-4(PLY,14)
651                      BH      A1220
652                      CH      1,N63(14)
653                      BNE      A1581
654                      LA      2,31(0,0)
655                      STH      2,DEPTH-2(PLY,14)
656      A1581      LA      FLAG,2(0,0)
657
658      A1600      SH      PLY,N2(14)
659                      SH      PLY1,N4(14)
660                      LCR      WHO,WHO
661                      LCR      NWHO,NWHO
662
663                      L      13,ASPACE0
664                      L      INDEX,HASH1(PLY1,14)
665                      N      INDEX,MASK(14)
666                      SLL      INDEX,3(0)
667                      LA      1,63(0,0)
668                      N      1,TABLE+4(INDEX,13)
669                      CH      1,DEPTH(PLY,14)
670                      BH      A1610
671
672                      L      1,HASH2(PLY1,14)
673                      ST      1,TABLE(INDEX,13)
674                      LH      1,SCORE(PLY,14)
675                      SLL      1,8(0)
676                      AH      1,BESTCD(PLY,14)
677                      SLL      1,2(0)
678                      AR      1,FLAG
679                      SLL      1,6(0)

```

```

680          AH      1,DEPTH(PLY,14)
681          ST      1,TABLE+4(INDEX,13)
682
683      A1610      CH      PLY,PLYP(14)
684          BH      A1580
685
686      *EXIT
687          L        13,RETURN
688          LM      14,12,12(13)  restore registers
689          XR      15,15          return code = 0
690          BR      14            return to caller
691
692      *
693          pseudo-subroutines
694
695      *MOVGEN
696      A1620      LH      LS,LSSAV-2(PLY,14)
697          LH      LB,LBSAV-2(PLY,14)
698          LA      X,128(0,0)
699
700      A1630      LH      FOM,QSP(X,14)
701          LH      PIECE,B(FOM,14)
702          CH      NWHO,D1(PIECE,14)
703          BE      A1631
704      A1880      SH      X,N2(14)
705          BP      A1630
706
707          STH      0,PASSE(14)
708          LH      X,CASTLE-4(PLY,14)
709          STH      X,CASTLE(PLY,14)
710          LTR      X,X
711          BZ      A1940
712          LH      FOM,KINGSQ(NWHO,14)
713          CH      FOM,KSQ(NWHO,14)
714          BE      A1890
715          STH      0,CASTLE(PLY,14)
716          B        A1940
717
718      A1890      CH      X,N2(14)
719          BE      A1920
720          LH      PIECE,ROOK(NWHO,14)
721          CH      PIECE,B+6(FOM,14)
722          BE      A1900
723          SH      X,N1(14)
724          STH      X,CASTLE(PLY,14)
725          B        A1910
726
727      A1900      C        0,B+2(FOM,14)
728          BNE      A1910
729          SH      LB,N4(14)
730          LA      FLAG,14(0,0)
731          LA      TOM,4(0,FOM)
732          ST      TOM,LISTMV(LB,14)
733          STC      FOM,LISTMV+2(LB,14)
734          STC      FLAG,LISTMV+1(LB,14)
735
736      A1910      CH      X,N2(14)

```

736		BL	A1940
737	A1920	LH	PIECE, ROOK (NWHO, 14)
738		CH	PIECE, B-8 (FOM, 14)
739		BE	A1930
740		SH	X, N2 (14)
741		STH	X, CASTLE (PLY, 14)
742		B	A1940
743			
744	A1930	CH	0, B-2 (FOM, 14)
745		BNE	A1940
746		C	0, B-6 (FOM, 14)
747		BNE	A1940
748		SH	LB, N4 (14)
749		LA	FLAG, 14 (0, 0)
750		LR	TOM, FOM
751		SH	TOM, N4 (14)
752		ST	TOM, LISTMV (LB, 14)
753		STC	FOM, LISTMV+2 (LB, 14)
754		STC	FLAG, LISTMV+1 (LB, 14)
755			
756	A1940	STH	LS, LSSAV (PLY, 14)
757		STH	LB, LBSAV (PLY, 14)
758		CR	LS, LB
759		BNL	A1940+2
760		BR	13
761			
762	A1631	LPR	PIECE, PIECE
763		CH	PIECE, N2 (14)
764		BNE	A1632
765			
766		LTR	WHO, WHO
767		BP	A1781
768		CH	FOM, N80 (14)
769		BL	A1820
770		B	A1782
771			
772	A1781	CH	FOM, N160 (14)
773		BH	A1820
774			
775	A1782	LR	TOM, FOM
776		AH	TOM, D5 (NWHO, 14)
777		CH	0, B (TOM, 14)
778		BNE	A1790
779		SH	LB, N4 (14)
780		ST	TOM, LISTMV (LB, 14)
781		STC	FOM, LISTMV+2 (LB, 14)
782			
783		LTR	WHO, WHO
784		BP	A1783
785		CH	FOM, N160 (14)
786		BL	A1790
787		B	A1784
788			
789	A1783	CH	FOM, N80 (14)
790		BH	A1790
791			

792	A1784	AH	TOM,D5(NWHO,14)
793		CH	0,B(TOM,14)
794		BNE	A1790
795		LA	FLAG,2(0,0)
796		CH	LB,N1(14)
797		ST	TOM,LISTMV(LB,14)
798		STC	FOM,LISTMV+2(LB,14)
799		STC	FLAG,LISTMV+1(LB,14)
800			
801	A1790	LR	TOM,FOM
802		AH	TOM,D4(NWHO,14)
803		LH	PIECE,B(TOM,14)
804		CH	WHO,D1(PIECE,14)
805		BNE	A1800
806		LA	LS,4(0,LS)
807		LH	MOSS,WT(PIECE,14)
808		ST	TOM,LISTMV(LS,14)
809		STC	FOM,LISTMV+2(LS,14)
810		STC	MOSS,LISTMV(LS,14)
811			
812	A1800	LR	TOM,FOM
813		AH	TOM,D6(NWHO,14)
814		LH	PIECE,B(TOM,14)
815		CH	WHO,D1(PIECE,14)
816		BNE	A1810
817		LA	LS,4(0,LS)
818		LH	MOSS,WT(PIECE,14)
819		ST	TOM,LISTMV(LS,14)
820		STC	FOM,LISTMV+2(LS,14)
821		STC	MOSS,LISTMV(LS,14)
822			
823	A1810	CH	0,PASSE(14)
824		BE	A1880
825		CH	TOM,PASSE(14)
826		BE	A1811
827		LR	TOM,FOM
828		AH	TOM,D4(NWHO,14)
829		CH	TOM,PASSE(14)
830		BNE	A1880
831			
832	A1811	LA	LS,4(0,LS)
833		LA	MOSS,1(0,0)
834		LA	FLAG,4(0,0)
835		ST	TOM,LISTMV(LS,14)
836		STC	FOM,LISTMV+2(LS,14)
837		STC	FLAG,LISTMV+1(LS,14)
838		STC	MOSS,LISTMV(LS,14)
839		B	A1880
840			
841	A1820	LR	TOM,FOM
842		AH	TOM,D5(NWHO,14)
843		CH	0,B(TOM,14)
844		BNE	A1840
845		LA	FLAG,12(0,0)
846			
847	A1830	LA	LS,4(0,LS)

848		LH	MOSS,WT2(FLAG,14)
849		ST	TOM,LISTMV(LS,14)
850		STC	FOM,LISTMV+2(LS,14)
851		STC	FLAG,LISTMV+1(LS,14)
852		STC	MOSS,LISTMV(LS,14)
853		SH	FLAG,N2(14)
854		CH	FLAG,N6(14)
855		BNL	A1830
856			
857	A1840	LR	TOM,FOM
858		AH	TOM,D4(NWHO,14)
859		LH	PIECE,B(TOM,14)
860		CH	WHO,D1(PIECE,14)
861		BNE	A1860
862		LA	FLAG,12(0,0)
863			
864	A1850	LA	LS,4(0,LS)
865		LH	MOSS,WT(PIECE,14)
866		AH	MOSS,WT2(FLAG,14)
867		ST	TOM,LISTMV(LS,14)
868		STC	FOM,LISTMV+2(LS,14)
869		STC	FLAG,LISTMV+1(LS,14)
870		STC	MOSS,LISTMV(LS,14)
871		SH	FLAG,N2(14)
872		CH	FLAG,N6(14)
873		BNL	A1850
874			
875	A1860	LR	TOM,FOM
876		AH	TOM,D6(NWHO,14)
877		LH	PIECE,B(TOM,14)
878		CH	WHO,D1(PIECE,14)
879		BNE	A1880
880		LA	FLAG,12(0,0)
881			
882	A1870	LA	LS,4(0,LS)
883		LH	MOSS,WT(PIECE,14)
884		AH	MOSS,WT2(FLAG,14)
885		ST	TOM,LISTMV(LS,14)
886		STC	FOM,LISTMV+2(LS,14)
887		STC	FLAG,LISTMV+1(LS,14)
888		STC	MOSS,LISTMV(LS,14)
889		SH	FLAG,N2(14)
890		CH	FLAG,N6(14)
891		BNL	A1870
892		B	A1880
893			
894	A1632	CH	PIECE,N6(14)
895		BNH	A1633
896			
897		LA	M,14(0,0)
898		LA	DELTA,4(0,0)
899		CH	PIECE,N10(14)
900		BE	A1650
901		BL	A1660
902		LA	DELTA,2(0,0)
903			

904	A1650	LA	M,16(0,0)
905	A1660	LR	TOM,FOM
906	A1670	AH	TOM,D(M,14)
907		CH	0,B(TOM,14)
908		BNE	A1680
909		SH	LB,N4(14)
910		ST	TOM,LISTMV(LB,14)
911		STC	FOM,LISTMV+2(LB,14)
912		B	A1670
913			
914	A1680	LH	PIECE,B(TOM,14)
915		CH	WHO,D1(PIECE,14)
916		BNE	A1690
917		LA	LS,4(0,LS)
918		LH	MASS,WT(PIECE,14)
919		ST	TOM,LISTMV(LS,14)
920		STC	FOM,LISTMV+2(LS,14)
921		STC	MASS,LISTMV(LS,14)
922			
923	A1690	SR	M,DELTA
924		BP	A1660
925		B	A1880
926			
927	A1633	BL	A1634
928			
929		LA	M,16(0,0)
930	A1710	LR	TOM,FOM
931		AH	TOM,D+16(M,14)
932		CH	0,B(TOM,14)
933		BNE	A1720
934		SH	LB,N4(14)
935		ST	TOM,LISTMV(LB,14)
936		STC	FOM,LISTMV+2(LB,14)
937		B	A1730
938			
939	A1720	LH	PIECE,B(TOM,14)
940		CH	WHO,D1(PIECE,14)
941		BNE	A1730
942		LA	LS,4(0,LS)
943		LH	MASS,WT(PIECE,14)
944		ST	TOM,LISTMV(LS,14)
945		STC	FOM,LISTMV+2(LS,14)
946		STC	MASS,LISTMV(LS,14)
947			
948	A1730	SH	M,N2(14)
949		BP	A1710
950		B	A1880
951			
952	A1634	LA	M,16(0,0)
953		LA	FLAG,16(0,0)
954	A1750	LR	TOM,FOM
955		AH	TOM,D(M,14)
956		CH	0,B(TOM,14)
957		BNE	A1760
958		SH	LB,N4(14)
959		ST	TOM,LISTMV(LB,14)

960		STC	FOM,LISTMV+2(LB,14)
961		STC	FLAG,LISTMV+1(LB,14)
962		B	A1770
963			
964	A1760	LH	PIECE,B(TOM,14)
965		CH	WHO,D1(PIECE,14)
966		BNE	A1770
967		LA	LS,4(0,LS)
968		LH	MASS,WT(PIECE,14)
969		ST	TOM,LISTMV(LS,14)
970		STC	FOM,LISTMV+2(LS,14)
971		STC	FLAG,LISTMV+1(LS,14)
972		STC	MASS,LISTMV(LS,14)
973			
974	A1770	SH	M,N2(14)
975		BP	A1750
976		B	A1880
977	*MAKE		
978	A1950	IC	TOM,MOVE+3(PLY1,14)
979		IC	FOM,MOVE+2(PLY1,14)
980		IC	FLAG,MOVE+1(PLY1,14)
981		XR	2,2
982		IC	2,MOVE(PLY1,14)
983			
984		AH	2,MATER-2(PLY,14)
985		LCR	2,2
986		STH	2,MATER(PLY,14)
987			
988		LH	1,B(TOM,14)
989		STH	1,SAVE(PLY,14)
990		LH	1,B(FOM,14)
991		STH	1,B(TOM,14)
992		STH	0,B(FOM,14)
993			
994		LTR	FLAG,FLAG
995		BZR	13
996		CH	FLAG,N14(14)
997		BNL	A1980
998		CH	FLAG,N4(14)
999		BH	A1970
1000		BE	A1960
1001			
1002		LR	1,TOM
1003		AH	1,D5(WHO,14)
1004		STH	1,PASSE(14)
1005		BR	13
1006			
1007	A1960	LR	1,TOM
1008		AH	1,D5(WHO,14)
1009		STH	0,B(1,14)
1010		BR	13
1011			
1012	A1970	LR	1,FLAG
1013		LTR	WHO,WHO
1014		BL	A1971
1015		LCR	1,1

1016	A1971	STH	1,B(TOM,14)
1017		BR	13
1018			
1019	A1980	STH	TOM,KINGSQ(NWHO,14)
1020		BHR	13
1021		CR	TOM,FOM
1022		BL	A1990
1023		LH	1,B+2(TOM,14)
1024		STH	1,B-2(TOM,14)
1025		STH	0,B+2(TOM,14)
1026		BR	13
1027	A1990	LH	1,B-4(TOM,14)
1028		STH	1,B+2(TOM,14)
1029		STH	0,B-4(TOM,14)
1030		BR	13
1031			
1032	*UNMAKE		
1033	A2010	IC	TOM,MOVE+3(PLY1,14)
1034		IC	FOM,MOVE+2(PLY1,14)
1035		IC	FLAG,MOVE+1(PLY1,14)
1036			
1037		LH	1,B(TOM,14)
1038		STH	1,B(FOM,14)
1039		LH	1,SAVE(PLY,14)
1040		STH	1,B(TOM,14)
1041			
1042		LTR	FLAG,FLAG
1043		BZR	13
1044		CH	FLAG,N14(14)
1045		BNL	A2040
1046		CH	FLAG,N4(14)
1047		BH	A2030
1048		BE	A2020
1049			
1050		STH	0,PASSE(14)
1051		BR	13
1052			
1053	A2020	LR	1,TOM
1054		AH	1,D5(WHO,14)
1055		LH	2,PAWN(WHO,14)
1056		STH	2,B(1,14)
1057		BR	13
1058			
1059	A2030	LH	1,PAWN(NWHO,14)
1060		STH	1,B(FOM,14)
1061		BR	13
1062			
1063	A2040	STH	FOM,KINGSQ(NWHO,14)
1064		BHR	13
1065		CR	TOM,FOM
1066		BL	A2050
1067		LH	1,B-2(TOM,14)
1068		STH	1,B+2(TOM,14)
1069		STH	0,B-2(TOM,14)
1070		BR	13
1071	A2050	LH	1,B+2(TOM,14)

1072		STH	1,B-4(TOM,14)
1073		STH	0,B+2(TOM,14)
1074		BR	13
1075			
1076	*CHECK		
1077	A2070	LA	1,16(0,0)
1078	A2080	LR	2,S
1079		AH	2,D(1,14)
1080		CH	0,B(2,14)
1081		BE	A2090
1082		LH	3,CHK4(1,14)
1083		AH	3,B(2,14)
1084		CH	C,CHK1(3,14)
1085		BER	13
1086		B	A2100
1087			
1088	A2090	AH	2,D(1,14)
1089		CH	0,B(2,14)
1090		BE	A2090
1091		LH	3,CHK4(1,14)
1092		AH	3,B(2,14)
1093		CH	C,CHK2(3,14)
1094		BER	13
1095	A2100	SH	1,N2(14)
1096		BP	A2080
1097			
1098		LH	3,NITE(C,14)
1099		LA	1,16(0,0)
1100	A2110	LR	2,S
1101		AH	2,D+16(1,14)
1102		CH	3,B(2,14)
1103		BER	13
1104		SH	1,N2(14)
1105		BP	A2110
1106			
1107		XR	C,C
1108		BR	13
1109			
1110		END	

Appendix 2: Test Problems and Solutions

The 300 problems from Reinfeld (1958) along with their accepted solutions are listed on the next 11 pages. Two lines are used for each problem. The first line is the pseudo-forsythe notation for the problem position. This is input to TECH3. The board position can be reconstructed by tracing from the top left corner (a8) of the board, as one would read, to the bottom right corner (h1) placing pieces as indicated. Lower case letters are black pieces while upper case letters are white pieces. Numbers indicate the number of empty squares between pieces or the edge of the board. A plus at the end means that white is to move in the position while a minus means that black is to move. As examples, Win at Chess #9 is the position shown in Figure 1, while Figure 2 is given by

83k44pPp13pP1P12pP42P53K48+

The second line, indented by 3 spaces, gives the problem number and the problem solution(s). The problem solution is the move to be played which gives the best result. Sometimes more than one best move is available. The corrections of Table I have been included.

2rr3kpp3pp11nnqbN1p3pN32pP42P3Q1PPB4PR4RK1+
 Win at Chess #1 MV=Qg3-g6
 87p5k25p2p1p2P2Pr1pPK21P1R3P8-
 Win at Chess #2 MV=Rb3xb2
 5rk11ppb3pp1pb46q13P1p1r2P1R2PPP1BQ1P15RKN+
 Win at Chess #3 MV=Re3-g3
 r1bq2rkpp3pbp2p1p1pQ7P3P42PB1N2PP3PPR2KR4+
 Win at Chess #4 MV=Qh6xh7
 5k26pppp1qN41p1p43P42PKP2QPP3r23R4-
 Win at Chess #5 MV=Qc6-c4
 7kp71R5K6r16p16P188+
 Win at Chess #6 MV=Rb6-b7
 rnbqkb1rpppp1ppp84P36n17PPPPNPPP1R1BQKBNR-
 Win at Chess #7 MV=Ng4-e3
 r4q1kp2bR1rp2p2Q1N5p25p22P5PP3PPPR5K1+
 Win at Chess #8 MV=Re7-f7
 3q1rk1p4pp12pb3p3p46Pr1PNQ4P1PB1PP14RRK1-
 Win at Chess #9 MV=Bd6-h2
 2br2k12q3rnp2NppQ12p1P3Pp5R4P31P3PPP3R2K1+
 Win at Chess #10 MV=Rh4xh7
 r1b1kb1r3q1ppppBp1pn28Np3P25B2PPP3PPR2Q1RK1+
 Win at Chess #11 MV=Bf3xc6
 4k1r12p3r11pR1p33pP2p3P2qPP4N21PQ4P5R1K-
 Win at Chess #12 MV=Qg4xf3
 5rk1pp4p12n1p2p2Npq32p56P1P3P1BPR4Q1K+
 Win at Chess #13 MV=Qf1xf8
 r2rb1k1pp1q1p1p2n1p1p12bp45P2PP1BPR1Q1BPN2PPR5K1+
 Win at Chess #14 MV=Qh3xh7
 1R61brk2p14p2pp1P1Pp2P76P11P4P12R3K1+
 Win at Chess #15 MV=Rb8xb7
 r4rk1ppp2ppp2n52bqp38P2PB31PP1NPPPR2Q1RK1+
 Win at Chess #16 MV=Ne2-c3
 1k5rpppbN1pp4q1r11P3p22Npp31QP5P4PPPR1B1R1K1+
 Win at Chess #17 MV=Nc4-e5
 R7P4k28888r76K1+
 Win at Chess #18 MV=Ra8-h8
 r1b2rk1ppbn1ppp4p31QP4q3P4N4N25PPPR1B2RK1+
 Win at Chess #19 MV= c5-c6
 r2qkb1r1ppb1pppp74p3P1Q1P32P55PPPR1B2KNR-
 Win at Chess #20 MV=Bd7-b5
 5rk11b3p1ppp3p23n1N21P6P1qB1PP13Q3P4R1K1+
 Win at Chess #21 MV=Qd2-h6
 r1bqk2rppp1nppp4p3n5N12BPp3P1P52P2PPPR1BQK2R+
 Win at Chess #22 MV=Ng5xf7
 r3nrk12p2p1pp1p1b1p12NpPq23R4P1N1Q31PP2PPP4R1K1+
 Win at Chess #23 MV= g2-g4
 6k11b1nqpbppp4p15P21PN54Q3P5PP1B2B1K1-
 Win at Chess #24 MV=Bg7-d4
 3R1rk185Qpp2p52P1p1q1P3P31P2PK28-
 Win at Chess #25 MV=Qg4-h4
 3r2k11p1b1pp1pq5p83NR32PQ3PPP3PP16K1-
 Win at Chess #26 MV=Bd7-f5
 7kpp4np2p3p13pN1q13P4Q71r3rPP2R2RK1+
 Win at Chess #27 MV=Qa3-f8
 1r1r2k14pp1p2p1b1p1p3R3RqBP44P31PQ2PPP6K1-
 Win at Chess #28 MV=Qb4-e1

r2q2k1pp1rbppp4pn22P51P3B26P1P3QBP1R3RK1+
 Win at Chess #29 MV= c5-c6
 1r3r24q1kpb1pp2p15p2pPn1N36P1P3PPBP2QRR1K1+
 Win at Chess #30 MV=Ne4xd6
 rb3qk1pQ3ppp4p33P481P3N21P3PPP3R2K1+
 Win at Chess #31 MV= d5-d6 or MV=Qb7xa8
 6k1p4p1p1p3np12q54p34P1N1PP3PPP3Q2K1+
 Win at Chess #32 MV=Qd1-d8
 8p1q2pkp2Pr2p18P3Q36P15P1P2R3K1+
 Win at Chess #33 MV=Qe4-e5 or MV=Qe4-f4
 7k1b1r2p1p6p1p2qN23bP33Q4P5PP1B1R3K-
 Win at Chess #34 MV=Bd4-g1
 r3r2k2R3pppp1q1p283P3R7PPP3PP13Q2K1+
 Win at Chess #35 MV=Rh4xh7
 3r42p1rk21pQq1pp17p1P1P4P4P26PPR1R3K1-
 Win at Chess #36 MV=Re7-e1
 2r52rk2pp1pn1pb2pN1p4P2P41N2B3nPR1KPPP3R4-
 Win at Chess #37 MV=Nc6xd4
 4k3p4prp1p62b582Q3P1P2R1PKP4q3+
 Win at Chess #38 MV=Rd2-d8
 r1br2k1pp2bppp2nppn282P1PB22N2P2PqN1B1PPR2Q1R1K+
 Win at Chess #39 MV=Nc3-a4
 3r1r1k1p4ppp4p281PQR46PqP3PP22R3K1-
 Win at Chess #40 MV=Rd8-c8
 1k65RP11P61K66r1888+
 Win at Chess #41 MV=Kb5-a5
 r1b1r1k1pp1n1pbp1qp3p13p41B1P4Q3PN2PP2BPPPR4RK1+
 Win at Chess #42 MV=Bb4-a5
 r2q3kp2P3p1p3p23QP1r18B7P5PP2R3K1+
 Win at Chess #43 MV=Qd5xa8 or MV=Ba3-e7
 3rb1k1pq3pbp4n1p13p42N52P2QB1PP3PPP1B1R2K1-
 Win at Chess #44 MV= d5xc4
 7k2p1b1pp81p2P31P3r22P3Q11P5PR4qBK-
 Win at Chess #45 MV=Qf1xa1
 r1bqr1k1pp1nb1p14p2p3p1p23P4P1N1PNP11PQ2PP13RKB1R+
 Win at Chess #46 MV=Nc3-b5
 r1b2rk1pp2bppp2n1pn2q5B12BP42N2N2PP2QPPP2R2RK1-
 Win at Chess #47 MV=Nc6xd4
 1rbq1rk1p1p1bppp2p2n28Q1BP42N5PP3PPPR1B2RK1-
 Win at Chess #48 MV=Rb8-b4
 2b3k14rrppp2p42pP2RQ1pP1Pp1N1P3P1P1q66RK+
 Win at Chess #49 MV=Qh5xh7
 k4r21R4pb1pQp1n1p3P45p1P3P2P1r1q1R2K8+
 Win at Chess #50 MV=Rb7xb6
 r1bq1r2pp4k14p2p3pPp1Q3N1R1P2PB46P16K1+
 Win at Chess #51 MV=Rf4-g4
 r1k51p3q21Qpb43N1p25Pp13P2PpPPPK3P4R3+
 Win at Chess #52 MV=Re1-e7
 6k16p1p73Pn35p24rBqPP4RP15QK1-
 Win at Chess #53 MV=Re3-e1
 r3kr21pp4p1p1p47q4P1n12PP2Q1PP4P1R1BB2K1-
 Win at Chess #54 MV=Qh5-h1
 r3r1k1pp1q1pp14b1p13p2B13Q1R28PPP3PP4R1K1+
 Win at Chess #55 MV=Qd4xg7
 r1bqk2rpppp1ppp5n22b1n34P31BP3Q1PP3PPPRNB1K1NR-
 Win at Chess #56 MV=Bc5xf2

r3q1krppp53p2pQ83PP1b15R2PPP3P15RK1+
 Win at Chess #57 MV=Rf3-f8
 882R51p2qp1k1P2r32PQ2P15K28+
 Win at Chess #58 MV=Qd3-d1
 r1b2rk12p1qnbpp1pp2p15p22PQP31PN2N1PPB3PP13R1RK1+
 Win at Chess #59 MV=Nc3-d5
 rn1qr1k11p2np22p3p181pPb47QPB1P1PP12KR1B1R+
 Win at Chess #60 MV=Qh3-h8
 3qrbk1ppp1r2n3pP2p3P42P4P1P3Q2PB6R4R1K+
 Win at Chess #61 MV=Qf3-f7
 6r13Pn1qkp1p1P1rp2Q2p22P51P4P1P3R2P5RK1-
 Win at Chess #62 MV=Rg6xg3
 r1brnbk1ppq2pp14p2p4N33P4P1PB1Q23B1PPPR3R1K1+
 Win at Chess #63 MV=Ne5xf7
 86pp3q1p23n1k21P63NQ2P5PP16K1+
 Win at Chess #64 MV= g2-g4
 1r1r1qk1p2n1p1pbbp1Pn1pQ2pNp32P2P1N1P5BP6P3R1RK1+
 Win at Chess #65 MV=Nd5-e7
 1k1r2r1ppq51bp4p3pQ382P2N2PP4P1R4R1K-
 Win at Chess #66 MV=Qc7xe5
 3r2k1p2q41p4p13rRp1p5P1P6PKP3R33Q4+
 Win at Chess #67 MV=Re5xd5
 6k15ppp1q62b582R1pPP11P2Q2P7K+
 Win at Chess #68 MV=Qe2xe3
 2k5pppr44R34Q32pp2q18PPP2PPP6K1+
 Win at Chess #69 MV= f2-f3
 2kr3rpppg1ppp3p1n2bQ2p31n1PP31PN1BN1P1PP2PP12KR3R-
 Win at Chess #70 MV=Nb4-a2
 2kr3rpp1q1ppp5n21Nb52Pp1B27QP4PPP1R3RK1+
 Win at Chess #71 MV=Nb5xa7
 r3r1k1pp1n1ppp2p54Pb22B2P2B1P5P5PPR2R2K1+
 Win at Chess #72 MV= e5-e6
 r1q3rk1ppbb1p14Np1pp3pP2P3P32N4R1PP1Q1PP3R2K1+
 Win at Chess #73 MV=Qe2-d2
 5r1kpp4pp2p52b1P34Pq21PB1p3P3Q1PP3N2K1-
 Win at Chess #74 MV=Qf4-f1
 r3r1k1pppq1ppp881Q4n17PPP2PP1RNB1R1K1-
 Win at Chess #75 MV=Qd7-d6
 r1b1qrk12p2ppppb1pnn21p2pNB13PP31BP5PP2QPPPRN1R2K1+
 Win at Chess #76 MV=Bg5xf6
 3r2k1ppp2ppp6q1b4n23nQB22p5P4PPPRN3RK1-
 Win at Chess #77 MV=Nf5-g3
 r2q3rppp2k24nbp15Q1p2P1NB28PP3P1P3RR1K1+
 Win at Chess #78 MV=Ne4-g5
 r3k2rpbp2pp13b1n21p63P3p1B2N1PqPP1PQP1PR1B2RK1-
 Win at Chess #79 MV=Qh3xh2
 r4rk1p1B1bpp11p2pn1p82PP43B1P2qP2QP1P3R1RK1+
 Win at Chess #80 MV=Rd1-a1
 r4rk11bR1bppp4pn21p2N31P6P3P34BPPP3R2K1-
 Win at Chess #81 MV=Be7-d6
 3rr1k1pp3pp14b382P1B2R6QPP3q1P15R1K+
 Win at Chess #82 MV=Be4-h7
 3rr1k1ppqbRppp2p583Q1n22P3N1PPB2PPP3R2K1+
 Win at Chess #83 MV=Qd4xd7
 r2q1r1k2p1b1ppp1n51p1Q1bN14n31BP1B3PP3PPPR4RK1+
 Win at Chess #84 MV=Qd5-g8

kr2R3p4r22pq42N2p1p3P2p1Q5P15P1P5BK1+
 Win at Chess #85 MV=Nc5-a6
 8p71ppk1n25pppP1PP42P1K1P15N1P8-
 Win at Chess #86 MV=Nf6-g4
 3p3k1p14r32ppNpp1PP1P42P3KP5P28-
 Win at Chess #87 MV=Re6xe5
 r6kp1Q4p2p1b1rq4p3B3P34P3PPP3P14RRK1-
 Win at Chess #88 MV=Rg6xg2
 1r3b1kp4rpp4pp23q42ppbPPQ6RKPP5P2B1NR2-
 Win at Chess #89 MV= g7-g5
 3qrrk11pp2pp11p2bn1p5N22P5P1P3B11P4PP2Q1RRK1+
 Win at Chess #90 MV=Nf5xg7
 2qr2k14b1p12p2p1p1pP1p3p2nP3PbQNB1PP1P3PK14RB2-
 Win at Chess #91 MV=Bb3-e6
 r4rk11p2ppbpp2pbnp1q73BPPP12N2B2PPP4PR2Q1RK1-
 Win at Chess #92 MV=Be6xg4
 r1b1k1nrpp3pQp4pq23pn38P1P52P2PPPR1B1KBNR+
 Win at Chess #93 MV=Bc1-h6
 8k7p73Qp2Pn1P53KP31q68-
 Win at Chess #94 MV= e5-e4
 2r51r64pNpk3pP1qp82P1QP25PK1R7+
 Win at Chess #95 MV=Nf6-g4
 r1b4kppp2Bb16Pp3pP31qnP1p1Q8PPP3P11K1R3R+
 Win at Chess #96 MV=Qh4-d8
 6k15p2p5np4B33P41PP1q3P3r1QP6RK+
 Win at Chess #97 MV=Qg2-a8
 1r3rk15pb1p2p2p1Q1n1q2p1NP1P33p1P1BPP1R3P1K2R3-
 Win at Chess #98 MV=Nc5xe4 or MV=Nc5-b3
 r1bq1r1k1pp1Np1pp2p2pQ4R3n78PPPP1PPPR1B3K1+
 Win at Chess #99 MV=Re5-h5
 8k1b5P4p21Pp2p1pK1P2P1P83B48+
 Win at Chess #100 MV= b5-b6
 5rk1p5pp882Pbp31P4P17P4RN1K-
 Win at Chess #101 MV=Bd4-c3
 2Q2n22R4p1p1qpp1k83P3P3B2P15PK1r7+
 Win at Chess #102 MV=Qc8xf8
 6k12pb1r1p3p1PpQp1nPp31q2P32N2P2PrB52K3RR+
 Win at Chess #103 MV=Qh6xg6
 b4r1kpgq2rp21p1bpn1p3PN2n2P2P2P2B3K1B2Q2N3R2R1+
 Win at Chess #104 MV=Qe2xh5
 r2r2k1pb3ppp1p1bp37q3n2nPPP1B2P11B1N1P2RQ2NRK1-
 Win at Chess #105 MV=Qh5xh4
 4rrk1pppb47p3P2pq3Qn3P5P11PP4PR3RNNK-
 Win at Chess #106 MV=Ne4-f2
 5n2pRrk2p1P4p1p4p33N45P26PP6K1+
 Win at Chess #107 MV=Nd4-b5
 r5k11q4pp2p5p1Q52P55R24RKPPr7+
 Win at Chess #108 MV=Qc5-e5
 rn2k1nrpbp2ppp3q41p2N32p5QP6PB1PPPPPR3KB1R-
 Win at Chess #109 MV= c4-c3
 2kr4bp3p2p2p2b1P72q51N4B11PPQ2P12KR4-
 Win at Chess #110 MV=Ba7-e3
 6k1p5p15p22P2Q23pN2p3PbK1P7P6q1-
 Win at Chess #111 MV=Qg1-f1
 r4kr1ppp54bq1b7B2PR1Q1p2N3P1PP3P1P2K1R3+
 Win at Chess #112 MV=Re1xe6

rnbqkb1r1p3ppp5N21p2p1B12P58PP2PPPPR2QKB1R-
 Win at Chess #113 MV=Qd8xf6
 r1b1rnk11p4ppp1p2p23pN2n3P1PPq2NBPR1PPPQ52R3K1+
 Win at Chess #114 MV=Bd3xh7
 4N2k5rpp1Q6p3q38P5P11P3P1P5K2+
 Win at Chess #115 MV=Ne8-d6
 r2r2k12p2pppp71p2P1n1P6q5P21PB1QP1PR5RK-
 Win at Chess #116 MV=Rd8-d2
 3r1rk1q4pppp1Rnp381p61N3P2PP3QPP3R2K1-
 Win at Chess #117 MV=Nd6-e4
 r5k1pb2rpp11p62p4q5R22PB2Q1P1P3PP5R1K+
 Win at Chess #118 MV=Rf4-h4
 r2qr1k1p1p2ppp2p52b54nPQ13B4PPP3PPR1B2R1K-
 Win at Chess #119 MV=Qd8xd3
 r4rk11bn2qnp3p1B1Qp2P1pP11pp55N1PPPB2P22KR3R+
 Win at Chess #120 MV= g5-g6
 6k15p1p2bP2pb4p32P51p1pNPPP1P1Q1BK11q6-
 Win at Chess #121 MV=Bc6xf3
 1k6ppp4p1n2pq21N2Rb22P2Q28P4KPP3r1B2-
 Win at Chess #122 MV=Rd1xf1
 6k11b2rp21p4p13P4PQ4P12N2q25P23R2K1-
 Win at Chess #123 MV=Bb7xd5 or MV=Re7-c7
 6k13r42R5P5P11P4p184rB26K1-
 Win at Chess #124 MV= g4-g3
 r1bqr1k1pp3ppp1bp53n43B42N2P1PPPP1B1P1R2Q1RK1-
 Win at Chess #125 MV=Bb6xd4
 r5r1pQ5p1qp2R22k1p34P32PP4P1P3PP6K1+
 Win at Chess #126 MV=Rf6xc6
 2k4r1pr1n3p1p1q2p5pp13P1P2P1P1P31R2Q1PP1RB3K1+
 Win at Chess #127 MV=Rb2xb7
 6rk1pp2Qrp3p1B21pb1p2R3n1q23P4PPP3PPR6K+
 Win at Chess #128 MV=Qf7-g6
 3r1r1k1b2b1p11p5p2p1Pp2q1B2P24P2P1BR1Q2K6R1-
 Win at Chess #129 MV=Bb7-f3
 6k11pp3q15r21PPp43P1pP13Qn2P3B44R1K1-
 Win at Chess #130 MV=Qg7-h6 or MV=Qg7-h8
 2rq1bk1p4p1p1p4p13b43B1Q28P4PpP3RR1K1+
 Win at Chess #131 MV=Re1-e8
 4r1k15bpp2p53pr381B3pPqPPR2P22R2QK1-
 Win at Chess #132 MV=Re5-e1
 r1b1k2r1pp1q2pp1n3p13QpP281BP3B1P5PP3R1RK1+
 Win at Chess #133 MV=Bg3-h4
 3r2k1p6p2Q3p14q32P1p3P3Pb21P3P1P2K2BR1-
 Win at Chess #134 MV=Rd8-d1
 3r1r1kN2qn1pp1p2np22p52Q1P2N3P4PP4PP3R1RK1-
 Win at Chess #135 MV=Ne6-d4
 6kr1q2r1p11p2N1Q15p21P1p46R17P2R3K1+
 Win at Chess #136 MV=Rc1-c8
 3b1rk11bq3pp5pn11p2rN22p1p32P1B2Q1PB2PPPR2R2K1+
 Win at Chess #137 MV=Rd1-d7
 r1bq3rppppR1p15n1k3P46pP3Q4PP1N1PP15K1R+
 Win at Chess #138 MV= h4-h5
 rnb3krppp2ppp1b63q43pN3Q4N2PPP2KPPR1B1R3+
 Win at Chess #139 MV=Ne4-f6
 r2b1rk1pq4p14ppQP3pB1p13P42R5PP3PP15RK1+
 Win at Chess #140 MV=Rc3-c7

4r1k1p1qr1p22pb1Bp11p5p3P1n1R1B3P2PP3PK12Q4R+
 Win at Chess #141 MV=Qc1xf4
 r2q3nppp2pk13p45Pr12NP1Qp12P2pP1PP3K24R2R+
 Win at Chess #142 MV=Re1-e8
 5b2pp2r1pk2pp1pRp4rP1N2P1P31P4QPP3q1P15R1K+
 Win at Chess #143 MV=Rg6xh6
 r2q1rk1pp3ppp2p2b28B2pPPb17PPP1N1P1R2Q1RK1-
 Win at Chess #144 MV= d4-d3
 r1bq41p4kp3p1n2p4pB12pQ481P4PP4RRK1+
 Win at Chess #145 MV=Re1-e8
 882Kp43P1B22P2k25p288+
 Win at Chess #146 MV=Bf5-c8
 r2r2k1ppqbpbbp2n2np12pp46P11P1PPNNPPBP2PB1R2QK2R-
 Win at Chess #147 MV=Nf6xg4
 2r1k36prp1nBP31p3p1p2q52P5P1R4PK2Q2R1+
 Win at Chess #148 MV=Rg1xg7
 6k16p12p4p4Pp24b1qP2Br41P2RQPK8-
 Win at Chess #149 MV=Be4xg2
 r3r1k15p2pQ1b2pB1p64p36P1Pq2BP1P2R3K1-
 Win at Chess #150 MV=Bd6-f8
 83b2kp4p1p1pr1n4N1N4P1P4P11K3P23R4+
 Win at Chess #151 MV=Na4-c3
 1br2rk11pqb1pppp3pn281P6P1N1PN1P1B3PP11QRR2K1+
 Win at Chess #152 MV=Nc3-d5 or MV=Nc3-e4
 2r3k1q4pppp3p3pnNp42rP42P2P24R1PP2R1Q1K1-
 Win at Chess #153 MV=Nb5xd4
 r1b2rk12p2pppp71p63P3q1BP3bPPP3QP1RNB1R1K1+
 Win at Chess #154 MV=Qf2xf7
 5bk11rQ4p5pp12pP43n1PP17P1q3BB14R1K1+
 Win at Chess #155 MV= d5-d6
 r1b1qN1k1pp3p1p2p3n4p1B181BP4QPP3KPP8+
 Win at Chess #156 MV=Qh3xh6
 5rk1p4ppp2p1b33Nq34P1n11p1B2QP1PPr2P11K2R2R+
 Win at Chess #157 MV=Nd5-e7
 5rk1n1p1R1bpp2p41qpP1QB17P2P3P1PP3P26K1+
 Win at Chess #158 MV=Re7xg7
 r1b2r25P1pppn3pk2p1p1Nq1bP1PQ23P4PB4BP1R3RK1+
 Win at Chess #159 MV=Ng5-e6
 qn1kr2r1pRbb3pP5pP2pP1pP3N1pQ13B43B1PP1R5K1+
 Win at Chess #160 MV=Qg4xd7
 3r3k3r1P1ppp1Nn32pp47Q6R1Pq4PP5RK1+
 Win at Chess #161 MV=Qh4xd8
 r3kbnrp4ppp2p1p38Q1B3b12N1B3PP3PqPR3K2R+
 Win at Chess #162 MV=Bc4-d5
 5rk12p4p2p4r3P44p1b11Q2NqPpPP3P1KR4R2-
 Win at Chess #163 MV=Qf3-g2
 86pp4p31p1n41NbKN1P1P4P1P1PR3K1r7+
 Win at Chess #164 MV=Rc2xc4
 1r5kp1p3pp884p3P1P1R31P1Q1qr12KR4+
 Win at Chess #165 MV=Re3-e2
 r3r1k15pp1p1p4p2Pp48q1NQP1BP5PP14K2R-
 Win at Chess #166 MV= d5-d4
 7Qppp2q23p2k1P2Ppr1N1PP57R5rP16K1-
 Win at Chess #167 MV=Rf2xg2
 r3k2rpb1q1p282p1pP24p1p1B1P1Q1P1P1P3K1R4R2-
 Win at Chess #168 MV=Qd7-d2

5rk11pp3bp3p2p12PPp31P2P32Q1B34q1PPR5K1-
 Win at Chess #169 MV=Bg7-h6
 5r1k6Rp1p2p3p2pBp21qnP44P3Q4PPP6K1+
 Win at Chess #170 MV=Qa2xc4
 2rq41b2b1kpp3p1p11p1nNp27P1B2B1Q1PP3PP13R2K1+
 Win at Chess #171 MV=Be3-h6
 5r1kp5pp81P1pq3P1p2nR1Q75BPP6K1-
 Win at Chess #172 MV=Qe5-e1
 2r1b31pp1qrk1p1n1P1p17R2B1p34Q1P1PP3PP13R2K1+
 Win at Chess #173 MV=Qe3-h6
 2r2rk16p1p3pq1p1p1b1p23P1n2PP3N23N1PPP1Q2RR1K-
 Win at Chess #174 MV=Nf4xg2
 r5k1pppb3p2np1n283PqNpP3Q2P1PPP5R4RK1+
 Win at Chess #175 MV=Nf4-h5
 r1bq3rppp2pk13p1pp182BbPQ22NP2P1PPP4PR4R1K-
 Win at Chess #176 MV=Rh8xh2
 r1b3r14qk21nn1p1p13pPp1Pp4P21p3BQNPKPBN33R3R-
 Win at Chess #177 MV=Qe7-a3
 3r2k1p1rn1p1p1p2pp26q13PQNP15P2P1P4RR5K1+
 Win at Chess #178 MV=Nf4xe6
 r1b2r1kpp4pp3p43B481QN3PnPP3q1PR3R2K-
 Win at Chess #179 MV=Qf2-g1
 r1q2rk1p3bppb3p1n1p2nPP31p2P1P16NPPP2QPB1R1BNK2R-
 Win at Chess #180 MV=Nf6xd5
 r3k2r2p2p2p2p1n21p2p34P2p1PPPPp1q1P5PR1N2QRK-
 Win at Chess #181 MV=Nf6-g4
 r1b2rk1ppgn1p1p2n1p1p12b3N12N5PP1BP31B3PPPR2QK2R+
 Win at Chess #182 MV=Qd1-h5
 1r2k1r15p2b3p31p2b1B13p3P3B4PP2KP22R3R1+
 Win at Chess #183 MV=Bg5-f6
 4kn2r4p1rp3bQ2q1nNP1Np1p5P8PPP3P12KR3R+
 Win at Chess #184 MV=Qf6-e7
 1r1rb1k12p3ppp2q1p23PpP1QPp1bP2N1B5R1P4PP2B4K+
 Win at Chess #185 MV=Qh5xh7
 r5r1p1q2p1k1p1R2pB3pP36bQ2p5P1P1NPPP6K1+
 Win at Chess #186 MV=Bh6-f8
 6k15p2p3p31p3qp12p1Qn22P1R3PP1r1PPP4R1K1-
 Win at Chess #187 MV=Nf4-h3
 3RNbk1pp3p24rQpp81qr57PP4P23R2K1+
 Win at Chess #188 MV=Qf6-g7
 3r1k21ppPR1n1p2p1rP13P3p4Rp1N5K2P1P2P28+
 Win at Chess #189 MV=Re7-e8
 8p2b2kp1q1p2p11P1Pp34P33B2P1P2Q3P2Nn3K-
 Win at Chess #190 MV=Bd7-h3
 2r1Rn1k1p1q2ppp75p23P41B4P1P1P1QP1P6K1+
 Win at Chess #191 MV=Qe2-c4
 r3k3ppp2Npp4Bn22b51n1pp3N4P2PPP3qPR2QKR2-
 Win at Chess #192 MV=Nb4-d3
 5bk1p4pppQp64B31P6Pq2P1P12rr1P1PR4RK1-
 Win at Chess #193 MV=Qb3xe3
 5rk1ppq2ppp2p54bN24P36Q1PPP2PPP3R2K1+
 Win at Chess #194 MV=Nf5-h6
 3r1rk11p3p2p3pnnp2p3p12P2q21P5PPB2QPPN3RR1K1+
 Win at Chess #195 MV= g2-g3
 rr4k1p1pq2ppQ1n1pn22bpp34P32PP1NN1PP3PPPR1B1K2R-
 Win at Chess #196 MV=Nc6-b4

7k1p4p17p3P1n24Q32P2P2PP3qRP7K-
 Win at Chess #197 MV=Qf2-f1
 2br2k1ppp2p1p4p1p14P2q2P1Bn22Q5PP3P1P4R1RK-
 Win at Chess #198 MV=Rd8-d3
 r1br2k1pp2nppp2n51B1q4Q74BN2PP3PPP2R2RK1+
 Win at Chess #199 MV=Rf1-d1 or MV=Bb5xc6
 2rqrn1kpb4pp1p2pp2n2P42P3N1P2B2Q11B3PPP2R1R1K1+
 Win at Chess #200 MV=Bb2xf6
 2b2r1k4q2p3p2pQ2pBp386P11PP2P1PR5K1+
 Win at Chess #201 MV=Ra1-a7
 QR2rq1k2p3p13p1pPp84P38P1r3PP1R4K1-
 Win at Chess #202 MV=Rc2xa2
 r4rk15pppp3q1n12p2NQ14n3P3P31B3PPP1R3RK1+
 Win at Chess #203 MV=Qg5-h6
 r1b1qrk11p3pppp1p53Nb35N2P71P4PQK1R1R3+
 Win at Chess #204 MV=Re1xe5
 r3rnk11pq2bb1p4p23p1Pp13B2P11NP4RP1QOB32K4R+
 Win at Chess #205 MV=Qd2xg5
 1Qq52P1p1kp3r1pp1887Pp4PP12R3K1-
 Win at Chess #206 MV=Rd6-c6
 r1bq2krp1pp1ppp1pn1p34P32Pb2Q1BR6P4PPP3K1BNR+
 Win at Chess #207 MV=Qg4xg7
 3r1bk1ppq3pp2p52P2Q1B81P4P1P6P5RK1+
 Win at Chess #208 MV=Bh5-f7
 4kb1r2q2p2r2p4pppBn1B1P6P6Q11PP52KRR3+
 Win at Chess #209 MV=Re1xe5
 3r1rk1pp1q1ppp3pn32pN45PP1P5PQ1PP1B31K1R4+
 Win at Chess #210 MV=Rd1-h1
 r1bqrk2pp1n1n1p3p1p2P1pP1P1Q2PpP1NP6R12PB44RBK1+
 Win at Chess #211 MV=Qh5xf7
 rn1qr2Qpbppk1p11p2pb24N33P42N5PPP3PPR4RK1+
 Win at Chess #212 MV=Qh8xg7
 3r1r1k1b4ppppn1p34Pp1RPn5P3P44QP21qB1NKR1+
 Win at Chess #213 MV=Rh5xh7
 r2r2k11p2qpp11np1p1p1p3N32PPN3bP5R4QPPP4R1K1+
 Win at Chess #214 MV=Ne4-g5
 3r2k1pb1q1pp11p2pb1p83N4P2QB31P3PPP1Br1R1K1+
 Win at Chess #215 MV=Qd3-h7
 r2qr1k11b1nbpppp3pn21p1pN33P1B22PB1N2PP2QPPPR4RK1+
 Win at Chess #216 MV=Ne5xf7
 r3kb1r1pp3p1p3bp1p5q23QN31P6PBP3P13RR1K1+
 Win at Chess #217 MV=Qd4-d7
 6k1pp5p2p3q16BP2nPr1Q18PP3R1K8+
 Win at Chess #218 MV=Bg5-h6
 7kp4q1p1pb52p54B2Q2P1B3P6P7K-
 Win at Chess #219 MV=Qf7-f1
 3rr1k1ppp2ppp85Q24n31B5RPPP1qPP15RK1-
 Win at Chess #220 MV=Qe2xf1
 r3k3P5bp2N1bp24p32p56NP1PP2PP13R2K1+
 Win at Chess #221 MV=Rd1-d8
 2r1r2k1q3pppp2Rp32p1P36QBp3P3bP3PPP3R2K1+
 Win at Chess #222 MV=Bh4-f6
 r1bqk2rpp3ppp5n281b1npB22N5PP1Q2PP1K2RBNR+
 Win at Chess #223 MV=Nc3xe4
 5rk1p1q3pp1p1r42p1pp1Q1PPn1P23B3PP2R2P13R2K1-
 Win at Chess #224 MV=Rd6-h6 or MV= e5-e4

4R34q1kp6p11Q3b21P1b1P26KP88-
 Win at Chess #225 MV=Qe7-h4
 2b2rk1p1p4p2p1p1p1br2N1Q11p2q38PB3PPP3R1RK1+
 Win at Chess #226 MV=Ne5-f7
 2k1rb1rppp3pp2np1q25b222P22P1BQ2PP1N1P1P2KR3R-
 Win at Chess #227 MV= d6-d5
 r4rk11bq1bp1p4p1p1p2p43BnP21N1B3RPPP3PPR2Q2K1+
 Win at Chess #228 MV=Bd3xe4
 8881p5rp1p1k1pNP2pBpP11P1K1P28-
 Win at Chess #229 MV= b5-b4
 2b51r62kBP1p1p2pP1P12pP41pP3K11R3P28-
 Win at Chess #230 MV=Rb7-b4
 r4rk11b1nqp1pp5p11p2PQ22p55N2PP3PPPR1BR2K1+
 Win at Chess #231 MV=Bc1-g5
 1R2rq1k2p3p1Q2p1pPp84P38P1r3PP1R4K1+
 Win at Chess #232 MV=Rb8xe8
 5rk1p1p2r1p2pp2p14p3PPpP33Pq1P11Q1R1R1P4NK2-
 Win at Chess #233 MV=Nd4-b3
 2kr1r2p6p5Pp12p51qp2Q1P7RPP61KR5+
 Win at Chess #234 MV=Rh3-b3
 5r21p1RRrk14Qq1p1PP3p184B31b3P1P6K1+
 Win at Chess #235 MV=Re7xf7 or MV=Qe6xf7
 1R6p5pk4p2p4P382r3qPP3R1b14Q1K1-
 Win at Chess #236 MV=Rc3-c1
 r5k1pQp2qpp84pbN13P46P1PPr4P1K1R3R-
 Win at Chess #237 MV=Rc2-c1
 1k1r4pp1r1pp14n1p12R52Pp1qP13P2QPP4PB11R4K1+
 Win at Chess #238 MV=Bg2xb7
 86k15pp1Q6p5P26PKP4q1P8-
 Win at Chess #239 MV=Qf2-f1
 2b4kp1b2p22p2q23p1PNp3P2R13B4P1Q2PKP4r3+
 Win at Chess #240 MV=Qc2xc6
 2rq1rk1pp3ppp2n2b24NR23P4PB5Q1P4PP3R2K1+
 Win at Chess #241 MV=Qh3xh7
 r1b1r1k1pp1nqp22p1p1pp84N3P1Q1P31P3PPP1BRR2K1+
 Win at Chess #242 MV=Rd1xd7
 1r3r1k3p41p1Nn1R14Pp1qpP3P1pP75Q1P6RK+
 Win at Chess #243 MV=Qf2-e2
 r6rpp3ppp3k1b22pb4B4Pq12P1Q3P5PP1RBR2K1+
 Win at Chess #244 MV=Qe3xc5
 4rrn1ppq3bk3pPnpp2p52PB42NQ1RPBPP5P5R1K+
 Win at Chess #245 MV=Qd3xg6
 6R14qp1pppr1n1pk81P2P1QP6N1P4PP16K1+
 Win at Chess #246 MV=Qg4-h5
 2k1r31p2Bq2p2Qp3Pb1p1p1P2pP1P22P52P2KP11R6+
 Win at Chess #247 MV=Rb1xb5
 5r1k1p4pp3q43Pp1R188PP4PP4Q1K1-
 Win at Chess #248 MV=Qd6-c5
 r4rk1pbq2pp11ppbnp1p82PP41P1Q1N2PBB2PPPR3R1K1+
 Win at Chess #249 MV= c4-c5
 1b5k7Pp1p2np12P2p2PP3P24RQ1Rq2r3P6K1+
 Win at Chess #250 MV=Re3-e8
 k7p4p2P1q1b1p13p3p3Q47P5PP11R4K1+
 Win at Chess #251 MV=Qd4-e5 or MV=Qd4-f4
 1rb1r1k1p1p2ppp5n22pP45P22QB4qNP3PP2KRB2R-
 Win at Chess #252 MV=Re8-e2

k5r1p4b22P55p23P1P24QBrqP5P14R1K1+
 Win at Chess #253 MV=Qe3-e8
 r6kpp3p1p2p1bp1qb3p34Pnr12PP2NPPP1Q1PPNR2B2RK-
 Win at Chess #254 MV=Nf4xh3
 3r3rp4pk15Rp13g41p1P2RQ5N2P1P4P2b4K+
 Win at Chess #255 MV=Rf6xg6
 3r1rk11pb1qp1p2p3p1p7P2Np2R1P5P1BP2PP13Q1BK1+
 Win at Chess #256 MV=Nd4-f5
 4r1k1pq3p1p2p1r1p12Q1p33nN1P11P6P1P2P1P3RR1K1+
 Win at Chess #257 MV=Rd1xd4
 r3brkn1p5p2p2Ppq2Pp3B3Pp2Q4P1R16PP5R1K+
 Win at Chess #258 MV=Bh5xg6
 r1bq1rk1ppp2ppp2np42bN1PN12B1P33p4PPP2nPPR1BQ1K1R+
 Win at Chess #259 MV=Qd1-h5
 2r2b1rp1Nk2pp3p1p2N2Qn34P3q6PP4PP11R3K1R+
 Win at Chess #260 MV=Qd5-e6
 r5k11bp3ppp2p41p65p21PBP1ngP1PP3Q1R4R1K-
 Win at Chess #261 MV=Nf3-d4
 6k1p1B1b2p2b3r12p54p31PP1N1PqP2R1P23Q2K1-
 Win at Chess #262 MV=Rg6-h6
 rnbqr2kpppp1Qpp8b2NN32B1n38PPPP1PPPR1B1K2R+
 Win at Chess #263 MV=Qf7-g8
 r2r2k11R2qp2p5pp2P5b1PN1b2P71Q3PPP1B1R2K1-
 Win at Chess #264 MV=Rd8xd4 or MV=Ra8-b8
 2r1k2r2pn1pp11p3n1pp3PP24q2BP1P52Q1N1PPR4RK1+
 Win at Chess #265 MV= e5xf6
 r3q2r2p1k1p1p5p11p2Nb21P2Nb2P72PNQbPPR2R3K-
 Win at Chess #266 MV=Rh8xh2
 2r1kb1rpp3ppp2n1b31q1N2B11P2Q38P4PPP3RK1NR+
 Win at Chess #267 MV=Nd5-c7
 2r3krppp2n1p7B5q1N1bp52Pp4PP2RPPPR2Q2K1+
 Win at Chess #268 MV=Re2-e8
 2kr2nrpp1n1ppp2p1p3q71b1P1B2P1N2Q1P1PP1BPP1R3K2R+
 Win at Chess #269 MV= a3xb4
 2r1r1k1pp1q1ppp3p1b23P43Q45N2PP2RPPP4R1K1+
 Win at Chess #270 MV=Qd4-g4
 2kr4ppp3Pp4RP1B2r55P21P6P2p43K4+
 Win at Chess #271 MV=Re6-d6
 nrq4r2k1p31p1pPnp1pRpP1p2P1P2P22P1BB21R2Q1P16K1+
 Win at Chess #272 MV=Be3xc5
 2k4Bbpp1qp2p1b57p1PN1n1p12Pr4P5PPR3QR1K-
 Win at Chess #273 MV=Ne4-g3
 81p6p5R1k7Prpp4K71NP58+
 Win at Chess #274 MV=Rg6-b6
 r1b2rk11p1n1pppp1p2q24p3P1B1Pn21QN2N21P3PPP3R1RK1-
 Win at Chess #275 MV=Nf4xg2
 r5k1pp1RR1pp1b66r12p5B6PP4qPK3Q4+
 Win at Chess #276 MV=Qd1-d5
 1r4r1p2kb2pbq2p33p1p25P22BB3QPP4PP3RKR2-
 Win at Chess #277 MV=Rg8-g3 or MV=Rg8xg2
 r2qkb1rpppb2pp3p1n25pN12BQP32N5PPP2PPPR1B1K2R+
 Win at Chess #278 MV=Bc4-f7
 r74b32p1r1k11p1pPp1q1P1P1P1pPR2NRpP2Q3K18+
 Win at Chess #279 MV=Ne3xf5
 r1r2bk15p1ppn4p1N2b43Pp3B3P32q1BPPPRQ3RK1-
 Win at Chess #280 MV=Bf8xa3

2R52R4p5p1k6n181P2QPPqr76K1+
 Win at Chess #281 MV=Rc7xh7
 6k12p3p11p1p1nN11B1P44PK282r3b17R+
 Win at Chess #282 MV=Rh1-h8
 3q1rk14bp1p1n2P2Q3p1p26x1Pp2R2N1B4PP7K+
 Win at Chess #283 MV=Nh3-g5
 3r3kpp4pp81P63N4Pn2P1qb1B1Q2B12R3K1+
 Win at Chess #284 MV=Nd4-f5
 2rr3k1b2bppPp2p1n2R73P41qB2P21P4Q11K5R+
 Win at Chess #285 MV=Qg2xg7
 3r1k21p6p4P22pP2Qb81P1KB3P6r8-
 Win at Chess #286 MV=Rd8xd5
 rn3k1rpp2bBpp2p2n2q5N13P41P6P1P3PPR1BQ1RK1+
 Win at Chess #287 MV=Qd1-h5
 r1b2rk1p4ppp1p1Qp34P2N1P68P3qPPP3R1RK1+
 Win at Chess #288 MV=Nh5-f6
 2r3k15p1pp3q1p12n3P11p1QP2P1P4N1PK62R5-
 Win at Chess #289 MV=Qe6-e5
 2k2r22p51pq5p1p1n3P1P2n1B1R4Pp2QR46K1-
 Win at Chess #290 MV=Nf4-e2
 5r1k3b2p1p6p1pRpR31P1P2q1P4pP15QnP1B4K1+
 Win at Chess #291 MV= h2-h3
 4r31Q1qk2pp4pp13Pb3P76PP5P24R1K1+
 Win at Chess #292 MV= d5-d6
 1nbq1r1k3rbp1pp1p1pp1Q1p6P1pPN35NP11P2PPBPR4RK1+
 Win at Chess #293 MV=Nf3-g5
 3r3k1r3p1pp1pB1p28p1qNP1Q1P6P1P4P13R3K+
 Win at Chess #294 MV=Nd5-f5 or MV=Bd6-f8
 4r3p4r1pR1p2pp11p1bk34pNPP2P1K32P2P23R4+
 Win at Chess #295 MV=Rd1xd5
 3r41p3k1pp1b1p1p14Q1Pn2B3KP4pP2PP2R1N16q1-
 Win at Chess #296 MV=Rd8-d4 or MV=Bc6-d7 or MV=Bc6xf3
 3r1rk1p3qp1p2bb2p12p53P41P6PBQN1PPP2R2RK1-
 Win at Chess #297 MV=Bd6xh2
 3Q4p3b1k12p2rPp2q54B3P2P47P6RK+
 Win at Chess #298 MV=Qd8-h8
 1n2rr21pk3pppNn2p22N1p386P1PP2PPKP2RR4+
 Win at Chess #299 MV=Nc5-a4
 b2b1r1k3R1ppp4qP24p1PQ4P35B24N1K18+
 Win at Chess #300 MV= g5-g6

Appendix 3: Results of TECH3 on Test Problems

Summarized output from TECH3 on each of the test problems is listed on the next 14 pages. After each iteration a line is printed. The information on each line consists of:

(1) The iteration depth. If the iteration depth equals 31 then a forced mating combination has been found and clearly it is not necessary to search further.

(2) The change in score. A positive change is good for the side-to-move. For example a change of 5 means that the side-to-move can improve the material balance from his point of view by the equivalent of a rook over what it is in the given position. If the iteration depth is 31 then the change in score is large since the value of winning is 63.

(3) The move TECH3 would play. This is compared with the problem solutions to determine if TECH3 gets credit.

(4) The CPU time, in seconds, used so far in the computation. The notation ">300." means that the next iteration could not be completed in less than 300 seconds.

Win at Chess # 1	4 4 Re7-f7 .139	4 0 Nc4-e5 .182
2 0 Ne5xc6 .008	5 5 Re7-f7 .361	5 0 Nc4-e5 .346
3 0 Ne5xc6 .057	6 5 Re7-f7 .870	6 1 Nc4-e5 2.98
31 65 Qg3-g6 .314		
Win at Chess # 2	Win at Chess # 9	Win at Chess # 18
2 0 Kf6-f7 .002	2 1 Rh4xg4 .011	2 0 Kg1-h1 .004
3 0 Kf6-f7 .016	3 3 Bd6-h2 .044	3 0 Kg1-h1 .011
4 0 Kf6-f7 .047	4 3 Bd6-h2 .125	4 0 Kg1-h1 .045
5 0 Kf6-f7 .094	31 66 Bd6-h2 .139	5 0 Kg1-h1 .102
6 0 Kf6-f7 .209		6 0 Kg1-f1 .420
7 0 Kf6-f7 .550	Win at Chess # 10	7 0 Kg1-f1 .639
8 0 Kf6-f7 1.17	2 2 Rh4xh7 .023	8 4 Ra8-h8 1.72
9 0 Kf6-f7 2.44	3 2 Rh4xh7 .059	9 4 Ra8-h8 2.36
10 0 Kf6-f7 6.56	4 2 Rh4xh7 .193	10 4 Ra8-h8 3.46
11 0 Kf6-f7 12.2	5 3 Rh4xh7 .495	11 4 Ra8-h8 6.24
12 0 Kf6-f7 24.5	6 3 Rh4xh7 1.50	
13 0 Kf6-f7 58.3	Win at Chess # 11	Win at Chess # 19
14 0 Kf6-f7 149.	2 0 Qd1xd7 .008	2 0 c5-c6 .005
>300.	3 1 Bf3xc6 .043	3 0 c5-c6 .151
	4 2 Bf3xc6 .198	4 1 c5-c6 .239
Win at Chess # 3	5 7 Bf3xc6 .599	5 0 c5-c6 .407
2 1 Re3-g3 .036		6 2 c5-c6 1.96
3 1 Re3-g3 .051	Win at Chess # 12	Win at Chess # 20
4 1 Re3-g3 .156	31 67 Qg4xf3 .003	2 0 g7-g6 .003
5 1 Re3-g3 .395		3 0 f7-f6 .135
6 1 Re3-g3 1.15	Win at Chess # 13	4 6 Bd7-b5 .273
Win at Chess # 4	2 2 Qf1xf8 .011	5 7 Bd7-b5 .390
31 63 Qh6xh7 .006	3 2 Qf1xf8 .040	6 7 Bd7-b5 .985
	4 2 Qf1xf8 .111	
Win at Chess # 5	5 2 Qf1xf8 .285	Win at Chess # 21
31 66 Qc6-c4 .006	6 2 Qf1xf8 .818	2 0 Qd2xc3 .009
		3 0 Qd2xc3 .040
Win at Chess # 6	Win at Chess # 14	4 0 Qd2xc3 .178
2 1 Kh6xg5 .002	2 0 Bd3-e2 .009	5 1 Qd2-h6 .785
3 1 Kh6xg5 .009	3 3 Qh3xh7 .070	
4 5 Rb6-b7 .029	4 3 Qh3xh7 .224	Win at Chess # 22
5 5 Rb6-b7 .048	5 3 Qh3xh7 .838	2 0 Bc4-e2 .012
6 6 Rb6-b7 .086		3 1 Ng5xf7 .047
7 6 Rb6-b7 .153	Win at Chess # 15	4 1 Ng5xf7 .107
8 6 Rb6-b7 .257	2 0 b2-b3 .007	5 1 Ng5xf7 .252
9 6 Rb6-b7 .414	3 2 Rb8xb7 .030	6 1 Ng5xf7 .835
10 6 Rb6-b7 .676	4 2 Rb8xb7 .089	
	5 2 Rb8xb7 .278	Win at Chess # 23
	6 2 Rb8xb7 .726	2 0 Nc5xe6 .004
Win at Chess # 7	Win at Chess # 16	3 0 Nc5xe6 .088
2 1 Ng4xe5 .002	2 0 Be3xc5 .001	4 0 Nc5xe6 .187
3 1 Ng4xe5 .014	3 0 Be3xc5 .024	5 0 Nc5xe6 .437
4 1 Ng4xe5 .052	4 0 Be3xc5 .062	6 3 g2-g4 2.82
5 6 Ng4-e3 .153	5 0 Be3xc5 .169	
6 6 Ng4-e3 .400	6 3 Ne2-c3 1.02	Win at Chess # 24
7 6 Ng4-e3 1.22		2 0 Qe7xe3 .005
Win at Chess # 8	Win at Chess # 17	3 0 Qe7xe3 .026
2 1 Nh6-f7 .017	2 0 Nc4-e5 .005	4 6 Bg7-d4 .137
3 4 Re7-f7 .055	3 0 Nc4-e5 .127	5 6 Bg7-d4 .247
		6 6 Bg7-d4 .634

Win at Chess # 25	4 0 Qe4-e5 .186	7 4 Kb5-a5 1.26
2 9 Qg4-h4 .010	5 0 Qe4-e5 .486	Win at Chess # 42
3 9 Qg4-h4 .025	6 3 Qe4-e5 2.95	2 0 Bb4-d6 .023
4 9 Qg4-h4 .065	Win at Chess # 34	3 0 Bb4-d6 .210
5 9 Qg4-h4 .152	2 -2 Bb7xe4 .060	4 0 Bb4-d6 .357
6 9 Qg4-h4 .321	3 2 Bd4-g1 .112	5 0 Bb4-d6 .525
7 10 Qg4-h4 .782	4 2 Bd4-g1 .145	6 3 Bb4-a5 2.84
Win at Chess # 26	5 2 Bd4-g1 .248	7 3 Bb4-a5 4.37
2 1 Qb6xb2 .002	6 2 Bd4-g1 .996	Win at Chess # 43
3 2 Bd7-f5 .138	Win at Chess # 35	2 0 Ba3-d6 .024
4 2 Bd7-f5 .195	2 0 Rc7-f7 .007	3 2 Ba3-e7 .090
5 2 Bd7-f5 .294	3 0 Rc7-f7 .036	4 2 Ba3-e7 .182
6 2 Bd7-f5 1.43	31 62 Rh4xh7 .138	5 2 Ba3-e7 .340
Win at Chess # 27	Win at Chess # 36	6 5 Ba3-e7 .944
31 66 Qa3-f8 .028	2 4 Re7-e1 .007	Win at Chess # 44
Win at Chess # 28	3 4 Re7-e1 .033	2 0 Ne6-f8 .021
2 0 Qb4xb2 .008	4 4 Re7-e1 .164	3 3 d5xc4 .105
3 0 Qb4xb2 .027	5 4 Re7-e1 .459	4 3 d5xc4 .160
4 2 Qb4-e1 .135	6 4 Re7-e1 1.45	5 3 d5xc4 .256
5 2 Qb4-e1 .265	Win at Chess # 37	6 3 d5xc4 1.02
6 2 Qb4-e1 .713	2 1 Nc6xd4 .021	Win at Chess # 45
Win at Chess # 29	3 1 Nc6xd4 .063	2 1 Qf1xa1 .004
2 0 Bf4-g5 .010	4 1 Nc6xd4 .215	3 5 Qf1xa1 .026
3 0 Bf4-g5 .086	5 1 Nc6xd4 .621	4 5 Qf1xa1 .084
4 0 Bf4-g5 .161	Win at Chess # 38	5 5 Qf1xa1 .292
5 0 Bf4-g5 .325	2 4 Rd2-d8 .010	6 5 Qf1xa1 .721
6 2 c5-c6 1.29	3 4 Rd2-d8 .031	Win at Chess # 46
Win at Chess # 30	4 4 Rd2-d8 .113	2 0 Nc3-e2 .012
2 0 Ne4-g5 .018	5 4 Rd2-d8 .336	3 0 Nc3-e2 .193
3 0 Ne4-g5 .106	6 4 Rd2-d8 .941	4 0 Nc3-e2 .346
4 0 Ne4-g5 .345	Win at Chess # 39	5 0 Nc3-e2 .923
5 1 Ne4xd6 .840	2 0 Bf4-d2 .012	6 0 Nc3-e2 5.92
Win at Chess # 31	3 4 Nc3-a4 .063	7 1 Nc3-b5 12.2
2 1 d5xe6 .005	4 4 Nc3-a4 .102	8 1 Nc3-b5 15.7
3 2 d5xe6 .065	5 4 Nc3-a4 .159	9 2 Nc3-b5 42.4
4 2 d5xe6 .130	6 4 Nc3-a4 .593	Win at Chess # 47
5 2 d5-d6 .295	Win at Chess # 40	2 1 Nc6xd4 .005
6 2 d5-d6 .793	2 0 Rd8xd4 .004	3 1 Nc6xd4 .024
Win at Chess # 32	3 4 Rd8-c8 .038	4 1 Nc6xd4 .094
2 0 Ng3-h1 .006	4 4 Rd8-c8 .099	5 1 Nc6xd4 .296
3 1 Qd1-d8 .073	5 4 Rd8-c8 .215	6 1 Nc6xd4 1.13
4 1 Qd1-d8 .138	6 4 Rd8-c8 .512	Win at Chess # 48
5 1 Qd1-d8 .272	Win at Chess # 41	2 0 Rb8-b7 .079
6 1 Qd1-d8 .820	2 4 Rf7-f8 .005	3 0 Rb8-b7 .092
Win at Chess # 33	3 4 Rf7-f8 .013	4 0 Rb8-b7 .224
2 0 Qe4-e5 .009	4 3 Rf7-f8 .087	5 0 Rb8-b7 .435
3 0 Qe4-e5 .059	5 4 Rf7-e7 .270	6 1 Rb8-b4 1.89
	6 4 Rf7-e7 .501	Win at Chess # 49

2 -1 e4-e5 .067	Win at Chess # 59	7 3 Qe2xe3 1.18
3 -1 e4-e5 .128	2 1 Qd4xg7 .015	
4 -1 e4-e5 .345	3 1 Qd4xg7 .039	Win at Chess # 69
5 0 Qh5-h6 .906	4 3 Nc3-d5 .142	2 0 Qe5-h8 .009
6 0 Qh5-h6 1.61	5 4 Nc3-d5 .349	3 1 Re6-e8 .056
7 6 Qh5xh7 8.50	6 4 Nc3-d5 .980	4 1 Re6-e8 .156
		5 1 f2-f3 .504
Win at Chess # 50	Win at Chess # 60	Win at Chess # 70
2 6 Rb7xb6 .024	31 66 Qh3-h8 .008	2 6 Nb4-a2 .033
31 70 Rb7xb6 .030		3 6 Nb4-a2 .070
Win at Chess # 51	Win at Chess # 61	4 6 Nb4-a2 .191
2 0 Nd4-f3 .019	31 65 Qf3-f7 .009	5 6 Nb4-a2 .534
3 5 Rf4-g4 .120	Win at Chess # 62	Win at Chess # 71
4 6 Rf4-g4 .316	2 0 Rg6-g5 .017	2 0 Qh3xd7 .006
5 10 Rf4-g4 1.25	3 1 Rg6xg3 .066	3 0 Qh3xd7 .039
	4 1 Rg6xg3 .185	4 0 Qh3xd7 .165
Win at Chess # 52	5 1 Rg6xg3 .902	5 0 Qh3xd7 .415
2 0 Nd5-c3 .014	Win at Chess # 63	6 0 Qh3xd7 1.89
3 0 Nd5-c3 .049	2 0 Ne5-g4 .019	7 0 Qh3xd7 6.70
4 0 Nd5-c3 .295	3 1 Ne5xf7 .180	8 0 Qh3xd7 25.5
5 1 Re1-e7 .906	4 1 Ne5xf7 .398	9 0 Qh3xd7 78.6
Win at Chess # 53	5 1 Ne5xf7 .805	10 0 Qh3xd7 210.
2 0 Re3xf3 .009		>300.
3 0 Re3xf3 .109	Win at Chess # 64	Win at Chess # 72
4 4 Re3-e1 .213	2 0 Qe3-f3 .027	2 1 e5-e6 .013
5 4 Re3-e1 .423	31 62 g2-g4 .091	3 1 e5-e6 .042
6 4 Re3-e1 .955		4 2 e5-e6 .102
Win at Chess # 54	Win at Chess # 65	5 2 e5-e6 .219
31 62 Qh5-h1 .011	2 0 Qh6xf8 .030	6 2 e5-e6 .656
Win at Chess # 55	3 5 Nd5-e7 .083	
2 0 Qd4-e5 .007	4 9 Nd5-e7 .218	Win at Chess # 73
3 0 Qd4-f2 .205	5 9 Nd5-e7 .593	2 0 Nc3-d5 .010
4 0 Qd4-f2 .441	Win at Chess # 66	3 0 Nc3-d5 .031
5 0 Qd4-f2 .838	2 0 Qc7xe5 .002	4 0 Nc3-d5 .181
31 64 Qd4xg7 1.51	3 0 Qc7xe5 .023	5 3 Qe2-d2 .535
Win at Chess # 56	4 0 Qc7xe5 .099	
2 5 Bc5xf2 .017	5 0 Qc7xe5 .208	Win at Chess # 74
3 5 Bc5xf2 .064	6 0 Qc7xe5 .481	2 2 Qf4-f1 .009
4 5 Bc5xf2 .209	7 0 Qc7xe5 1.61	3 2 Qf4-f1 .027
5 5 Bc5xf2 .743	Win at Chess # 67	4 3 Qf4-f1 .126
Win at Chess # 57	2 0 Qd1xd5 .006	5 3 Qf4-f1 .331
2 0 Rf3-f8 .011	3 4 Re5xd5 .032	6 3 Qf4-f1 1.07
31 65 Rf3-f8 .016	4 4 Re5xd5 .083	Win at Chess # 75
Win at Chess # 58	5 4 Re5xd5 .196	2 0 Re8xe1 .003
2 0 Rc6-c8 .017	6 4 Re5xd5 .524	3 0 Re8xe1 .031
3 1 Rc6-c5 .168	Win at Chess # 68	4 0 Re8xe1 .122
4 1 Rc6-c5 .388	2 0 Rc3-c4 .003	5 0 Re8xe1 .378
5 4 Qd3-d1 1.12	3 1 Qe2xe3 .019	6 6 Qd7-d6 1.18
	4 2 Qe2xe3 .067	Win at Chess # 76
	5 3 Qe2xe3 .122	2 0 Bb3xe6 .017
	6 3 Qe2xe3 .319	3 0 Bb3xe6 .068

4	0	Bb3xe6	.220	3	0	Re8xb8	.029	Win at Chess # 91
5	0	Bb3xe6	.613	4	3	Nc5-a6	.145	2 0 Nd4-e6 .012
6	2	Bg5xf6	2.33	5	4	Nc5-a6	.287	3 0 Nd4-e6 .050
7	2	Bg5xf6	7.38	6	4	Nc5-a6	.752	4 0 Nd4-e6 .341
Win at Chess # 77				Win at Chess # 86				5 0 Nd4-e6 .580
2	0	Nd4-e6	.008	2	0	Nf6-g8	.004	6 0 Nd4-e6 1.59
3	0	Nd4-e6	.094	3	0	Nf6-g8	.024	7 0 Nd4-e6 5.91
4	5	Nf5-g3	.320	4	0	Nf6-g8	.071	8 0 Nd4-e6 13.1
5	6	Nf5-g3	.634	5	0	Nf6-g8	.195	9 0 Nd4-e6 38.9
Win at Chess # 78				6	0	Nf6-g8	.538	10 1 Bb3-e6 141.
2	1	Ne4-g5	.034	7	0	Nf6-g8	1.40	11 1 Bb3-e6 261.
3	2	Ne4-g5	.121	8	0	Nf6-g8	3.56	Win at Chess # 92
4	2	Ne4-g5	.412	9	0	Nf6-g8	8.68	2 0 Nf6-d7 .011
5	2	Ne4-g5	1.42	10	0	Nf6-g8	20.7	3 0 Nf6-d7 .038
Win at Chess # 79				11	0	Nf6-g8	49.5	4 0 Nf6-d7 .137
2	-1	Nf6-d7	.330	12	0	Nf6-g8	117.	5 0 Nf6-d7 .663
31	64	Qh3xh2	.380	13	0	Nf6-g8	287.	6 0 Nf6-d7 2.44
Win at Chess # 80				Win at Chess # 87				7 0 Nf6-d7 8.88
2	0	d4-d5	.011	2	0	c5xd4	.004	8 0 Nf6-d7 23.5
3	0	d4-d5	.083	3	0	c5xd4	.018	9 0 Nf6-d7 74.7
4	0	d4-d5	.199	4	0	c5xd4	.056	10 0 Nf6-d7 273.
5	0	d4-d5	.580	5	0	c5xd4	.138	Win at Chess # 93
6	0	d4-d5	3.34	6	0	c5xd4	.356	2 0 Qg7xf6 .009
7	0	d4-d5	7.45	7	0	c5xd4	.939	3 0 Qg7xf6 .032
8	0	d4-d5	21.8	8	0	c5xd4	2.29	4 0 Qg7xf6 .130
9	0	d4-d5	103.	9	0	c5xd4	5.42	5 0 Qg7xf6 .405
10	4	Rd1-a1	230.	10	0	c5xd4	11.8	6 0 Qg7xf6 1.30
			>300.	11	0	c5xd4	26.9	7 0 Qg7xf6 3.69
				12	0	c5xd4	62.1	8 0 Qg7xf6 10.8
				13	0	c5xd4	143.	9 2 Bc1-h6 33.4
							>300.	Win at Chess # 94
Win at Chess # 81				Win at Chess # 88				2 5 e5-e4 .038
2	-2	Ra8xa3	.029	2	0	Be6xa2	.012	3 5 e5-e4 .098
3	-2	Ra8xa3	.059	3	0	Be6xa2	.083	4 5 e5-e4 .283
4	-2	Ra8xa3	.111	31	65	Rg6xg2	.277	5 5 e5-e4 .833
5	-1	Be7-d6	.593	Win at Chess # 89				Win at Chess # 95
Win at Chess # 82				2	0	Qd5-d6	.007	2 5 Nf6-g4 .015
2	0	Be4-h7	.035	3	0	Qd5-d6	.188	3 5 Nf6-g4 .029
3	0	Be4-h7	.110	4	0	Qd5-d6	.344	4 5 Nf6-g4 .064
4	0	Be4-h7	.335	5	0	Qd5-d6	.657	5 5 Nf6-g4 .250
5	0	Be4-h7	.976	6	3	g7-g5	3.90	6 5 Nf6-g4 .753
Win at Chess # 83				7	3	g7-g5	8.45	Win at Chess # 96
2	3	Re7xe8	.009	Win at Chess # 90				2 1 Qh4-d8 .025
3	3	Re7xe8	.045	2	0	Nf5-e3	.014	3 1 Qh4-d8 .051
4	3	Qd4xd7	.211	3	0	Nf5-e3	.075	4 1 Qh4-d8 .155
5	3	Qd4xd7	.700	4	0	Nf5-e3	.426	5 1 Qh4-d8 .466
Win at Chess # 84				5	0	Nf5-e3	.772	6 1 Qh4-d8 1.26
31	65	Qd5-g8	.007	6	1	Nf5xg7	1.67	Win at Chess # 97
Win at Chess # 85				7	1	Nf5xg7	3.52	2 0 Be5-f4 .010
2	0	Re8xb8	.007	8	1	Nf5xg7	8.08	31 61 Qg2-a8 .037

Win at Chess # 98	2 0 Nd4-f5 .007	4 2 Qd8xf6 .047
2 1 Nc5xe4 .009	3 1 Qh5xh4 .057	5 2 Qd8xf6 .152
3 2 Nc5xe4 .140	4 1 Qh5xh4 .588	6 2 Qd8xf6 .412
4 5 Nc5-b3 .250	Win at Chess #106	7 2 Qd8xf6 1.33
5 5 Nc5-b3 .376	2 0 Ne4-f2 .012	Win at Chess #114
6 6 Nc5xe4 .961	3 1 Rf8xf1 .059	2 0 g4xh5 .015
Win at Chess # 99	4 6 Ne4-f2 .221	3 0 g4xh5 .051
2 0 Re5-e4 .037	5 7 Ne4-f2 .555	4 1 Bd3xh7 .176
3 0 Re5-e4 .106	Win at Chess #107	5 1 Bd3xh7 .400
31 63 Re5-h5 .269	2 0 Rb7xc7 .004	6 1 Bd3xh7 1.87
Win at Chess #100	3 0 Rb7xc7 .014	Win at Chess #115
2 0 Bd2-e3 .003	4 0 Rb7xc7 .036	2 0 Ne8-d6 .014
3 0 Bd2-e3 .008	5 0 Rb7xc7 .075	3 0 Ne8-d6 .027
4 0 Bd2-e3 .020	6 3 Nd4-b5 .288	4 1 Ne8-d6 .107
5 0 Bd2-e3 .043	7 3 Nd4-b5 .497	5 2 Ne8-d6 .193
6 0 Bd2-e3 .084	8 3 Nd4-b5 .900	6 2 Ne8-d6 .536
7 0 Bd2-e3 .176	Win at Chess #108	Win at Chess #116
8 0 Bd2-e3 .338	2 0 Qc5-d6 .012	2 0 b5xa4 .005
9 0 Bd2-e3 .615	3 0 Qc5-d6 .075	3 0 b5xa4 .069
10 0 Bd2-e3 1.21	4 0 Qc5-d6 .303	4 0 b5xa4 .180
11 0 Bd2-e3 2.23	5 0 Qc5-d6 1.04	5 0 b5xa4 .424
12 0 Bd2-e3 4.25	6 4 Qc5-e5 3.03	6 0 b5xa4 2.81
13 0 Bd2-e3 7.41	7 4 Qc5-e5 7.55	7 0 b5xa4 7.22
14 0 Bd2-e3 12.8	Win at Chess #109	8 1 Rd8-d2 19.3
15 0 Bd2-e3 22.6	2 0 Qd6xa3 .006	Win at Chess #117
16 0 Bd2-e3 61.3	3 0 Qd6xa3 .038	2 0 Qa7xf2 .005
17 0 Bd2-e3 144.	4 0 Qd6xa3 .108	3 0 Qa7xf2 .023
18 0 Bd2-e3 212.	5 0 Qd6xa3 .280	4 4 Nd6-e4 .104
>300.	6 1 c4-c3 1.02	5 4 Nd6-e4 .183
Win at Chess #101	Win at Chess #110	6 4 Nd6-e4 .391
2 0 Bd4-e5 .002	2 0 d6-d5 .006	7 4 Nd6-e4 1.18
3 0 Bd4-e5 .028	3 0 d6-d5 .070	Win at Chess #118
4 0 Bd4-c3 .096	4 6 Ba7-e3 .239	2 0 Rf4-f5 .013
5 0 Bd4-c3 .152	5 6 Ba7-e3 .371	3 0 Rf4-f5 .147
6 0 Bd4-c3 .588	6 6 Ba7-e3 1.01	4 0 Rf4-f5 .307
Win at Chess #102	Win at Chess #111	5 0 Rf4-f5 .838
2 13 Qc8xf8 .009	2 1 Qg1xh2 .017	6 0 Rf4-f5 4.17
31 63 Qc8xf8 .014	3 0 Qg1xh2 .124	7 4 Rf4-h4 11.1
Win at Chess #103	4 4 Qg1-f1 .395	8 4 Rf4-h4 19.1
2 1 Rg1xg6 .039	5 5 Qg1-f1 .871	Win at Chess #119
3 1 Rg1xg6 .144	Win at Chess #112	2 2 Ne4-f2 .010
31 64 Qh6xg6 .384	2 -3 Re1xe6 .080	3 2 Ne4-f2 .031
Win at Chess #104	3 1 Re1xe6 .119	4 2 Ne4-f2 .121
2 0 Ne5xf7 .041	4 1 Re1xe6 .360	5 3 Qd8xd3 .440
3 4 Qe2xh5 .135	5 1 Re1xe6 1.01	6 3 Qd8xd3 1.35
4 4 Qe2xh5 .331	Win at Chess #113	Win at Chess #120
31 64 Qe2xh5 .617	2 2 g7xf6 .005	2 1 Bf6xg7 .009
Win at Chess #105	3 2 Qd8xf6 .015	3 1 Bf6xg7 .097
		4 1 Bf6xg7 .233

5	1	Bf6xg7	.515	4	5	Qf7-g6	.325	3	0	Qd7xa7	.142
6	0	Bf6xg7	2.01	5	5	Qf7-g6	.673	4	0	Qd7xa7	.268
7	0	Bf6xg7	6.60					5	0	Qd7xa7	.492
8	2	g5-g6	18.5	Win at Chess #129				6	2	Ne6-d4	2.17
Win at Chess #121				2	0	g7-g5	.013	Win at Chess #136			
2	0	Bh6xe3	.005	3	0	g7-g5	.060	2	1	Qg6xf5	.011
3	7	Bc6xf3	.032	4	0	g7-g5	.175	31	62	Rc1-c8	.067
4	7	Bc6xf3	.087	5	0	g7-g5	.442	Win at Chess #137			
5	7	Bc6xf3	.236	6	0	g7-g5	1.93	2	0	Nf5-h4	.007
6	7	Bc6xf3	.712	7	0	g7-g5	5.51	3	0	Nf5-h4	.168
Win at Chess #122				8	0	g7-g5	17.2	4	0	Nf5-h4	.454
2	7	Rd1xf1	.015	9	1	Bb7-f3	55.0	5	0	Nf5-h4	1.01
3	7	Rd1xf1	.055	10	1	Bb7-f3	106.	6	1	Rd1-d7	7.09
4	7	Rd1xf1	.208	11	1	Bb7-f3	223.				
5	7	Rd1xf1	.788				>300.	Win at Chess #138			
Win at Chess #123				Win at Chess #130				2	0	d5-d6	.009
2	0	Qf3-f6	.010	2	0	Ne3-c4	.022	3	0	d5-d6	.041
3	1	Re7-c7	.075	3	0	Ne3-c4	.043	4	0	d5-d6	.215
4	1	Re7-c7	.209	4	0	Ne3-c4	.109	5	0	d5-d6	.690
5	3	Bb7xd5	.680	5	0	Ne3-c4	.276	6	0	d5-d6	2.60
Win at Chess #124				6	0	Ne3-c4	.984	7	0	d5-d6	8.57
2	0	Rd7-d8	.004	7	0	Rf6-h6	4.26	8	6	h4-h5	39.5
3	0	Rd7-d8	.029	8	0	Rf6-h6	7.39	Win at Chess #139			
4	0	Rd7-d8	.155	9	1	Qg7-h8	26.8	2	0	Kf2-g3	.100
5	0	Rd7-d8	.541	Win at Chess #131				3	0	Kf2-g3	.157
6	2	g4-g3	1.42	2	0	Bd4-e5	.006	31	65	Ne4-f6	.310
7	2	g4-g3	2.89	3	0	Bd4-e5	.070	Win at Chess #140			
8	2	g4-g3	6.94	4	1	Qf4-e5	.253	2	0	h6xg7	.010
Win at Chess #125				5	1	Qf4-e5	.398	3	0	h6xg7	.046
2	1	Bb6xd4	.016	6	1	Qf4-e5	1.82	4	2	Be5-c7	.358
3	1	Bb6xd4	.046	7	3	Re1-e8	5.68	5	2	Be5-c7	.661
4	3	Bb6xd4	.135	Win at Chess #132				6	4	Be5-c7	1.76
5	3	Bb6xd4	.288	2	0	Qh3xf1	.009	7	8	Rc3-c7	6.29
6	3	Bb6xd4	.937	3	0	Qh3xf1	.031	Win at Chess #141			
Win at Chess #126				4	0	Qh3xf1	.111	2	0	Kg2-f1	.010
2	6	Rf6xc6	.029	5	0	Qh3xf1	.353	3	0	Kg2-f1	.030
3	6	Rf6xc6	.098	6	0	Qh3xf1	.969	4	0	Kg2-f1	.142
4	6	Rf6xc6	.320	31	62	Re5-e1	3.59	5	0	Kg2-f1	.350
5	6	Rf6xc6	1.07	Win at Chess #133				6	0	Kg2-f1	2.36
Win at Chess #127				2	0	e5-e6	.012	7	0	Kg2-f1	3.61
2	0	f4xg5	.003	3	0	e5-e6	.130	8	0	Kg2-f1	8.79
3	0	f4xg5	.020	4	0	e5-e6	.544	9	0	Kg2-f1	26.1
4	0	f4xg5	.076	5	0	e5-e6	1.36	10	0	Kg2-f1	78.3
5	0	f4xg5	.228	6	1	Bg3-h4	4.43	11	0	Kg2-f1	239.
6	1	Rb2xb7	.957	Win at Chess #134							>300.
Win at Chess #128				2	1	Qe5xh2	.009	Win at Chess #142			
2	2	Bf6xg7	.011	3	1	Qe5xh2	.094	2	0	d4-d5	.015
3	2	Bf6xg7	.099	31	65	Rd8-d1	.303	3	1	Nc4-e3	.137
				Win at Chess #135				4	3	Re1-e8	.444
				2	0	Qd7xa7	.003	5	3	Re1-e8	1.10

Win at Chess #143	4 1 Be4xg2 .296	5 0 Qg3xe5 .430
2 -1 Qg3-f3 .085	5 1 Be4xg2 .677	6 0 Qg3xe5 1.18
31 63 Rg6xh6 .175	Win at Chess #150	7 3 h3xg4 6.68
Win at Chess #144	2 1 Qb2xe2 .004	8 3 h3xg4 12.2
2 0 Bg4xe2 .001	3 2 Bd6-f8 .067	9 3 h3xg4 25.3
3 0 Bg4xe2 .021	4 2 Bd6-f8 .159	10 3 h3xg4 65.8
4 0 Bg4xe2 .055	5 1 Bd6-f8 .629	11 3 Nd5-e7 172.
5 0 Bg4xe2 .150	Win at Chess #151	>300.
6 0 Bg4xe2 .735	2 0 Nc4-d6 .007	Win at Chess #158
7 2 d4-d3 2.66	3 0 Nc4-d6 .048	2 1 Qf5-e6 .013
8 2 d4-d3 5.14	4 0 Nc4-d6 .146	31 64 Re7xg7 .030
Win at Chess #145	5 0 Nc4-d6 .244	Win at Chess #159
2 1 Bg5xf6 .016	6 0 Nc4-d6 1.47	2 5 Ng5-e6 .017
3 1 Bg5xf6 .086	7 0 Nc4-d6 3.07	3 5 Ng5-e6 .036
4 1 Bg5xf6 .256	8 0 Nc4-d6 6.28	4 5 Ng5-e6 .111
5 1 Bg5xf6 .829	9 1 Na4-c3 24.3	5 6 Ng5-e6 .355
6 1 Bg5xf6 3.82	Win at Chess #152	6 6 Ng5-e6 .922
7 1 Bg5xf6 12.4	2 1 Nc3-d5 .019	Win at Chess #160
8 7 Re1-e8 43.4	3 1 Nc3-d5 .082	31 63 Qg4xd7 .002
Win at Chess #146	4 1 Nc3-d5 .170	Win at Chess #161
2 0 Bf5-g6 .002	5 1 Nc3-d5 .352	2 0 Nd6-e8 .024
3 0 Bf5-g6 .013	6 1 Nc3-d5 2.12	3 2 Rg3-g8 .144
4 0 Bf5-g6 .026	Win at Chess #153	31 64 Qh4xd8 .293
5 0 Bf5-g6 .052	2 0 g7-g6 .041	Win at Chess #162
6 0 Bf5-g6 .117	3 0 g7-g6 .310	2 0 Ke1-d2 .044
7 0 Bf5-h3 .344	4 0 g7-g6 .876	3 0 Ke1-d2 .063
8 0 Bf5-h3 .430	5 1 Nb5xd4 1.61	4 0 Ke1-d2 .230
9 0 Bf5-h3 .808	6 1 Nb5xd4 2.82	5 4 Bc4-d5 .872
10 -1 Bf5-h3 4.42	7 1 Nb5xd4 5.01	Win at Chess #163
11 -1 Bf5-h3 5.04	Win at Chess #154	2 0 c6xd5 .008
12 -1 Bf5-h3 6.42	31 60 Qf2xf7 .005	3 0 c6xd5 .044
13 -1 Bf5-c8 14.9	Win at Chess #155	4 0 c6xd5 .368
14 -1 Bf5-c8 31.5	2 0 Qc7-c8 .012	5 0 c6xd5 1.57
Win at Chess #147	3 0 Qc7-c8 .054	6 0 c6xd5 3.68
2 0 d5-d4 .026	4 0 Qc7-c8 .149	7 0 c6xd5 10.4
3 0 d5-d4 .118	5 0 Qc7-c8 .688	8 0 c6xd5 45.3
4 0 d5-d4 .385	6 0 Qc7-c8 1.69	9 0 c6xd5 141.
5 0 d5-d4 .843	7 0 Qc7-c8 4.43	>300.
6 1 Nf6xg4 2.34	8 0 Qc7-c8 12.5	Win at Chess #164
7 1 Nf6xg4 5.71	9 0 Qc7-c8 45.7	2 0 Nb4xd5 .009
Win at Chess #148	10 0 Qc7-c8 93.3	3 2 Rc2xc4 .059
2 0 Qd1-e2 .025	>300.	4 2 Rc2xc4 .169
3 0 Qd1-e2 .045	Win at Chess #156	5 2 Rc2xc4 .449
4 1 Rg1xg7 .167	31 66 Qh3xh6 .003	6 2 Rc2xc4 1.20
5 2 Rg1xg7 .402	Win at Chess #157	Win at Chess #165
6 6 Rg1xg7 1.38	2 0 Qg3xe5 .011	2 1 Qd2-d8 .007
Win at Chess #149	3 0 Qg3xe5 .046	3 1 Qd2-d8 .022
2 0 Be4-d5 .012	4 0 Qg3xe5 .170	4 1 Qd2-d8 .059
3 0 Be4-d5 .079		

10 6 Bd7-h3 86.7	Win at Chess #198	2 0 Nf4-h3 .016	Win at Chess #206
Win at Chess #191	3 1 Nf4-e2 .109	2 -1 a2-a1=Q.022	
2 0 Re8xc8 .002	4 1 Nf4-e2 .193	3 0 Rd6-c6 .049	
3 0 Re8xc8 .021	5 1 Nf4-e2 .357	4 0 Rd6-c6 .147	
4 7 Qe2-c4 .124	6 4 Rd8-d3 1.48	5 0 Rd6-c6 .336	
5 7 Qe2-c4 .190		6 0 Rd6-c6 .734	
31 62 Qe2-c4 .259	Win at Chess #199		Win at Chess #207
Win at Chess #192	2 0 Bb5xc6 .008	2 0 f2-f4 .027	
2 6 Nb4-d3 .028	3 0 Bb5xc6 .250	3 0 f2-f4 .043	
3 6 Nb4-d3 .081	4 0 Bb5xc6 .440	4 0 f2-f4 .229	
4 7 Nb4-d3 .357	5 1 Bb5xc6 .795	5 0 f2-f4 .445	
5 7 Nb4-d3 1.23	Win at Chess #200	6 1 Qg4xg7 1.68	
Win at Chess #193	2 0 d5xe6 .006	Win at Chess #208	
2 0 f7-f6 .009	3 0 d5xe6 .064	2 0 Qf5-f7 .010	
3 0 f7-f6 .048	4 0 d5xe6 .239	3 0 Qf5-f7 .070	
4 0 f7-f6 .160	5 0 d5xe6 .780	4 0 Qf5-f7 .319	
5 0 f7-f6 .476	6 1 Bb2xf6 2.09	5 0 Qf5-f7 1.00	
6 0 f7-f6 1.65	7 1 Bb2xf6 5.35	6 0 Qf5-f7 2.56	
7 0 f7-f6 4.17	Win at Chess #201	7 3 Bh5-f7 7.73	
8 4 Qb3xe3 16.5	2 0 Bd5-e4 .011	Win at Chess #209	
Win at Chess #194	3 0 Bd5-e4 .114	2 1 a4xb5 .008	
2 -1 Nf5-h6 .040	4 3 Ra1-a7 .275	3 2 a4xb5 .075	
3 -1 Nf5-h6 .115	5 3 Ra1-a7 .486	4 5 Re1xe5 .510	
4 -1 Nf5-h6 .287	6 4 Ra1-a7 1.24		
5 -1 Nf5-h6 .952	Win at Chess #202	Win at Chess #210	
Win at Chess #195	2 1 Rc2xa2 .005	2 0 Nd5-e3 .006	
2 0 Rd1xd8 .005	3 1 Rc2xa2 .017	3 0 Nd5-e3 .064	
3 0 Rd1xd8 .040	4 1 Rc2xa2 .058	4 0 Nd5-e3 .197	
4 0 Rd1xd8 .106	5 1 Rc2xa2 .125	5 0 Nd5-e3 .488	
5 0 Rd1xd8 .306	6 1 Rc2xa2 .298	6 1 Be2-b5 1.76	
6 0 Rd1xd8 1.38	7 1 Rc2xa2 .717	7 1 Be2-b5 5.19	
7 0 Rd1xd8 3.27	Win at Chess #203	8 2 Be2-b5 14.5	
8 0 Rd1xd8 7.74	2 0 Qg5-h5 .010	9 2 Be2-b5 41.0	
9 2 g2-g3 27.4	3 0 Qg5-h5 .032	10 3 Rd1-h1 138.	
Win at Chess #196	4 1 Qg5-h6 .223	>300.	
2 0 d5xe4 .007	5 1 Qg5-h6 .362	Win at Chess #211	
3 0 d5xe4 .058	31 63 Qg5-h6 .907	2 1 Qh5xh7 .006	
4 0 d5xe4 .259	Win at Chess #204	3 1 Qh5xh7 .056	
5 0 d5xe4 .569	2 0 Nd5-e3 .007	4 1 Qh5xh7 .236	
6 0 d5xe4 2.28	3 1 Nd5-c7 .069	31 63 Qh5xf7 .356	
7 0 d5xe4 6.34	4 1 Nd5-c7 .209	Win at Chess #212	
8 0 d5xe4 17.5	5 1 Nd5-c7 .543	2 0 Ne5-g6 .022	
9 0 d5xe4 62.5	6 1 Re1xe5 1.73	3 0 Ne5-g6 .075	
10 0 d5xe4 202.	7 1 Re1xe5 4.80	4 0 Ne5-g6 .289	
>300.	Win at Chess #205	31 66 Qh8xg7 1.15	
Win at Chess #197	2 0 Bd4-e3 .009	Win at Chess #213	
2 0 Qf2-f1 .012	3 0 Bd4-e3 .135	2 0 Bc1-e3 .038	
31 68 Qf2-f1 .022	4 1 Qd2xg5 .402	3 0 Bc1-e3 .075	
	5 1 Qd2xg5 .895	4 0 Bc1-e3 .187	

5 0 Bc1-e3 .434	5 0 Ng3-h5 .272	5 2 d6-d5 .560
6 -1 Bc1-e3 3.94	6 0 Ng3-h5 1.06	
7 4 Rh5xh7 6.07	7 0 Ng3-h5 3.05	Win at Chess #228
8 4 Rh5xh7 9.11	8 0 Ng3-h5 5.81	2 0 Bd3xe4 .003
9 4 Rh5xh7 19.7	9 1 Ng3-h5 12.0	3 0 Bd3xe4 .066
10 4 Rh5xh7 48.1	10 2 Rd1-d8 39.7	4 0 Bd3xe4 .196
11 4 Rh5xh7 116.	11 2 Rd1-d8 78.5	5 0 Bd3xe4 .430
>300.	12 2 Rd1-d8 215.	6 0 Bd3xe4 2.65
	>300.	
Win at Chess #214	Win at Chess #222	Win at Chess #229
2 0 Ne5-f3 .012	2 1 b2xa3 .003	2 0 Ke4-e5 .004
3 0 Ne5-f3 .094	3 1 b2xa3 .029	3 0 Ke4-e5 .024
4 0 Ne5-f3 .289	4 1 b2xa3 .142	4 0 Ke4-e5 .058
5 0 Ne5-f3 .693	5 1 b2xa3 .357	5 0 Ke4-e5 .122
6 0 Ne5-f3 3.65	6 1 b2xa3 1.18	6 0 Ke4-e5 .296
7 0 Ne5-f3 13.1	7 1 b2xa3 4.02	7 0 Ke4-e5 .728
8 1 Ne4-g5 31.1	8 1 b2xa3 9.63	8 0 Ke4-e5 1.60
	9 1 b2xa3 29.7	9 0 Ke4-e5 3.69
Win at Chess #215	10 1 b2xa3 95.5	10 0 Ke4-e5 8.00
2 2 Re1xc1 .018	11 1 b2xa3 230.	11 0 Ke4-e5 17.6
3 2 Re1xc1 .060	>300.	12 0 Ke4-e5 40.3
31 66 Qd3-h7 .133		13 0 Ke4-e5 90.9
		14 0 Ke4-e5 215.
		>300.
Win at Chess #216	Win at Chess #223	Win at Chess #230
2 0 Ne5xd7 .008	2 0 Bf4-g5 .005	2 0 Kc6-d7 .001
3 0 Ne5xd7 .038	3 0 Bf4-g5 .033	3 0 Kc6-d7 .011
4 0 Ne5xd7 .145	4 0 Bf4-g5 .235	4 0 Kc6-d7 .028
5 1 Ne5xf7 .551	5 1 Nc3xe4 .782	5 0 Kc6-d7 .063
		6 0 Kc6-d7 .193
Win at Chess #217	Win at Chess #224	7 0 Kc6-d7 .447
2 1 Ne4xf6 .039	2 1 e5xf4 .006	8 0 Kc6-d7 .915
3 2 Ne4-g3 .127	3 1 e5xf4 .222	9 0 Kc6-d7 2.25
4 3 Ne4-g3 .956	4 1 e5xf4 .384	10 0 Kc6-d7 4.78
31 65 Qd4-d7 1.28	5 1 e5xf4 .672	11 0 Kc6-d7 10.1
	6 1 e5xf4 1.96	12 0 Kc6-d7 33.4
Win at Chess #218	7 1 e5xf4 5.94	13 0 Kc6-d7 64.0
2 0 h5xg6 .009	8 1 e5xf4 13.9	14 0 Kc6-d7 126.
3 0 h5xg6 .052	9 2 Rd6-h6 73.7	>300.
4 5 Bg5-h6 .285		
5 4 Bg5-h6 .636	Win at Chess #225	Win at Chess #231
	2 0 Qe7-f7 .024	2 3 Rd1xd7 .006
Win at Chess #219	31 63 Qe7-h4 .066	3 3 Rd1xd7 .025
2 1 Bc6xe4 .006	Win at Chess #226	4 3 Qf5xd7 .106
31 65 Qf7-f1 .035	2 0 Qg5-g3 .015	5 3 Qf5xd7 .231
Win at Chess #220	3 0 Qg5-g3 .137	6 1 Rd1xd7 1.37
2 0 Ne4-f6 .009	4 0 Qg5-g3 .299	7 0 Rd1xd7 3.14
3 0 Ne4-f6 .064	5 1 Rd1-d8 .726	8 0 Rd1xd7 6.66
4 0 Ne4-f6 .533	6 1 Rd1-d8 2.07	9 1 Bc1-g5 18.7
5 5 Qe2xf1 1.22	7 2 Qg5-e7 7.83	
6 4 Qe2xf1 4.83	8 3 Ne5-f7 20.4	Win at Chess #232
		2 0 Rb8xe8 .002
Win at Chess #221	Win at Chess #227	3 0 Rb8-b4 .048
2 0 Nc6-d8 .005	2 0 Bf5-g6 .008	4 0 Qa6-a7 .197
3 0 Nc6-d8 .042	3 0 Bf5-e6 .107	5 0 Qa6-a7 .278
4 0 Ng3-h5 .141	4 0 Bf5-e6 .211	

6	0	Qa6-a7	.829	3	0	Qg3xf4	.049				
7	0	Qa6-a7	2.14	4	0	Qg3xf4	.141	Win at Chess #244			
8	0	Qa6-a7	4.68	5	0	Qg3xf4	.451	2	0	Qe3-g3	.033
9	3	Rb8xe8	13.9	6	0	Qg3xf4	1.12	3	1	Rd1xd5	.203
10	3	Rb8xe8	29.4	7	1	Bg2xb7	3.36	31	64	Qe3xc5	.606
				8	3	Bg2xb7	11.2				
				9	2	Bg2xb7	24.7	Win at Chess #245			
Win at Chess #233								2	0	Bd4xf6	.005
2	0	Qe3xf2	.010	Win at Chess #239				3	1	Nc3-b5	.152
3	0	Qe3xf2	.040	2	0	f6-f5	.007	4	1	Nc3-b5	.322
4	0	Qe3xf2	.274	3	0	f6-f5	.058	5	2	Qd3xg6	.920
5	5	Nd4-b3	.836	4	0	f6-f5	.170	Win at Chess #246			
Win at Chess #234				5	0	f6-f5	.405	31	62	Qg4-h5	.013
2	1	Rc1xc4	.008	6	0	f6-f5	1.15	Win at Chess #247			
3	1	Rc1xc4	.054	7	0	f6-f5	2.98	2	-1	Qd6-c5	.060
4	1	Rc1xc4	.248	8	0	f6-f5	7.94	3	-1	Qd6-c5	.180
5	1	Rc1xc4	.633	9	1	Qf2-e2	25.7	4	-1	Qd6-c5	.417
6	1	Rc1xc4	2.10	10	1	Qf2-e2	44.2	5	-1	Qd6-c5	.971
7	1	Rc1xc4	9.56	11	1	Qf2-e2	97.3	6	0	Qd6-c5	2.20
8	4	Rh3-b3	34.6	12	1	Qf2-e2	277.	7	0	Qd6-c5	4.32
Win at Chess #235				Win at Chess #240				8	-1	Qd6-c5	18.0
2	0	Qe6xf6	.016	2	0	Ng5-f3	.012	9	-1	Qd6-c5	47.3
3	0	Qe6xf6	.057	3	0	Ng5-f3	.030	10	1	Rb1xb5	101.
4	1	Qe6-e4	.228	4	0	Ng5-f3	.120	11	3	Rb1xb5	209.
5	1	Qe6-e4	.540	5	0	Ng5-f3	.357				>300.
6	1	Qe6-e4	1.32	6	1	Qc2xc6	.892	Win at Chess #248			
7	1	Qe6-e4	3.60	Win at Chess #241				2	0	Qd6xd5	.010
8	1	Qe6-e4	9.13	2	0	Ne5xc6	.013	3	1	Qd6xd5	.035
9	1	Qe6-e4	22.5	3	0	Ne5xc6	.053	4	1	Qd6xd5	.175
10	1	Qe6-e4	58.5	4	1	Rf5xf6	.269	5	1	Qd6xd5	.565
11	1	Qe6-e4	147.	5	1	Rf5xf6	.623	6	1	Qd6xd5	1.37
			>300.	6	1	Rf5xf6	2.10	7	1	Qd6xd5	4.58
Win at Chess #236				7	1	Rf5xf6	8.18	8	1	Qd6xd5	12.1
2	1	Qg3xe1	.021	8	1	Rf5xf6	24.4	9	1	Qd6xd5	29.2
3	1	Qg3xe1	.056	9	1	Rf5xf6	77.3	10	1	Qd6xd5	91.1
4	1	Qg3xe1	.191	31	63	Qh3xh7	164.	11	1	Qd6xd5	274.
5	1	Qg3xe1	.556	Win at Chess #242				Win at Chess #249			
6	2	Rc3-c1	1.36	2	0	Ne4-g3	.007	2	0	Qd3-e3	.012
7	2	Rc3-c1	3.20	3	0	Ne4-g3	.069	3	0	Qd3-e3	.058
8	2	Rc3-c1	7.81	4	0	Ne4-g3	.215	4	0	Qd3-e2	.511
Win at Chess #237				5	1	Rd1xd7	.515	5	0	Qd3-e2	.852
2	1	Rc2xh2	.039	Win at Chess #243				6	1	d4-d5	2.79
3	-1	Qf7-f8	.705	2	0	Nd6-c4	.015	7	0	d4-d5	12.3
4	-1	Qf7-e8	1.49	3	0	Nd6-c4	.073	8	0	d4-d5	23.0
5	-1	Qf7-e8	3.41	4	0	Nd6-c4	.238	9	1	d4-d5	99.1
6	-1	Qf7-e8	10.1	5	0	Nd6-c4	.707	10	1	d4-d5	206.
7	-1	Qf7-e8	28.0	6	0	Nd6-c4	2.49				>300.
8	0	Rc2-c1	65.5	7	0	Nd6-c4	8.07	Win at Chess #250			
9	1	Rc2-c1	85.1	8	0	Nd6-c4	20.8	2	-1	Re3-e7	.094
			>300.	9	0	Nd6-c4	68.2	3	-1	Re3-e7	.196
Win at Chess #238				10	0	Nd6-c4	226.	4	-1	Re3-e7	.349
2	0	Qg3xf4	.017				>300.				

5	0	Re3-e1	1.02	7	3	Rd1xd4	3.98	5	0	Qc2xe4	.141
6	0	Re3-e1	2.60	8	3	Rd1xd4	13.9	6	0	Qc2xe4	.621
7	0	Re3-e1	4.07					7	0	Qc2xe4	1.59
31	62	Re3-e8	9.83	Win at Chess #258				8	0	Qc2xe4	4.07
Win at Chess #251				2	0	Rg3-g5	.012	9	0	Qc2xe4	12.4
2	0	Qd4-h8	.015	3	0	Rg3-g5	.117	10	0	Qc2xe4	34.8
3	0	Qd4-h8	.051	4	0	h2-h3	.357	11	0	Qc2xe4	118.
4	0	Qd4-h8	.167	5	1	Bh5xg6	.622	>300.			
5	0	Qd4-h8	.536	Win at Chess #259				Win at Chess #266			
6	0	Qd4-h8	1.44	2	-2	Ng5xf7	.043	2	0	Bf2-g3	.043
7	0	Qd4-h8	4.67	3	-1	Qd1-h5	.149	3	0	Bf2-g3	.104
8	2	Qd4-e5	11.5	4	-2	Qd1-h5	.406	4	0	Bf2-g3	.282
9	8	Qd4-e5	23.9	5	-2	Qd1-h5	.946	5	0	Bf2-g3	.828
Win at Chess #252				Win at Chess #260				31	63	Rh8xh2	2.04
2	2	Qa2-a1	.015	2	0	Nc7-e6	.024	Win at Chess #267			
3	3	Nf6xd5	.079	3	-1	Nc7-e6	.329	2	0	a2-a3	.067
4	3	Nf6xd5	.302	4	-1	Nc7-e6	.847	3	4	Nd5-c7	.118
5	3	Nf6xd5	1.46	31	63	Qd5-e6	1.50	4	13	Nd5-c7	.258
6	3	Nf6xd5	4.52	Win at Chess #261				31	63	Nd5-c7	.274
7	3	Nf6xd5	14.3	2	2	Qg3xg2	.017	Win at Chess #268			
8	4	Re8-e2	72.2	3	2	Qg3xg2	.056	2	-1	Re2-e3	.027
Win at Chess #253				4	2	Qg3xg2	.205	3	4	Re2-e8	.066
2	0	c6-c7	.009	5	2	Qg3xg2	.681	4	4	Re2-e8	.156
3	0	c6-c7	.044	6	2	Qg3xg2	1.93	5	5	Re2-e8	.381
31	65	Qe3-e8	.089	7	2	Qg3xg2	5.29	6	7	Re2-e8	1.18
Win at Chess #254				8	7	Nf3-d4	17.4	Win at Chess #269			
2	0	Rg4-g5	.013	Win at Chess #262				2	0	Bf4-g3	.011
3	1	Nf4xh3	.037	2	0	Bc6-e8	.009	3	0	Bf4-g3	.058
4	1	Nf4xh3	.102	3	0	Bc6-e8	.048	4	0	Bf4-g3	.393
5	2	Nf4xh3	.228	4	0	Bc6-e8	.293	5	0	Bf4-g3	.983
6	3	Nf4xh3	.836	5	0	Bc6-e8	.727	6	0	Bf4-g3	2.41
Win at Chess #255				6	0	Bc6-e8	2.14	7	0	Bf4-g3	9.14
2	-4	Rf6xg6	.074	7	0	Bc6-e8	10.3	8	0	Bf4-g3	22.6
3	-4	Rf6xg6	.145	8	0	Bc6-e8	27.9	9	0	Bf4-g3	52.4
4	1	Rf6xg6	.273	9	0	Bc6-e8	86.2	10	0	Bf4-g3	196.
5	1	Rf6xg6	.637	10	0	Bc6-e8	300.	>300.			
Win at Chess #256				Win at Chess #263				Win at Chess #270			
2	0	g2-g3	.013	2	0	Qf7-h5	.034	2	1	Re2xe8	.004
3	0	g2-g3	.038	3	0	Qf7-h5	.098	3	1	Re2xe8	.021
4	0	Rh4-g4	.257	31	62	Qf7-g8	.360	4	1	Re2xe8	.071
5	0	Rh4-g4	.478	Win at Chess #264				5	1	Re2xe8	.183
6	0	Rh4-g4	1.40	2	2	Bf4xh2	.014	6	1	Re2xe8	.702
7	1	Nd4-f5	4.90	3	3	Rd8xd4	.098	7	4	Qd4-g4	3.04
Win at Chess #257				4	3	Rd8xd4	.310	8	1	Re2xe8	11.7
2	0	Ne4-f6	.006	5	3	Rd8xd4	.671	9	1	Re2xe8	19.3
3	0	Ne4-f6	.085	Win at Chess #265				10	1	Re2xe8	38.4
4	0	Ne4-f6	.210	2	0	Qc2xe4	.002	11	1	Re2xe8	81.0
5	0	Ne4-f6	.529	3	0	Qc2xe4	.017	12	1	Re2xe8	207.
6	3	Rd1xd4	1.44	4	0	Qc2xe4	.053	>300.			
				Win at Chess #271							

2 0 g7-g8=Q.002			2 0 Qg2-e2 .021
3 -4 Re6-d6 .101	Win at Chess #278		3 0 Qg2-e2 .062
4 -4 Re6-d6 .160	2 0 e4xf5 .013		4 0 Qg2-e2 .262
5 -4 Re6-d6 .282	3 1 Ng5-f7 .100		31 69 Qg2xg7 .499
6 0 Re6-d6 .546	4 1 Ng5-f7 .274	Win at Chess #286	
	31 60 Bc4-f7 .483	2 0 b7-b6 .014	
Win at Chess #272		3 2 Rd8xd5 .083	
2 0 Be3-f2 .006	Win at Chess #279	4 2 Rd8xd5 .228	
3 0 Be3-f2 .061	2 0 Ne3-g4 .012	5 2 Rd8xd5 .705	
4 0 Be3-f2 .205	3 0 Ne3-g4 .260		
5 0 Be3-f2 .502	4 0 Ne3-g4 .420	Win at Chess #287	
6 0 Be3-f2 2.18	5 1 Ne3xf5 .834	2 0 d4-d5 .033	
7 1 Be3xc5 7.92		3 0 d4-d5 .157	
	Win at Chess #280	4 0 d4-d5 .652	
Win at Chess #273	2 0 Qc2xb1 .005	5 0 Bf7-h5 2.16	
2 1 Ne4-f2 .025	3 0 Qc2xb1 .024	6 0 Bf7-h5 6.02	
3 1 Ne4-f2 .115	4 0 Qc2xb1 .082	7 1 Qd1-h5 23.1	
4 1 Ne4-f2 .325	5 0 Qc2xb1 .247		
5 2 Ne4-g3 .976	6 0 Qc2xb1 .935	Win at Chess #288	
	7 0 Qc2xb1 2.21	2 0 Qd6-c6 .024	
Win at Chess #274	8 0 Qc2xb1 6.34	3 0 Qd6-c6 .061	
2 0 Rg6-g7 .007	9 0 Qc2xb1 18.3	4 -1 Qd6-c6 .293	
3 0 Rg6-g7 .047	10 1 Bf8xa3 79.5	5 -1 Qd6-c6 1.33	
4 0 Rg6-g5 .154	11 2 Bf8xa3 200.	6 2 Nh5-f6 2.51	
5 0 Rg6-g5 .219	>300.	7 3 Nh5-f6 4.36	
6 0 Rg6-g5 .390			
7 1 Rg6-g5 .655	Win at Chess #281	Win at Chess #289	
8 2 Rg6-g5 1.21	2 -2 Rc7-c2 .098	2 0 Qe6-g4 .014	
9 2 Rg6-g5 2.03	3 -2 Rc7-c2 .167	3 0 Qe6-g4 .049	
10 2 Rg6-g5 3.76	4 -2 Rc7-c2 .361	4 0 Qe6-g4 .222	
11 2 Rg6-g5 7.13	5 -1 Rc7xh7 1.02	5 1 Qe6-e5 .738	
12 1 Rg6-b6 21.5			
	Win at Chess #282	Win at Chess #290	
Win at Chess #275	2 0 Rh1-h8 .027	2 0 Ne5-g6 .012	
2 0 Nf4-g6 .008	3 0 Rh1-h8 .042	3 0 Ne5-g6 .070	
3 0 Nf4-g6 .171	4 0 Rh1-h8 .121	31 63 Nf4-e2 .217	
4 0 Nf4-g6 .427	5 0 Rh1-h8 .335		
5 0 Nf4-g6 .863	31 65 Rh1-h8 .572	Win at Chess #291	
6 0 Nf4-g6 5.47		2 1 Re5xd5 .002	
7 0 Nf4-g6 12.5	Win at Chess #283	3 1 Re5xd5 .050	
8 0 Nf4-g6 26.4	2 1 Re3xb3 .017	4 1 Re5xd5 .154	
9 1 b7-b5 80.9	3 1 Re3xb3 .069	5 0 Re5-e7 .904	
10 1 b7-b5 182.	4 1 Re3xb3 .279	6 1 Kg1-h1 3.01	
>300.	5 1 Re3xb3 .734	7 1 Kg1-h1 4.79	
	6 1 Re3xb3 1.96	8 1 Kg1-h1 7.93	
Win at Chess #276	7 1 Re3xb3 7.94	9 1 Kg1-h1 22.8	
2 -3 Re7xg7 .080	31 69 Nh3-g5 38.8	10 1 Kg1-h1 68.4	
3 -3 Re7xg7 .155		11 3 h2-h3 207.	
4 5 Qd1-d5 .380	Win at Chess #284	>300.	
5 10 Qd1-d5 .954	2 -2 Qd2-e2 .043	Win at Chess #292	
	3 -2 Qd2-e2 .087	2 0 Qb7xd7 .016	
Win at Chess #277	4 -2 Nd4-f5 .164	3 0 Qb7xd7 .055	
2 0 Ba6xd3 .007	5 -2 Nd4-f5 .445	4 2 d5-d6 .271	
3 0 Ba6xd3 .071	6 -2 Nd4-f5 1.17	5 4 d5-d6 .646	
4 0 Ba6xd3 .311			
5 1 Rg8xg2 1.26	Win at Chess #285		

Win at Chess #293	4	1	g5-g6	.213
2 0 a4xb5	5	1	g5-g6	.443
3 0 a4xb5	6	1	g5-g6	1.15
4 0 a4xb5				
5 0 a4xb5				
6 0 a4xb5				
7 0 a4xb5				
8 0 a4xb5				
9 0 a4xb5				
				>300.

Win at Chess #294	2	0	Qg4-f4	.023
3 0 Qg4-f4	3	0	Qg4-f4	.060
4 0 Qg4-f4	4	0	Qg4-f4	.201
5 0 Qg4-f4	5	0	Qg4-f4	.589
6 1 Qg4-f4	6	1	Qg4-f4	1.81
7 1 Qg4-f4	7	1	Qg4-f4	4.91
8 6 Bd6-f8	8	6	Bd6-f8	15.5
9 6 Bd6-f8	9	6	Bd6-f8	30.6

Win at Chess #295	2	0	Nf4xd5	.006
31 65 Rd1xd5	31	65	Rd1xd5	.014

Win at Chess #296	2	-1	Bc6-d7	.112
3 -1 Bc6-d7	3	-1	Bc6-d7	.266
4 -1 Bc6-d7	4	-1	Bc6-d7	.806

Win at Chess #297	2	0	Bd6-f4	.007
3 0 Bd6-f4	3	0	Bd6-f4	.062
4 0 Bd6-f4	4	0	Bd6-f4	.214
5 0 Bd6-f4	5	0	Bd6-f4	.627
6 0 Bd6-f4	6	0	Bd6-f4	2.35
7 0 Bd6-f4	7	0	Bd6-f4	7.95
8 0 Bd6-f4	8	0	Bd6-f4	22.0
9 0 Bd6-f4	9	0	Bd6-f4	79.9
10 0 Bd6-f4	10	0	Bd6-f4	258.

Win at Chess #298	2	0	Qd8-e8	.009
3 0 Qd8-e8	3	0	Qd8-e8	.026
31 62 Qd8-h8	31	62	Qd8-h8	.149

Win at Chess #299	2	0	Nc5-d7	.013
3 0 Nc5-d7	3	0	Nc5-d7	.051
4 0 Nc5-d7	4	0	Nc5-d7	.316
5 4 Nc5-a4	5	4	Nc5-a4	1.03

Win at Chess #300	2	0	f6xg7	.009
3 0 f6xg7	3	0	f6xg7	.045