Tiny Basic 版

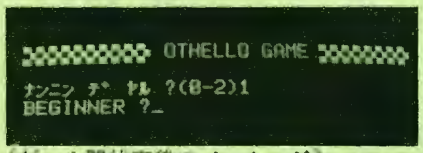
The Othello !

・・・PART-I

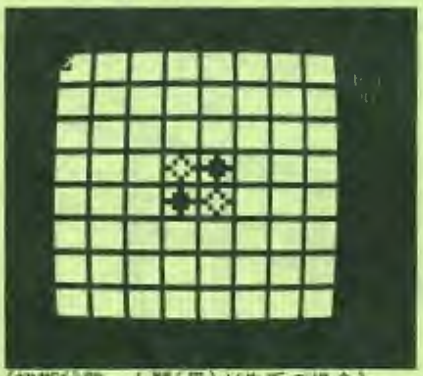
乾 謙一

『オセロ』をプログラミングするのには様々なアルゴリズムが考えられますが，ここに紹介するナカモズTiny Basic版の『オセロ』は，後述の“必勝手順探索プログラム”に木探索を用いている点で，一味違った趣きを持っています．

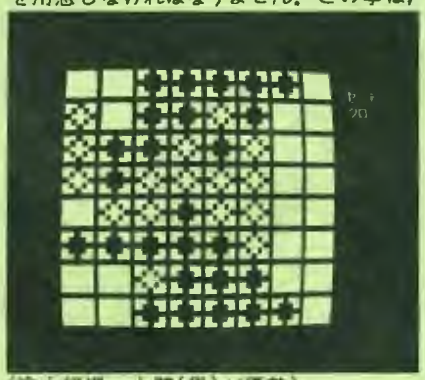
本誌には度々御登場頂いている乾謙一氏の手によるものですが，今後拡張されることも十分期待されます．あんまり急いで読むと消化不良を起こす恐れがありますので，じっくりと味わってみて下さい．(編集部)



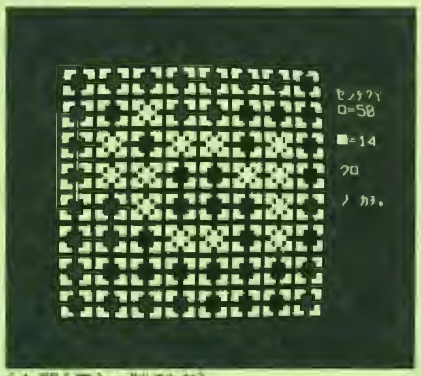
〈ゲーム開始直後のメッセージ〉



〈初期状態；人間(黒)が先手の場合〉



〈途中経過；人間(黒)が優勢〉



〈人間(黒)の勝利!?〉

ナカモズTiny Basicのアプリケーションとして，オセロを製作しましたので紹介させて頂きます．

このオセロを製作するにあたって，「コンピュータが1手に要する時間は，何秒を限度とするか？」という事を考えました．これが極端に長いと，いくらコンピュータが強くても対戦者が興味を失ってしまいます．これは意見の分かれるところですが，5秒，あるいは10秒というところでしょう．ちなみに，8月号のマイクロオセロは30秒（クロック1MHz），Apple IIのアプリケーションも試してみましたが，マイクロオセロと同じ位で約45秒でした．

アルゴリズム

次にアルゴリズムですが，先を読まなければ人間の相手にはなりませんから，取り敢えず1手先を読むことにします．オセロでは，1手先といっても石を置けば盤が変化しますから，もう1つ別の盤を用意しなければなりません．この事は，当初かなり困難であろうと予測していたのですが，オセロでは，石の置ける位置が限られていますから（せいぜい20ヵ所位），割合に簡単な処理でできることがわかりました．

しかし，いくら先を読むといっても最後まで読むわけではありませんから，ゲーム途中で有利不利を判断する評価関数を作らなければなりません．これには，まず機械語サブルーチン（ラベル“ＴＡＢＬＥ”）で，全ての位置についてそこに石を置いた時に裏返せる相手の石の数を調べて，メモリ内にテーブルを作ります．

次に，オセロではコーナーが有利である等，位置によって有利不利があります

88	16	38	33	33	38	16	88
16	0	18	18	18	18	0	16
38	18	28	23	23	28	18	38
33	18	23			23	18	33
33	18	23			23	18	33
38	18	28	23	23	28	18	38
16	0	18	18	18	18	0	16
88	16	38	33	33	38	16	88

表2 ペナルティ・データ・テーブル

から，このテーブルにペナルティを加えます．これは，機械語“ＰＥＮＡＬ”によって行なわれます．コンピュータは，自分のメモリ内に1手先の盤を実現させて，

（今回の自分の石の点数）

ー（次の相手の点数の最大値）を評価値として最良手を選びます．ただし，相手にパスさせる様な手にはプラス5点としています．

以上の手は主に中盤に有効ですから，序盤ではテーブルを逆に使って，相手に石を多く取らせます．終盤ではペナルティを加えずに，1手先で少しでも自分の石が多くなる様に打ちます．

マイクロオセロとの比較

1) 表示はグラフィックで，座標を考える必要がない．
2) 人間対人間の場合でも，パスを自動的に判断する．
3) パターンは，毎回変化する．
4) コンピュータに，学習させる事ができる．

コンピュータの強さに関して言えば，

5) 必要ならばコーナーの隣でも打つが，コーナーを取られそうになれば積極的に守る．
6) 相手にパスさせる様に打つ．

プログラムの説明

100〜 170 サブルーチン REVERSE
900〜1090 コンピュータルーチン
3000〜3040 イニシャライズ
4000〜4380 メインプログラム
4500〜4540 サブルーチン DISPLAY
5000〜5080 データ

マシン語サブルーチン（$2000〜$20D0）
$2000 TRANS
$20D0 PENAL
$203A TABLE

コンピュータルーチンについて，もう少し詳しく説明してみましょう．

910〜 990 主に序盤で用いる所です．
940〜 950 石の数を無視して，ペナルティのみを用いています．
960〜 970 むしろ石の数の少ないところに打ちます．

どちらを使うかは乱数で決めています．

1000〜1090 中盤以降で用いる所です．

自分の石が打てる座標と，その点数を一担スタックにセーブした後，機械語サブルーチン“TRANS”で盤1の内容を盤2に転送します．次にスタックから1個取り出して盤2に石を置き，100〜170のサブルーチンで石を裏返してから，相手の石の点数を計算して評価値を出します．

これを繰り返して評価値最大の所に打ちます．表1に変数と配列（1バイト単位）

REVERSE

変数		配列	
@	データ アドレス	@	ペナルティ・データ・テーブル
A	石の種類	↓	
B	石の数	↓	
C	人数	@+63	
D	黒の石の数		
E	白の石の数	Z	盤1
F	X，Yの一時記憶	↓	
G		↓	
H	座標の一時記憶	Z+63	
K	初心者のフラグ		
L	ループカウンター他	Z+64	評価値テーブル
M		↓	
N	評価値の最大値	↓	
O	スタック ポインター	Z+127	
P	POKEするアドレスの一時記憶		
Q	Z+128	Q	盤2
R	パスの回数	↓	
U	ユーザー関数の出力	↓	
V	V-RAM アドレス	Q+63	
W	V-RAM上アドレスの一時記憶		
X	座標	Z+192	座標と評価値のスタック（下図参照）
Y		↓	
Z	配列の先頭番号 @+64		

座標	評価値	座標	評価値				

#(Z+192)

↑ STACK POINTER 変数O
⇒PUSH
PULL ←

表1 変数と配列

を示します．

プログラムをタイプインする前に…

システムに合わせて変更が必要となりますので記しておきます．

① グラフィックの出ない場合
適当なキャラクターと入れ換える．

② 配列の番地，V－ＲＡＭの番地
3010　@＝＄１Ｆ00を変更する．
（後に 300バイト程使用します．）
3020　Ｖ＝＄ＣＥ00を変更する．

③ データバスが正論理の場合
4140 …：＃Ｗ＝＄ＦＦ－“＞”を単に　＃Ｗ＝“＞”とする．

④ Ｈ68／ＴＲのコンソール使用の場合
4160，4170，4180の“Ｒ”，“Ｄ”，＄Ｄを適当なキーコードに変更する．

⑤ 機械語をリロケートする場合
ユーザー関数の飛び先を変更する．

…以上の変更で完全に動作します．

遊び方

さっそくプログラムを走らせてみましょう．まず，何人でやるか？と聞いてきます．さらに，ＢＥＧＩＮＮＥＲ？と聞いてきます．ここでＹを入力すれば，コンピュータは弱くなります．１人でプレイする時，人間は黒で，コンピュータは白です．先手後手は自由に選べます．

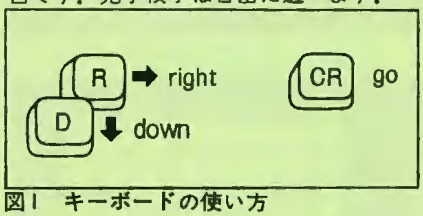


図１　キーボードの使い方

人間の手番では，画面の左上に点滅する＞印が現れます．ここでＲを押すと，マークは右の方へ動いてゆきます．Ｄを押すと下に動きます．(図１参照)最後は，チョンチョンとやれば１マスずつ動きます．もし行き過ぎたら，そのまま押し続けるとまた反対側に戻ります．Ｈ68／ＴＲのコンソールを使用する時は，上下左右に動かせる様にする方が良いでしょう．(CR)を押せば，マークの位置に石がセットされます．もちろん，石をはさんでいなければ，(CR)キーは反応しません．

コンピュータが１手に要する時間は，中盤でも５～６秒以内です．序盤では，もっと速くて１秒位です．強さの方は言葉で説明できませんが，かなり強いとだけ申しておきましょう．(人間の相手としては，適当？)

学習するコンピュータ

このプログラムでは，ペナルティデータテーブル(表２)をＤＡＴＡ文として，Basic側に持っています．これはＲＵＮすると，メモリーの@～@＋63番地に書き込まれます．このテーブルの値によって，コンピュータの手は大きく変化します．特に，コーナーの隣は±１位でも効いてきます．

ですから，あなた自身で試合をやってみてデータの値を改良してみて下さい．(注意：下手な人が相手をすると，段々下手になる？)

また，データ文を何組も用意しておいてＲＥＳＴＯＲＥ文で切り換えても，試合の状況に応じて書き換えても効果的でしょう．

最後に……

とにかく，これだけのプログラムが３Ｋバイト以下にまとまったのも，Ｎ.Ｔ.Ｂ.のおかげです．私自身，「オセロに強い」というわけではありませんから，このプログラムを走らせてみて，「ここはこうした方が良い．」といったアドバイスをしていただければ幸いです．最後に，このプログラムの製作に際して協力して下さった，南憲明さん，山西一啓さんと，Ｎ.Ｔ.Ｂ.を提供して下さった山下春夫さんに感謝いたします．

・・・PART-II
心勝手順探索プログラム！

ここに紹介するプログラムは少し毛色の変わったもので，マイコンに完全なプレイをさせることを目的として作られています．

もちろん実行速度のことを考えれば，このプログラムは終盤でしか使用できませんから，“詰めオセロ”を解くプログラムと考えていただいても結構です．これは，12月号の“騎士巡歴問題”を見て思いついたのですが，普通，この種の思考プログラムに使われる“minimax法”とは少し異なっています．

必勝手順探索のアルゴリズム

このプログラムの特徴は，あらかじめ何手読むのかを決めておくのではなくて，このプログラムに飛んだ時点から最後までを読むというところにあります．図２を見て下さい．自分の手番から始めて４手目に勝負がつく場合です．

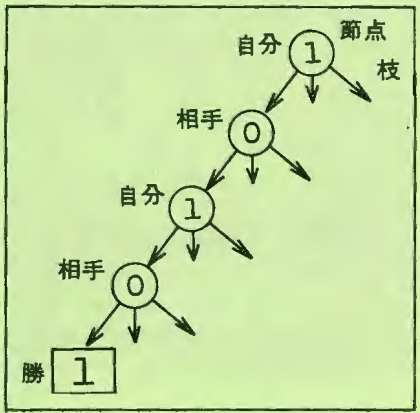


図２　探索のモデル

実際の試合では，パスのことや，途中で勝負がつく場合のことも考慮しなければなりません．さて，そのアルゴリズムは……

1. 自分の石，相手の石を交互に置きながら勝負がつくまで石を埋める．その際，自分の手番の節点には1，相手の手番の節点には0のマークを付けておく．
2. 勝負が自分の勝ちなら1，負けなら0をback up値として，一番近い節点まで後退する．
3. ここで，もしback up値と節点マークが一致したら，さらに後退する．一致しなければ，ステップ1に戻って他の枝を調べる．
4. すべての枝を調べたら，一致しなくても一番近い節点まで後退して，ステップ3に戻って繰り返す．
5. 以上の操作を終了して，back up値が1なら必勝手順が存在する．

図3ⓘ～ⓥにその例を示します．

ⓘ まず，勝負がつくまで前進してback up値1を得る．

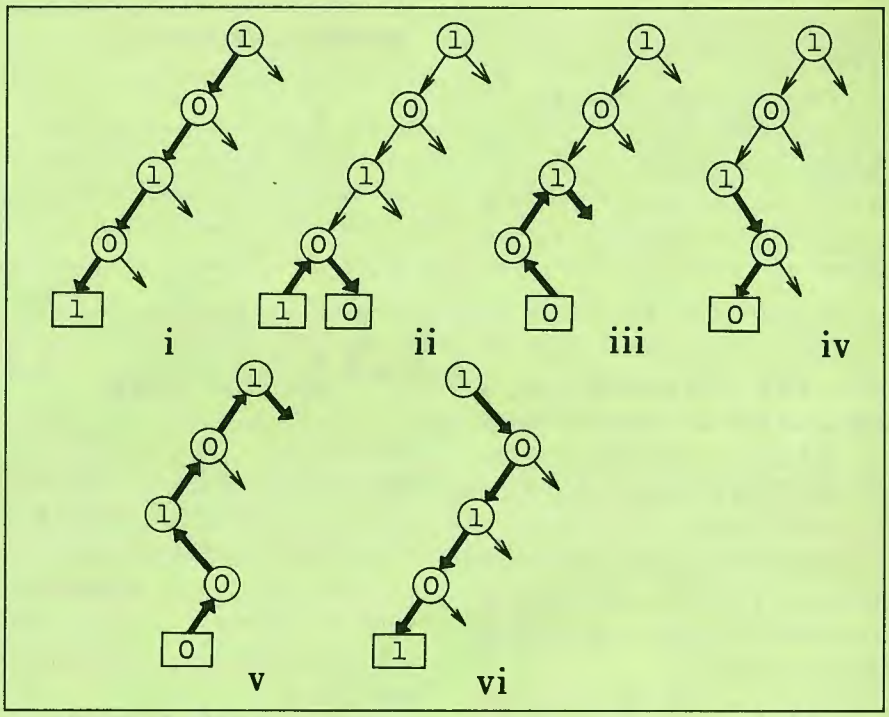
ⓘⓘ 後退する．一致しないので前進して0を得る．

ⓘⓘⓘ 後退する．一致するのでもう1歩後退する．一致しないので前進する．

ⓘⓥ ここでback up値0を得る．

ⓥ 一致するので後退する．一致しないが枝がないので後退，さらに後退する．

ⓥⓘ 結局，この枝には必勝法はなかったので，別の枝を探索する．



プログラムの実際

ここでは，私の“Tinyオセロ”に接続した場合のプログラムを例にとって説明してみましょう．

その前に，Basicで書かれていたサブルーチンＲＥＶＥＲＳＥをマシン語に直しましたので，そのソースリストを図4に示しておきます．アセンブラを御使いの方は，“Tinyオセロ”のプログラムの後に接続してアセンブルしていただければＯ．Ｋ．

ADDRESS	LABEL	MNEMONIC	A.M.	OPERAND	COMMENTS
20F6	REVERS	LDAB	#	$FB	ベクトル DY
		STAB		DY	イニシャライズ
	RE1	LDAB	#	$FF	ベクトル DX
		STAB		DX	イニシャライズ
	RE2	PSHA			Aには座標が入っている．
		ANDA	#	$7	
		STAA		WX	
		PULA			(X,Y)に分解
		PSHA			する．
		ANDA	#	$38	
		STAA		WY	
		CLRB			カウンタークリア
	RE3	LDAA		DX	
		ADDA		WX	WX←WX+DX
		STAA		WX	
		BITA	#	$8	X方向はみ出し．
		BNE		RE5	
		LDAA		DY	
		ADDA		WY	WY←WY+DY
		STAA		WY	
		BITA	#	$40	Y方向はみ出し．
		BNE		RE5	
		ADDA		WX	
		LDX		DATA	
		JSR		SUM	盤2を調べる．
		LDAA		192,X	
		BEQ		RE5	石がない．

ADDRESS	LABEL	MNEMONIC	A.M.	OPERAND	COMMENTS
		INCB			
		CMPA		VA	
		BNE		RE3	相手の石がある．
	RE4	LDAA		WX	
		SUBA		DX	
		STAA		WX	
		LDAA		WY	
		SUBA		DY	
		STAA		WY	バックしながら
		ADDA		WX	石を裏返す．
		LDX		DATA	
		JSR		SUM	
		LDAA		VA	
		STAA		192,X	
		DECB			
		BNE		RE4	
	RE5	PULA			
		LDAB		DX	
		INCB			ベクトルDXを
		STAB		DX	インクリメント
		CMPB	#	$2	
		BNE		RE2	
		LDAB		DY	
		ADDB	#	$8	ベクトルDYを
		STAB		DY	インクリメント
		CMPB	#	$10	
		BNE		RE1	
		RTS			

図4 サブルーチン REVERSE ソースリスト

です．オブジェクトリストは，＄2000～＄215Cにアセンブルしてあります．

この改造とBasic側の若干の改良によって，これまで約5秒程かかっていた“1手先読み”の思考時間が，約1秒になりました．(この1手先読みは，自分の手番から数えれば3手先読みだという説もある．)

さて，オセロでは，置いた石を取り除いただけでは元の盤に戻ってくれませんから，各節点での盤を全部残しておかなければなりませんが，それではメモリを食い過ぎるので，前進する際に毎回最初から盤を作り直す様にしました．(サブルーチンＳＥＴＵＰ)

各節点での可能な手は，スタックに格納されています．スタックは，配列の最後に 256バイトを使用して図5の様に構成されています．

最大14バイトまでのスタックを16個持っていますから，一応16手までは安全に読めると思います．ここで，ⓐには変数Ａの値，つまり，黒の手番なら1，白の手番なら2が書き込まれます．ⓑには，

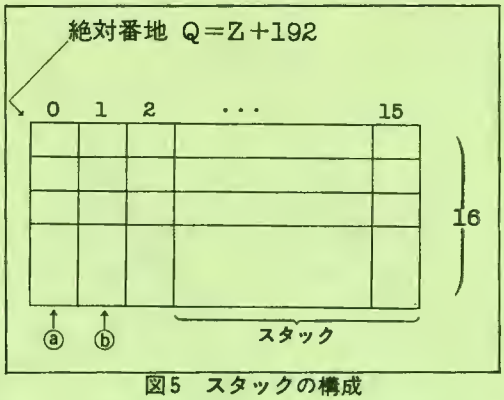


図5　スタックの構成

それ以後のスタックがどこまで有効か，つまりポインターが入ります．ⓑに2が入っていれば，そのスタックは空です．

フローチャートを図6に示します．サブルーチンＳＥＴＵＰは，最初機械語で製作していたのですが，メモリ不足のためアセンブルできなくなったのでBasicに変更しました．

このプログラムは，変数Ｔの値によって制御されています．つまり，Ｔ＝2の時前進操作を行ない，後退する時はＴにback up値を入れておきます．

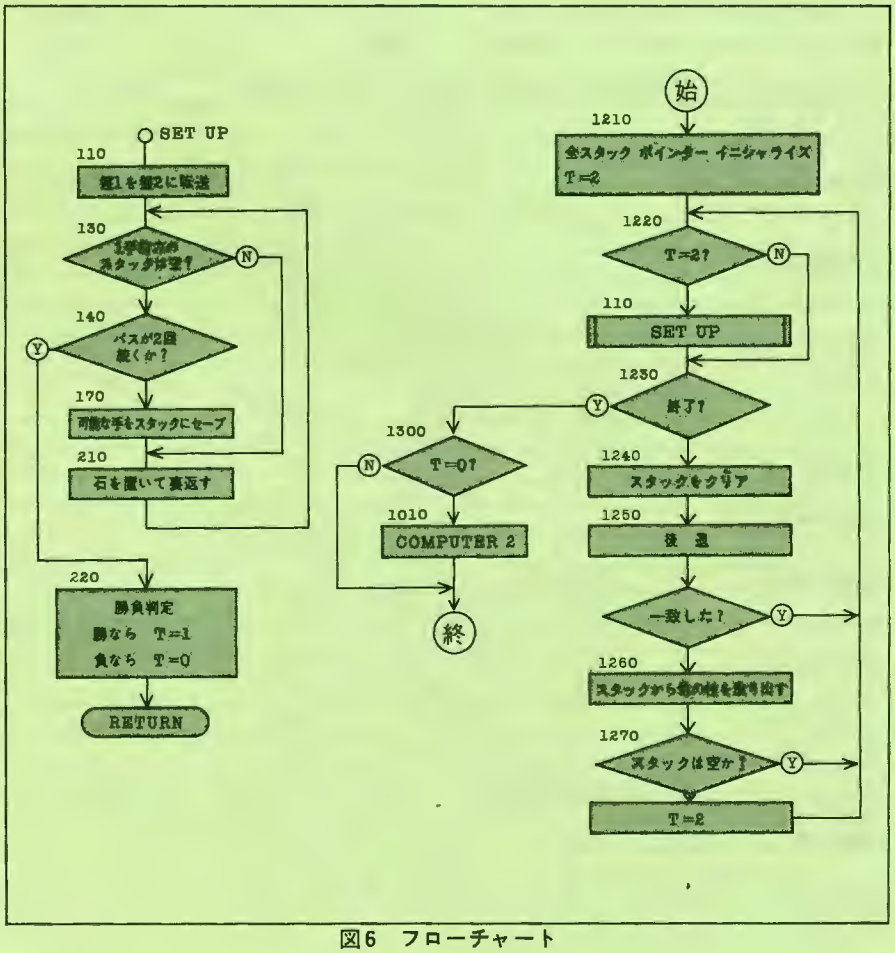


図6　フローチャート

実行例とその結果

このプログラムで8手読むのに，平均1分位かかります．というのは，このアルゴリズムでは，どの程度調べる枝を省略できるかを運に頼るしかなく，速い時は20秒位ですし，3分かかったという報告もあるのです．

また，このプログラムは必勝手順を見つけるものの，それは最良手順とは限りません．

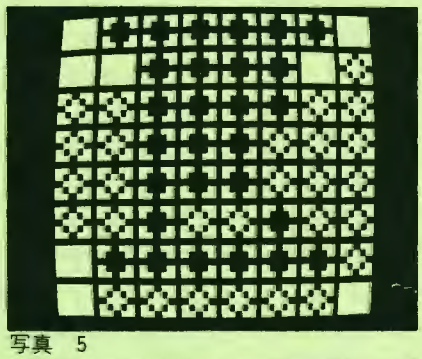


写真　5

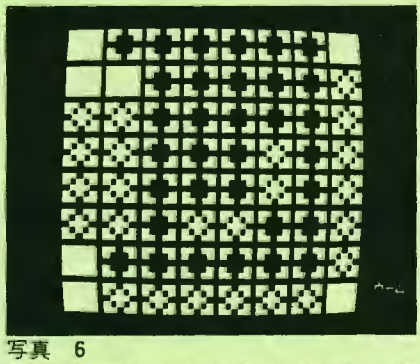


写真　6

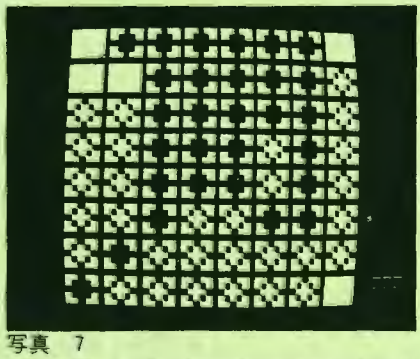


写真　7

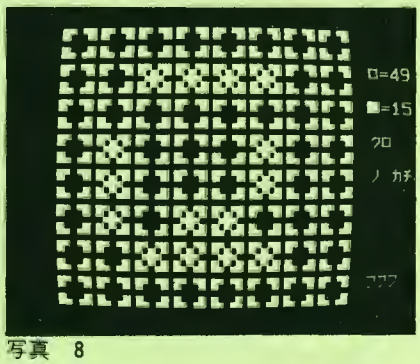


写真　8

必勝手順が見つからない時は何の情報も得られないので，１手先読みのプログラムに飛ばして相手のミスを待ちます．

写真は，コンピュータ同士で試合をさせたところです．写真１は黒の手番です．ここまでは，１手約１秒でスピーディーに試合を進めてきますが，行番号 920によって53手目から黒の８手読みに移ります．

ここですでに黒は必勝手順を見つけてしまうので(写真２)，白が７手読みをしても，もう必勝手順は見つかりません．黒はゆうゆうと試合を進めていきます．(写真３，４)

もちろん最後は黒が勝ちます．これなら，10手以上の難かしい詰めオセロでも，時間さえかければ解くことができます．

是非一度実験してみて下さい．

LISTING TABLE

Part-I Tiny Othello

```
)L.
10 REM**********************
11 REM* TINY OTHELLO GAME  *
12 REM*       V.3.1        *
13 REM* BY KEN'ICHI-INUI   *
14 REM**********************
20 G.3010
100 REM REVERSE
110 F=X:G=Y:F.L=-1TO1:F.M=-1TO1
120 IF(L=0)+(M=0)=2 N.M
130 X=F:Y=G:DO:X=X+L:Y=Y+M:IF(X)7)+(X(0)+(Y)7)+(
Y(0) U.1:G.170
140 P=0+X+8*Y
150 IF@P=0 U.1:G.170
160 U.@P=A:DO:X=X-L:Y=Y-M:@(@+X+8*Y)=A:U.(X=F)+(
Y=G)=2
170 N.M:N.L:R.
900 REM COMPUTER
910 IFU.($203A,1)=0 R=R+1:G.4300
920 IF(8)14)+(K=0)=2T.1020
930 IF R.(2)=0 T.960
940 F.L=64TO127:IF@(Z+L))0 @(Z+L)=1
950 N.L:G.900
960 F.L=64TO127:IF@(Z+L))0 @(Z+L)=20-@(Z+L)
970 N.L
980 U=U.($200D):L=63:DO:L=L+1:U.@(Z+L)=U
990 Y=(L-64)/8:X=MOD:G.4190
1000 REM FINAL STAGE
1020 IFB(55 U=U.($200D)
1030 O=192:F.L=0TO63
1040 IF@(Z+L+64))0 @(Z+0)=L:@(Z+0+1)=@(Z+L+64):D
=0+2
1050 N.L:IF0=194 Y=@(Z+192)/8:X=MOD:G.4190
1060 N=-99:DO:O=0-2:U=U.(@200D):Y=@(Z+0)/8:X=MOD

1070 GOS.110:A=3-A:U=U.($203A,2):IFB(55 U=U.($20
0D)
1075 IFU=0 U=-5
1080 A=3-A:IFN(@(Z+0+1)-U N=@(Z+0):N=@(Z+0+1)-U
1090 U.0=192:Y=N/8:X=MOD:G.4190
3000 REM INISIAL
3010 @=$1E00
3020 V=$CE00
3030 Z=@+64:Q=Z+120
3040 F.L=@ TO@+63:@L=RE.+R.(2):N.L
4000 REM MAIN PROG.
4010 P.C.5;"%%%%%%%%% OTHELLO GAME %%%%%%%%"
4020 R=0
4030 F.L=0TO63:@(Z+L)=0:N.L
4035 @(Z+27)=2:@(Z+28)=1:@(Z+35)=1:@(Z+36)=2
4040 P."ｶﾝｺｳ ﾃﾞ ﾅﾑ ?(0-2)";:D=G.-48:IFC)2T.4040
4045 P.:IFC(2 P."BEGINNER ?";:K=C.="Y"
4050 C.:CU.(0,0):F.L=0TO7
4060 P."■■■■■■■■■■■■■■■■"
4070 P."■■■■■■■■■■■■■■■■";:IFL(>7P.
4080 N.L:A=1:X=4:Y=3:GOS.4500:X=3:Y=4:GOS.4500
4090 A=2:X=3:Y=3:GOS.4500:X=4:Y=4:GOS.4500:B=0
4095 IFC=1 CU.(25,1):P."ｾﾝｶﾃ?";:A=(G.="Y")+1
4100 A=3-A
4110 IFU.($203A,1)=0 R=R+1:G.4300
4120 R=0:B=B+1:IF(C=0)+(C=1)*(A=2) T.910
4130 X=0:Y=0:CU.(25,2):P.C.(A=1)*"ｱ"+(A=2)*"ｼﾛ";"
ﾛ";
4140 DO:V=V+3*X+64*Y:L=@V:@V=$FF-")"
4150 F.M=(M=180)*150TO160:N.M
4155 @V=L:M=KEY
4160 IFM="R" X=X+1:IFX)7 X=0
4170 IFM="D" Y=Y+1:IFY)7 Y=0
4180 U.(M=$D)+(@(Z+64+X+8*Y))0)=2
4190 @(Z+X+8*Y)=A:GOS.4500:F=X:G=Y:F.L=-1TO1:F.M
=-1TO1
4200 IFOR(L,M)=0T.4260
4210 X=F:Y=G:DO:X=X+L:Y=Y+M:IF(X)7)+(X(0)+(Y)7)+
(Y(0) U.1:G.4260
4220 P=Z+X+8*Y
4230 1F0=8P U.1:G.4260
4240 U.A=0P:DO:X=X-L:Y=Y-M:IF(X=F)*(Y=G) U.1:G.4
260
4250 @(Z+X+8*Y)=A:GOS.4500:U.0
4260 N.M:N.L
4270 D=0:E=0:F.L=0TO63:M=@(Z+L):D=D+(M=1)
4280 E=E+(M=2):N.L
4290 1F(D=0)+(E=0)+(D+E=64)T.4320
4300 1FR)1T.4320
4310 G.4180
4320 CU.(25,2):P."□=";D;
4330 CU.(25,4):P."■=";E;
4335 IFD=E T.4360
4340 CU.(25,6):P.C.(D)E)*"ｱ"+(D<E)*"ｼﾛ";"□";
4350 CU.(25,8):P."ﾉｺﾘ ﾏｽ=";:G.4370
4360 CU.(25,6):P."ﾋｷｱﾜｹ";
4370 IFG.="Y" G.4010
4380 END
4500 P.C.6:C.7;:IFA=2T.4530
4510 CU.(3*X,2+Y):  P."■";
4520 CU.(3*X,2+Y+1):P."▙▟";:R.
4530 CU.(3*X,2+Y):  P."▛▜";
4540 CU.(3*X,2+Y+1):P."▙▟";:R.
5000 REM DATA
5010 *88,16,38,33,33,38,16,88
5020 *16,0,18,18,18,18,0,16
5030 *38,18,28,23,23,28,18,38
5040 *33,18,23,0,0,23,18,33
5050 *33,18,23,0,0,23,18,33
5060 *38,18,28,23,23,28,18,38
5070 *16,0,18,18,18,18,0,16
5080 *88,16,38,33,33,38,16,88

READY
)

 78.5     RAM68 R1.0

    0001                *
    0002                *
    0003                *
    0004                *********************
    0005                *SUBROUTINE FOR
    0006                * TINY OTHERO GAME
    0007                *PROGRAMED BY KENY
    0008                *1978/11/26
    0009                *********************
    0010 0003   VA      EQU   $3
    0011 0000   DATA    EQU   $0
    0012                *
    0013                *
    0014 003C           ORG   $3C
    0015 003C   0001    COUW  RMB   1
    0016 003D   0001    MAX   RMB   1
    0017 003E   0002    IXS   RMB   2
    0018 0040   0002    WK1   RMB   2
    0019 0042   0002    WK2   RMB   2
    0020 0044   0001    DX    RMB   1
    0021 0045   0001    DY    RMB   1
    0022 0046   0001    WX    RMB   1
    0023 0047   0001    WY    RMB   1
    0024                *
    0025        2000    ORG   $2000
    0026                *
    0027                **** TRANS ****
    0028                *
    0029 2000 DE 00     TRANS LDX   DATA
    0030 2002 86 40           LDA A  #64
    0031 2004 E6 40     TR1   LDA B  64,X
    0032 2006 E7 C0           STA B  192,X
    0033 2008 08              INX
    0034 2009 4A              DEC A
    0035 200A 26 F8           BNE   TR1
    0036 200C 39              RTS
    0037                *
    0038                **** PENAL ****
    0039                *
    0040 200D DE 00     PENAL LDX   DATA
    0041 200F 86 40           LDA A  #64
    0042 2011 7F 003D         CLR   MAX
    0043 2014 E6 00     PE2   LDA B  128,X
    0044 2016 27 00.          BEQ   PE1
    0045 2018 EB 00           ADD B  0,X
    0046 201A E7 00           STA B  128,X
    0047 201C D1 3D           CMP B  MAX
    0048 201E 23 00.          BLS   PE1
    0049 2020 D7 3D           STA B  MAX
    0050 2022 08        PE1   INX
         2017 0A
         201F 02
    0051 2023 4A              DEC A
    0052 2024 26 EE           BNE   PE2
    0053 2026 5F              CLR B
    0054 2027 96 3D           LDA A  MAX
    0055 2029 39              RTS
    0056                *
    0057                **** SUM ****
    0058                *
    0059 202A 36        SUM   PSH A
    0060 202B DF 3E           STX   IXS
    0061 202D 9B 3F           ADD A  IXS+1
    0062 202F 24 03           BCC   ++5
    0063 2031 7C 003E         INC   IXS
    0064 2034 97 3F           STA A  IXS+1
    0065 2036 DE 3E           LDX   IXS
    0066 2038 32              PUL A
    0067 2039 39              RTS
    0068                *
    0069                *
    0070                **** TABLE ****
    0071                *
    0072 203A 8C 0001   TABLE CPX   #1
    0073 203D 26 00.          BNE   TABLE2
    0074 203F B6 40           LDA A  #64
    0075 2041 20 00.          BRA   TA1
    0076 2043 86 C0     TABLE2 LDA A  #192
         203E 04
    0077 2045 DE 00     TA1   LDX   DATA
         2042 02
    0078 2047 BD E1           BSR   SUM
    0079 2049 DF 40           STX   WK1
    0080 204B DE 00           LDX   DATA
    0081 204D B6 00           LDA A  #128
    0082 204F 8D D9           BSR   SUM
    0083 2051 DF 42           STX   WK2
    0084 2053 7F 003D         CLR   MAX
    0085 2056 4F              CLR A
    0086                *
    0087 2057 DE 40     TA2   LDX   WK1
    0088 2059 8D CF           BSR   SUM
    0089 205B 6D 00           TST   X
    0090 205D 27 00.          BEQ   TA3
    0091 205F 5F              CLR B
    0092 2060 20 00.          BRA   TA4
    0093 2062 7F 003C   TA3   CLR   COUW
         205E 03
    0094 2065 C6 F8           LDA B  #$F8
    0095 2067 D7 45           STA B  DY
```

```
0096 2069 C6 FF     TA8   LDA B  #$FF
0097 206B D7 44           STA B  DX
0098 206D 36        TA6   PSH A
0099 206E 84 07           AND A  #$07
0100 2070 97 46           STA A  WX
0101 2072 32              PUL A
0102 2073 36              PSH A
0103 2074 84 38           AND A  #$38
0104 2076 97 47           STA A  WY
0105 2078 5F              CLR B
0106 2079 96 44     TA5   LDA A  DX
0107 207B 9B 46           ADD A  WX
0108 207D 97 46           STA A  WX
0109 207F B5 00           BIT A  #$00
0110 2081 26 00.          BNE    TA9
0111 2083 96 45           LDA A  DY
0112 2085 9B 47           ADD A  WY
0113 2087 97 47           STA A  WY
0114 2089 B5 40           BIT A  #$40
0115 208B 26 00.          BNE    TA9
0116 208D 9B 46           ADD A  WX
0117 208F DE 40           LDX    WK1
0118 2091 8D 97           BSR    SUM
0119 2093 6D 00           TST    X
0120 2095 27 00.          BEQ    TA9
0121 2097 96 03           LDA A  VA
0122 2099 A1 00           CMP A  X
0123 209B 27 00.          BEQ    TA7
0124 209D 5C              INC B
0125 209E 20 D9           BRA    TA5
0126 20A0 5F        TA9   CLR B
     20A2 1D
     20A3 13
     20A5 09
0127 20A6 32        TA7   PUL A
     20A7 84
0128 20A8 DB 3C           ADD B  COUN
0129 20AA D7 3C           STA B  COUN
0130 20AC D6 44           LDA B  DX
0131 20AE 5C              INC B
0132 20AF D7 44           STA B  DX
0133 20B1 C1 02           CMP B  #$02
0134 20B3 26 BE           BNE    TA6
0135 20B5 D6 45           LDA B  DY
0136 20B7 CB 00           ADD B  #$00
0137 20B9 D7 45           STA B  DY
0138 20BB C1 10           CMP B  #$10
0139 20BD 26 00           BNE    TA0
0140 20BF D6 3C           LDA B  COUN
0141 20C1 D1 3D           CMP B  MAX
0142 20C3 23 00.          BLS    TA4
0143 20C5 D7 3D           STA B  MAX
0144 20C7 DE 42     TA4   LDX    WK2
     20C9 5F
     20CA 02
0145 20CB 8D 20ZA         JSR    SUM
0146 20CD E7 00           STA B  X
0147 20CF 4C              INC A
0148 20D0 81 40           CMP A  #64
0149 20D2 26 0A           BNE    TA2
0150 20D4 5F              CLR B
0151 20D5 96 3D           LDA A  MAX
0152 20D7 39              RTS
0153                *
0154                *
0155 20D8                 END

VA     0003    DATA   0000    COUN   003C
MAX    003D    IXS    003E    WK1    0040
WK2    0042    DX     0044    DY     0045
WX     0046    WY     0047    TRANS  2000
TA1    2004    PENAL  200D    PE2    2014
PE1    2022    SUM    202A    TABLE  203A
TABLE2 2043    TA1    2045    TA2    2057
TA3    2062    TA4    20C1    TA0    2069
TA6    206D    TA5    2079    TA9    20A0
TA7    20A1


TOTAL ERROR 00

78.5     RAM68 R1.0

78.5     RAM68 R1.0
```

Part-II　必勝手順探索ルーチンつき　Othello

```
)
)
READY
)L.
10 REM**********************
11 REM* TINY OTHELLO GAME  *
12 REM*       V.4.1        *
13 REM* BY KEN'ICHI-INUI   *
14 REM**********************
20 G.3010
100 REM#### SET UP ####
110 U=U.($2000)
120 J=0:D=A:A=3-A
130 IF@(J+1))2 T.200
140 DO:A=3-A:R=R+1
150 IFR)2 U.!:G.220
160 U.U.($2006,2):R=0
170 &J=A:I=2:F.L=Z+64TOZ+127
180 IF@L @(J+I)=L-Z-64:I=I+1
190 N.L:@(J+1)=I:G.210
200 R=&J:I=@(J+1)
210 U=U.($2009,0,@(J+1-1)):J=J+16:G.130
220 T=0:R=0:A=D:F.L=Z+128 TOZ+191
230 T=T+(&L=A)-(&L=3-A):N.L
240 T=T)0:R.
300 REM### CHANGE ###
310 P=F+G#8
320 IFP=0 T.360
330 IFP=7 T.370
340 IFP=56T.380
350 IFP=63T.390
355 R.
360 @(Z+1)=33:@(Z+8)=33:@(Z+9)=20:R.
370 @(Z+6)=33:@(Z+14)=20:@(Z+15)=33:R.
380 @(Z+48)=33:@(Z+49)=20:@(Z+57)=33:R.
390 @(Z+54)=20:@(Z+55)=33:@(Z+62)=33:R.
900 REM#### COMPUTER ####
910 IF U.($2006,1)=0 R=R+1:G.4300
920 IF(B)52)+(K=0)=2T.1210
925 IF B)12 T.1020
930 IF R.(2)=0 T.960
940 F.L=64TO127:IF@(Z+L))0 @(Z+L)=1
950 N.L:G.980
960 F.L=64TO127:IF@(Z+L))0 @(Z+L)=20-@(Z+L)
970 N.L
980 U=U.($2003):L=63:DO:L=L+1:U.@(Z+L)=U
990 Y=(L-64)/8:X=MOD:G.4190
1000 REM#### COMPUTER 2 ####
1010 U=U.($2006,1)
1020 IFB(56 U=U.($2003)
1030 D=192:F.L=0TO63
1040 IF@(Z+L+64))0 @(Z+O)=L:@(Z+O+1)=@(Z+L+64):O
=O+2
1050 N.L:IFO=194 Y=@(Z+192)/8:X=MOD:G.4190
1060 M=-999:DO:O=0-2:U=U.($2000):U=U.($2009,0,@(
Z+O))
1070 A=3-A:U=U.($2006,2):IFB(55 U=U.($2003)
1075 IFU=0 U=-6
1080 A=3-A:IFK(@(Z+O+1)-U H=@(Z+O):K=@(Z+O+1)-U
1090 U.O=192:Y=H/8:X=MOD:G.4190
1200 REM#### COMPUTER 3 ####
1210 F.L=0+1 TO 0+256 S.16:&L=2:N.L:T=2
1215 CU.(25,14):P."ｹﾝｻｸ";
1220 IFT=2 GOS.110
1230 IFJ=0 T.1300
1240 @(J+1)=2
1250 J=J-16:IFT=(&J=A) T.1220
1260 @(J+1)=@(J+1)-1
1270 IF@(J+1))2 T=2
1280 G.1220
1300 IFT=0 CU.(25,14):P."   ";:G.1010
1315 CU.(25,14):P."777";
1320 Y=@(0+@(0+1)-1)/8:X=MOD:G.4190
3000 REM INITIAL
3010 0=$1C00
3020 U=$CE00
3030 Z=0+64:0=Z+192:RES.5010
3040 F.L=0 TO0+63:&L=RE.+R.(2):N.L
4000 REM MAIN PROG.
4010 P.C.5;"XXXXXXXXXX OTHELLO GAME XXXXXXXX"
4020 R=0
4030 F.L=0TO63:@(Z+L)=0:N.L
4035 @(Z+27)=2:@(Z+28)=1:@(Z+35)=1:@(Z+36)=2
4040 P."ｶﾝｺｳ ﾃﾞｽ ﾈ ｱｲﾃ(0-2)";:C=G.-48:IFC)2T.4040
4045 P.:IFC(2 P."BEGINNER ?";:K=G."Y"
4050 C.:CU.(0,0):F.L=0TO7
4060 P."■■■■■■■■■■"
4070 P."■■■■■■■■■■";:IFL()7P.
4080 N.L:A=1:X=4:Y=3:GOS.4500:X=3:Y=4:GOS.4500
4090 A=2:X=3:Y=3:GOS.4500:X=4:Y=4:GOS.4500:B=0
4095 IFC=1 CU.(25,1):P."ｾﾝﾃ?";:A=(G."Y")+1
4100 A=3-A
4110 IF U.($2006,1)=0 R=R+1:G.4300
4120 R=0:B=0+1:IF(C=0)+(C=1)+(A=2) T.910
4130 X=0:Y=0:CU.(25,2):P.C.(A=1)+"ｱ"+(A=2)+"ｼ";"
0";
4140 DO:U=U+3+X+64+Y:L=80:U=XOR(L,87)
4150 F.N=(N=180)+150TO160:N.N
4155 &U=L:N=KEY
4160 IFN="R" X=X+1:IFX)7 X=0
4170 IFN="D" Y=Y+1:IFY)7 Y=0
4180 U.(N=0)+(@(Z+64+X+8+Y))0)=2
4190 @(Z+X+8+Y)=A:GOS.4500:F=X:G=Y:F.L=-1TO1:F.N
=-1TO1
4210 X=F:Y=G:DO:X=X+L:Y=Y+M:IF(X)7)+(X(0)+(Y)7)+
(Y(0) U.!:G.4260
4220 P=Z+X+8+Y
4230 IF@=AP U.!:G.4260
4240 U.A=AP:DO:X=X-L:Y=Y-M:IF(X=F)+(Y=G) U.!:G.4
260
4250 @(Z+X+8+Y)=A:GOS.4500:U.0
4260 N.M:N.L
4270 GOS.310
4300 IFR(2 T.4100
4305 D=0:E=0:F.L=0TO63:D=D+(@(Z+L)=1)
4310 E=E+(@(Z+L)=2):N.L
4320 CU.(25,2):P."□=";D;
4330 CU.(25,4):P."■=";E;
4335 IFD=E T.4360
4340 CU.(25,6):P.C.(D)E)+"ｱ"+(D(E)+"ｼ";"□";
4350 CU.(25,8):P."ﾉ ｶﾁ,";:G.4370
4360 CU.(25,6):P."ﾋｷｱﾜｹ";
4370 IFG."Y" G.3010
4380 END
4500 P.C.6:C.7;:IFB=2T.4530
4510 CU.(3+X,2+Y):P."F";
4520 CU.(3+X,2+Y+1):P."L";:R.
4530 CU.(3+X,2+Y):P."GO";
4540 CU.(3+X,2+Y+1):P."UU";:R.
5000 REM DATA
5010 +89,16,39,34,34,39,16,89
5020 +16,0,19,19,19,19,0,16
5030 +39,19,29,24,24,29,19,39
5040 +34,19,24,0,0,24,19,34
5050 +34,19,24,0,0,24,19,34
5060 +39,19,29,24,24,29,19,39
5070 +16,0,19,19,19,19,0,16
5080 +89,16,39,34,34,39,16,89

READY


)R.
****************************************
$2000   7E 20 BC 7E 20 19 7E 20
$2008   46 7E 20 F6 DE 00 86 40
$2010   E6 40 E7 C0 08 4A 26 F8
$2018   39 DE 00 86 40 7F 00 3D
$2020   E6 00 27 0A EB 00 E7 00
$2028   D1 3D 23 02 D7 3D 08 4A
$2030   26 EE 5F 96 3D 39 36 0F
$2038   3E 9B 3F 24 03 7C 00 3E
$2040   97 3F DE 3E 32 39 0C 00
$2048   02 26 04 86 40 20 02 86
$2050   C0 DE 00 8D E1 DF 40 DE
$2058   00 86 08 8D D9 DF 42 7F
$2060   00 3D 4F DE 40 8D CF 6D
$2068   00 27 03 5F 20 73 7F 00
$2070   3C C6 F0 D7 45 C6 FF D7
$2078   44 36 04 07 97 46 32 36
****************************************
$2080   04 38 97 47 5F 96 44 9B
$2088   46 97 46 85 08 26 33 96
$2090   45 9B 47 97 47 85 40 26
$2098   29 9B 46 DE 40 8D 97 6D
$20A0   00 27 1F 96 03 A1 00 27
$20A8   1A 5C 96 46 27 10 96 47
$20B0   27 0C 86 07 91 46 27 06
$20B8   05 38 91 47 26 C7 CB 04
$20C0   20 C3 5F 32 D8 3C D7 3C
$20C8   D6 44 5C D7 44 C1 02 26
$20D0   A8 D6 45 CB 00 D7 45 C1
$20D8   10 26 9A D6 3C D1 3D 23
$20E0   02 D7 3D DE 42 00 20 36
$20E8   E7 00 4C 01 40 27 03 7E
$20F0   20 63 5F 96 3D 39 C6 F8
$20F8   D7 45 C6 FF D7 44 36 04
****************************************
$2100   07 97 46 32 36 04 38 97
$2108   47 5F 96 44 9B 46 97 46
$2110   85 08 26 34 96 45 9B 47
$2118   97 47 85 40 26 2A 9B 46
$2120   DE 00 00 20 36 A6 C0 27
$2128   1F 5C 91 03 26 DC 96 46
$2130   90 44 97 46 96 47 90 45
$2138   97 47 9B 46 DE 00 00 20
$2140   36 96 03 A7 C0 5A 26 E6
$2148   32 D6 44 5C D7 44 C1 02
$2150   26 AC D6 45 CB 00 D7 45
$2158   C1 10 26 9E 39 0F 00 0F

READY
```