

On a decomposition method for finding winning strategy in Hex game

Jing Yang, Simon Liao, Mirek Pawlak

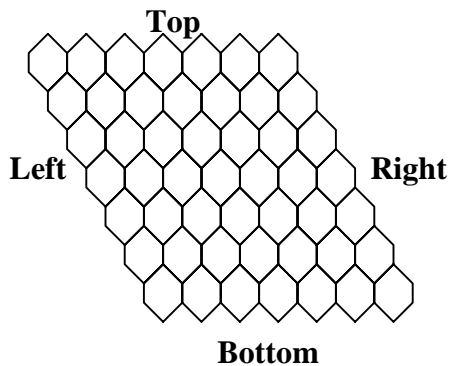
Abstract ---- This paper describes a decomposition method that results a successful finding of the winning strategies in Hex game played on a 7×7 board, which is the largest Hex board for which a winning strategy is actually known.

Index Terms ---- Artificial intelligence, Hex, winning strategy, decomposition method.

I. INTRODUCTION

Hex is a board game in which two players play in turn and try to build a connected chain of pieces across opposite sides of the board. The Hex board is a hexagonal tiling of n rows and m columns. Usually 11×11 is the widely accepted standard board size. Figure 1 is an empty 7×7 Hex board.

Figure 1: An empty Hex 7×7 board



The rules of Hex game are simple:

- One player plays Black and the other plays White. Black owns the Top and Bottom sides and White owns the Left and Right sides.
- Black player plays first.
- Players take turns placing a piece of their color on an unoccupied hexagon.

Jing Yang, Electrical & Computer Engineering Department, University of Manitoba, Room 504 Engineering Building, Winnipeg, Manitoba, Canada R3T 5V6 Phone: (204)2697144, Email: jingyang@ee.umanitoba.ca

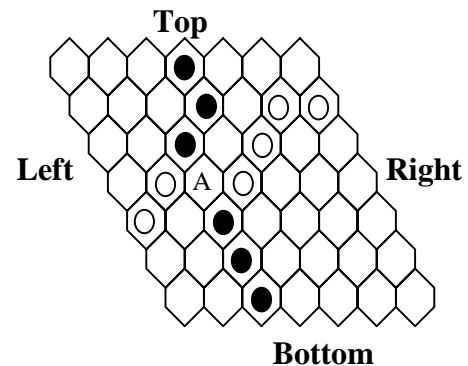
Simon Liao, Business Computing Department, University of Winnipeg, Canada

Mirek Pawlak, Electrical & Computer Engineering Department, University of Manitoba, Canada

- A game is won when one player establishes an unbroken chain of their pieces connecting their sides of the board.

For example, Figure 2 is in the middle stage of a Hex game and it is Black's turn to play. If Black plays the next move at position "A", Black will win the game. However, if Black plays at any other position rather than "A", White can play at position "A" and declare the win.

Figure 2: Take position "A" to win the game



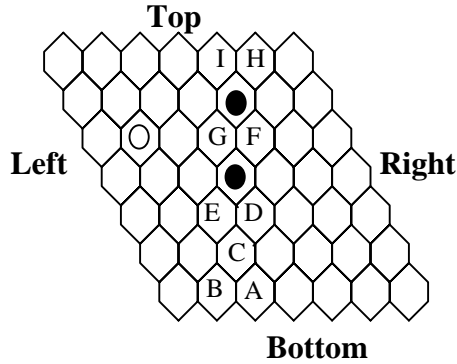
Hex was first invented by Danish mathematician Piet Hein in 1942. John Nash, who became the 1994 Nobel Laureate for Economics for his pioneering analysis of equilibria in the theory of non-competitive games, also invented the game independently in 1948. In 1949, John Nash proved that there is no tie in Hex game so that the first player has a theoretical win for Hex in any size, though there was no winning strategy was indicated. For over half a century, Hex has constantly drawn the attention of serious mathematical researchers. However, a solution (winning strategy) for Hex has continued to elude researchers.

II. THE APPROACH FOR FINDING SOLUTION

Hex is a NP-complete problem. It has an enormous number of possible games when the size of the board is larger than or equal to seven. However, two important characteristics of Hex make it possible to find a winning solution on smaller boards. First, Hex has a relatively smaller move selection space in most of situations comparing with several other board games such as Go and Chess. For example, in Figure 3, if White does not play on "A", "B", "C", "D", "E", "F", "G", "H" or "I", Black can play at "C" to win the

game, since the three black stones (in a vertical line) will allow Black to connect from Top to Bottom no matter what White does. This good characteristic means that, in this huge solution tree, many branches will terminate early. We call it the “sudden death” property.

Figure 3: The “sudden death” in Hex game



The second important characteristic of Hex is that a given Hex game sometimes can be viewed as the sum of several smaller “local games”. We call those “local games” LocalPatterns. A “local game” is a part of a region in a given Hex game. “Local games” in a given Hex game do not overlap each other. If we can find the winning strategies on some “local games”, these strategies will lead to a winning strategy for a bigger “local game”, which can be decomposed to the known “local games”. For example, considering the game in Figure 4, the two black stones will be able to connect with Top side when Black plays with a correct strategy, which will be proved in later time (in LocalPattern4). In this case, the connection status will not be threaten unless some white stones have been played on Δ hexagon region. Similarly, the two black stones can connect with Bottom side and the connection status will not be threaten unless some white stones have been played on \times local area (in LocalPattern5). Region Δ and region \times are two typical “local games”. They do not overlap each other and the strategies used on the given game in Figure 4 can be separated from the two local regions. Note that, we ignore those blank (empty) hexagons in Figure 4 because they do not have any effect to Black’s connection whether they are Black or White.

Small “local games” can be further decomposed into smaller “local games”. The game in Figure 5 is developed from Figure 4 and it gives another example for “local game” concept. The different marks in Figure 5 denote different “local games”. If we prove that those “local games” are locally connected, it is obvious that Black in Figure 5 has a connected group from Top to Bottom. If we decompose the Hex board from the beginning, and can figure out all possible ways of decomposition to White’s possible defensive moves and the further decompositions for each “local game”, we can find the winning solution. A big advantage of the idea of “local game” is that the same “local game” will appear in many different games. In fact,

we only need a total of 41 different “local games” to present the whole solution tree for all Hex games played on a 7×7 board.

Figure 4: An example for local games

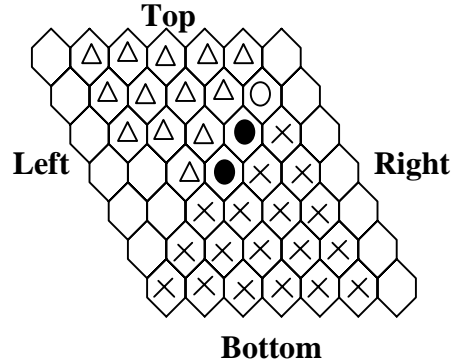
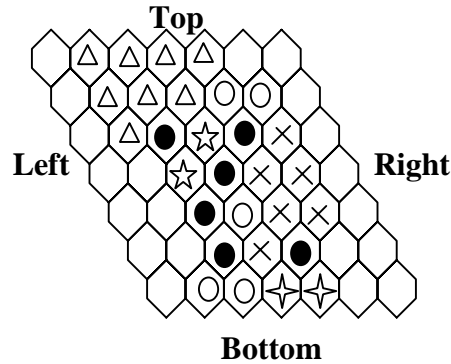


Figure 5: Local game further developed from Figure 4



III. THE IMPLEMENTATION DETAIL

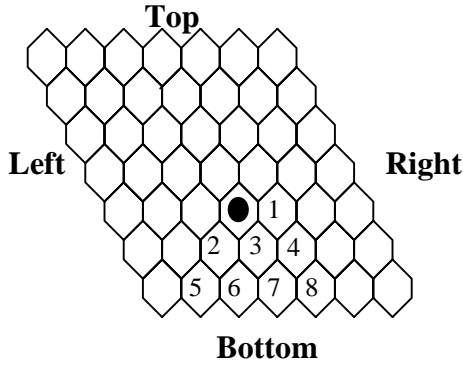
Based on the idea of “local game”, a solution tree to cover all possible White defensive moves on 7×7 board has been found. The solution has been implemented by Java applet programming to make everyone be able to check the solution online. The web site is: <http://www.ee.umanitoba.ca/~jingyang/jhex.html>.

To make the solution tree concise, we use a special logical language that is similar to C language. The value of variable WhiteMove always shows the position White plays on the given LocalPattern (corresponding the position number shown on the LocalPattern). Similarly, the value of variable BlackMove shows Black’s playing position. The variable SumOfLocalGames can be viewed as a list that stores all the “local games” information on current situation in the game. An addition operation “+” on SumOfLocalGames denotes the inserting of a new local game item on the list, and a subtraction operation “-” on it will delete an old local game item from the list. LocalPatternK denotes the Kth LocalPattern that associates with the strategies and positional information. The parameters in LocalPatternK, for example the 1,2,3,4,5,6,7, and 8 in LocalPatternK(1,2,3,4,5,6,7,8), are used to represent the positions in LocalPatternK corresponding to the given Pattern (the parent of the LocalPatternK).

Before the proof, we use Pattern13 as an example to explain our special logical language in detail.

A. Detail Explanations for LocalPattern13

Figure 6: LocalPattern13



```

If (WhiteMove == 1 || 3 || 4 || 7 || 8) {
BlackMove = 2;
SumOfLocalGames = SumOfLocalGames - LocalPattern13
+ LocalPattern2(5,6); }
else if (WhiteMove == 2 || 5 || 6) {
BlackMove = 4;
SumOfLocalGames = SumOfLocalGames - LocalPattern13
+ LocalPattern2(1,3) + LocalPattern2(7,8); }

```

The above LocalPattern13 and its associated playing strategies can be translated as follow: If the White move is on either position 1, 3, 4, 7, or 8, in LocalPattern13, Black will response on position 2. Then, on the “local games” list (SumOfLocalGames), LocalPattern13 will be deleted and a new LocalPattern2 will be added. The new LocalPattern2 will apply local game with strategies on LocalPattern2 and the new position 1 in LocalPattern2 is the old position 5 in LocalPattern13, the new position 2 in LocalPattern2 is the old position 6 in LocalPattern13. Figure 7 shows the updated situation in this case.

If the White’s move is on either position 2, 5, or 6 in LocalPattern13, Black will response on position 4. Then, on the “local games” list (SumOfLocalGames), LocalPattern13 will be removed and two new LocalPattern2s will be added. In one of LocalPattern2s, new position 1 is on the old position 1 in LocalPattern13, and new position 2 is on the old position 3 in LocalPattern13. In the other LocalPattern2, new position 1 is on the old position 7 in LocalPattern13, and new position 2 is on the old position 8 in LocalPattern13. Figure 8 shows the updated situation in this case.

Figure 7: Developed from LocalPattern13 first rule

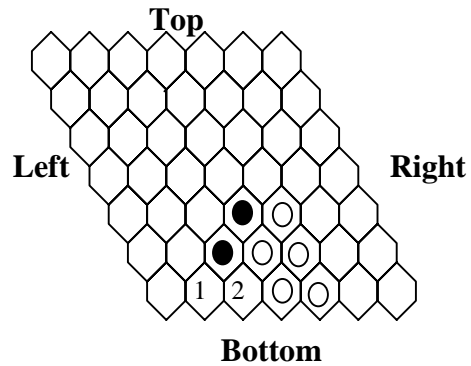
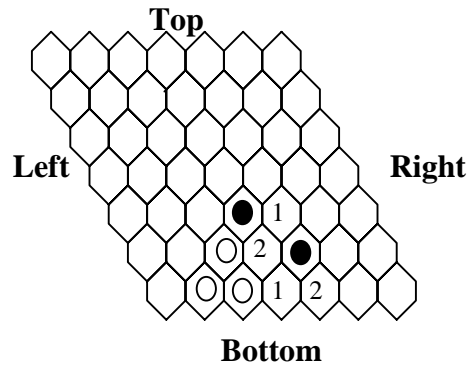


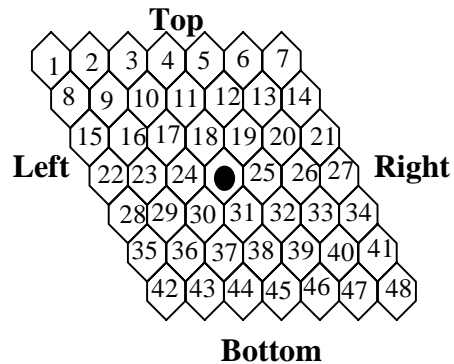
Figure 8: Developed from LocalPattern13 second rule



B. Whole solution tree description

All other LocalPatterns can be explained in the same way as in LocalPattern13. Since some LocalPatterns have identical associated playing strategies based on pattern position numbers though their pattern shapes are different, we collect them into one LocalPattern distinguished by (a), (b) or more. For example, LocalPattern2(a) and LocalPattern2(b) belong to this case.

Local Pattern 1:

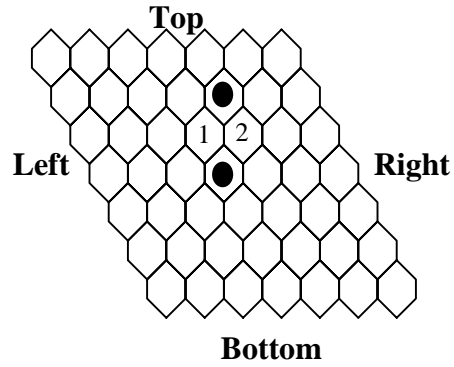


```

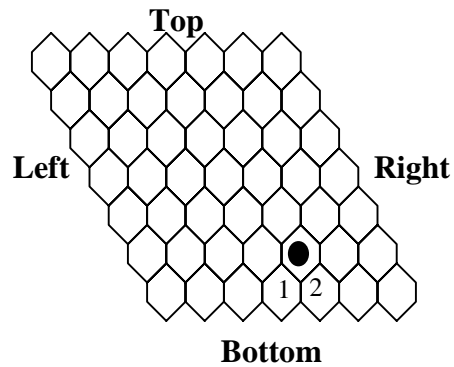
If (WhiteMove == 1 || 2 || 3 || 4 || 8 || 9 || 10 || 11 || 15 || 16 ||
17 || 22 || 23 || 24) {
BlackMove = 12;
SumOfLocalGames = SumOfLocalGames - LocalPattern1
+ LocalPattern2(5,6) + LocalPattern2(18,19) +
LocalPattern3(7,13,14,20,21,25,26,30,31,32,33,34,36,37,38
,39,40,41,42,43,44,45,46,47, 48); }
else if (WhiteMove == 14 || 20 || 21) {
BlackMove = 37;
SumOfLocalGames = SumOfLocalGames - LocalPattern1
+ LocalPattern2(30,31)+ LocalPattern2(43,44) +
LocalPattern3(42,36,35,29,28,24,23,19,18,17,16,15,13,12,
11,10,9,8,7,6,5,4,3,2,1); }
else if (WhiteMove == 13 || 7) {
BlackMove = 19;
SumOfLocalGames = SumOfLocalGames - LocalPattern1
+ LocalPattern4(24,18,17,16,12,11,10,9,6,5,4,3,2) +
LocalPattern5(20,25,26,30,31,32,33,36,37,38,39,40,42,43,
44,45,46,47); }
else if (WhiteMove == 12) {
BlackMove = 19;
SumOfLocalGames = SumOfLocalGames - LocalPattern1
+ LocalPattern6(24,18,17,16,13,11,10,9,7,6,5,4,3,2) +
LocalPattern5(20,25,26,30,31,32,33,36,37,38,39,40,42,43,
44,45,46,47); }
else if (WhiteMove == 6) {
BlackMove = 20;
SumOfLocalGames = SumOfLocalGames - LocalPattern1
+ LocalPattern2(19,25) +
LocalPattern7(24,18,17,16,14,13,11,10,9,7,5,4,3,2) +
LocalPattern8(26,30,31,32,33,36,37,38,39,40,42,43,44,45,
46,47); }
else if (WhiteMove == 5) {
BlackMove = 13;
SumOfLocalGames = SumOfLocalGames - LocalPattern1
+ LocalPattern2(6,7) +
LocalPattern9(19,18,17,12,11,10,4,3) +
LocalPattern10(20,25,26,30,31,32,33,36,37,38,39,40,42,43,
44,45,46,47); }
else if (WhiteMove == 18) {
BlackMove = 19;
SumOfLocalGames = SumOfLocalGames - LocalPattern1
+ LocalPattern11(3,4,5,6,7,10,11,12,13,14,16,17,20,23,24,
25,26,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,
45,46,47,48); }
else if (WhiteMove == 19) {
BlackMove = 18;
SumOfLocalGames = SumOfLocalGames - LocalPattern1
+LocalPattern12(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,
20,21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,
39,40,41,42,43,44,45,46,47,48);
else (omit because of the symmetric characteristics)

```

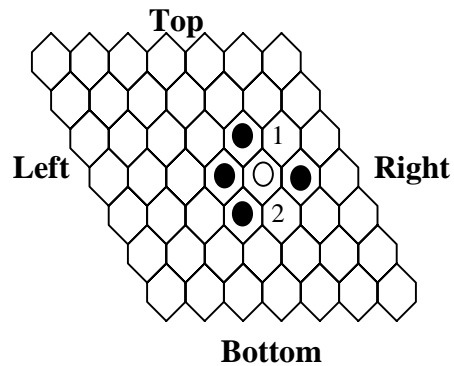
Local Pattern 2(a):



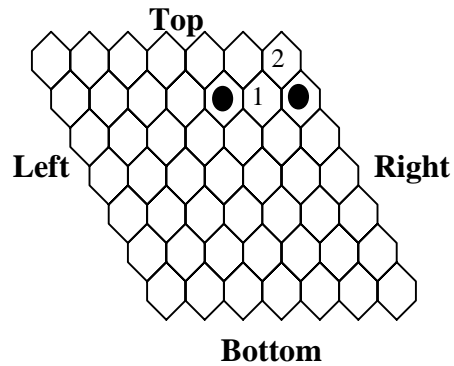
Local Pattern 2(b):



Local Pattern 2(c):



Local Pattern 2(d):



```

If (WhiteMove == 1) {
BlackMove = 2;
SumOfLocalGames = SumOfLocalGames - LocalPattern2;
}

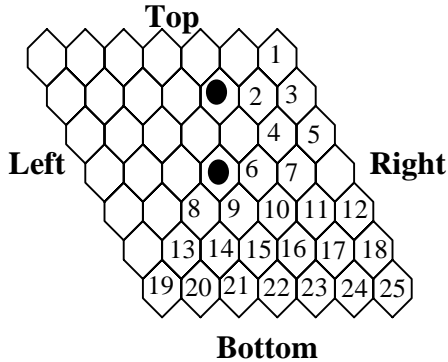
```

```

else if (WhiteMove == 2) {
BlackMove = 1;
SumOfLocalGames = SumOfLocalGames - LocalPattern2;
}

```

Local Pattern 3:

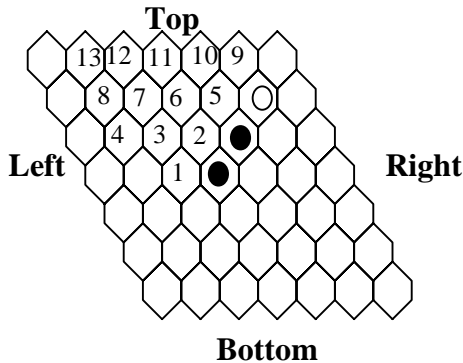


```

If (WhiteMove == 1 || 2 || 3 || 4 || 5 || 6 || 7 || 10 || 11 || 12 ||
13 || 15 || 16 || 17 || 18 || 19 || 22 || 23 || 24 || 25) {
BlackMove = 14;
SumOfLocalGames = SumOfLocalGames - LocalPattern3
+ LocalPattern2(8,9) + LocalPattern2(20,21); }
else if (WhiteMove == 8 || 14 || 20) {
BlackMove = 10;
SumOfLocalGames = SumOfLocalGames - LocalPattern3
+ LocalPattern2(6,9) +
LocalPattern13(11,15,16,17,21,22,23,24); }
else if (WhiteMove == 21) {
BlackMove = 13;
SumOfLocalGames = SumOfLocalGames - LocalPattern3
+ LocalPattern2(19,20) +
LocalPattern9(8,9,10,14,15,16,22,23); }
else if (WhiteMove == 9) {
BlackMove = 8;
SumOfLocalGames = SumOfLocalGames - LocalPattern3
+ LocalPattern14(1,2,3,4,5,6,7,10,11,13,14,15,16,19,20,21,
22,23); }

```

Local Pattern 4:



```

If (WhiteMove == 1 || 2 || 3 || 4 || 6 || 7 || 8 || 11 || 12 || 13) {
BlackMove = 5;
SumOfLocalGames = SumOfLocalGames - LocalPattern4
+ LocalPattern2(10,9); }
else if (WhiteMove == 5 || 9) {

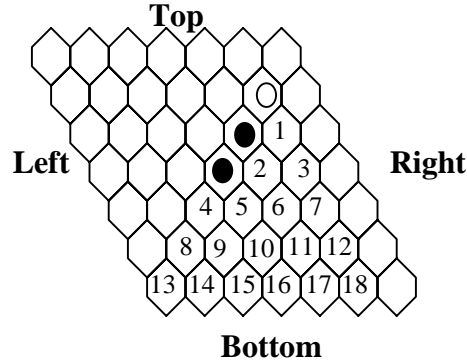
```

```

BlackMove = 3;
SumOfLocalGames = SumOfLocalGames - LocalPattern4
+ LocalPattern2(1,2) +
LocalPattern13(4,6,7,8,10,11,12,13); }
else if (WhiteMove == 10) {
BlackMove = 5;
SumOfLocalGames = SumOfLocalGames - LocalPattern4
+ LocalPattern15(1,2,3,6,7,9,11,12); }

```

Local Pattern 5:

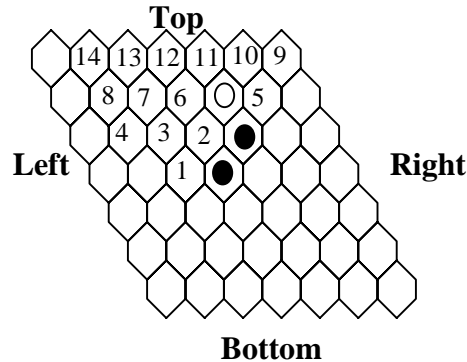


```

If (WhiteMove == (1 || 2 || 3 || 6 || 7 || 10 || 11 || 12 || 16 || 17
|| 18) {
BlackMove = 9;
SumOfLocalGames = SumOfLocalGames - LocalPattern5
+ LocalPattern2(4,5) + LocalPattern2(14,15); }
else if (WhiteMove == 4 || 8 || 9 || 13 || 14) {
BlackMove = 6;
SumOfLocalGames = SumOfLocalGames - LocalPattern5
+ LocalPattern2(2,5) +
LocalPattern13(7,10,11,12,15,16,17,18); }
else if (WhiteMove == 15) {
BlackMove = 8;
SumOfLocalGames = SumOfLocalGames - LocalPattern5
+ LocalPattern2(13,14) +
LocalPattern9(4,5,6,9,10,11,16,17); }
else if (WhiteMove == 5) {
BlackMove = 4;
SumOfLocalGames = SumOfLocalGames - LocalPattern5
+ LocalPattern16(1,2,3,6,7,8,9,10,11,13,14,15,16,17); }

```

Local Pattern 6:



```

If (WhiteMove == (1 || 2 || 3 || 4 || 6 || 7 || 8 || 11 || 12 || 13 ||
14) {
BlackMove = 5;
SumOfLocalGames = SumOfLocalGames - LocalPattern6
+ LocalPattern2(9,10); }

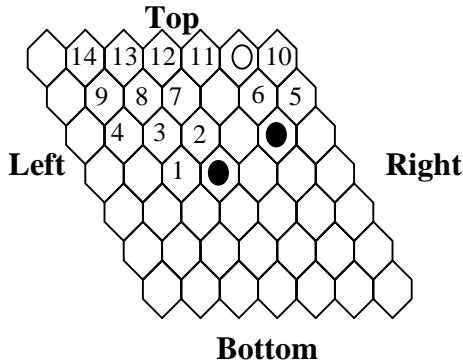
```

```

else if (WhiteMove == 10 || 9 || 5) {
BlackMove = 3;
SumOfLocalGames = SumOfLocalGames - LocalPattern6
+ LocalPattern2(1,2) +
LocalPattern13(4,6,7,8,11,12,13,14); }

```

Local Pattern 7:

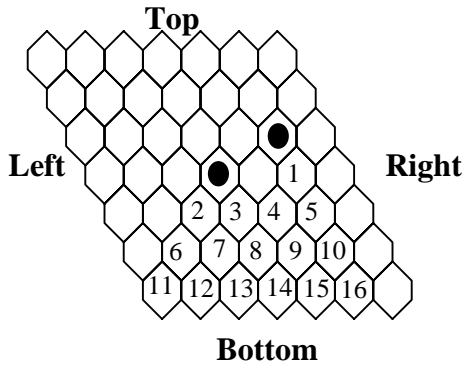


```

If (WhiteMove == (1 || 2 || 3 || 4 || 7 || 8 || 9 || 11 || 12 || 13 ||
14) {
BlackMove = 10;
SumOfLocalGames = SumOfLocalGames - LocalPattern7
+ LocalPattern2(5,6); }
else if (WhiteMove == 10 || 6 || 5) {
BlackMove = 3;
SumOfLocalGames = SumOfLocalGames - LocalPattern7
+ LocalPattern2(1,2) +
LocalPattern13(4,7,8,9,11,12,13,14); }

```

Local Pattern 8:



```

If (WhiteMove == 1 || 4 || 5 || 6 || 8 || 9 || 10 || 11 || 14 || 15 ||
16) {
BlackMove = 7;
SumOfLocalGames = SumOfLocalGames - LocalPattern8
+ LocalPattern2(2,3) + LocalPattern2(12,13); }
else if (WhiteMove == 2 || 7 || 12) {
BlackMove = 4;
SumOfLocalGames = SumOfLocalGames - LocalPattern8
+ LocalPattern2(1,3) +
LocalPattern13(5,8,9,10,13,14,15,16); }
else if (WhiteMove == 13) {
BlackMove = 6;
SumOfLocalGames = SumOfLocalGames - LocalPattern8
+ LocalPattern2(11,12) + LocalPattern9(2,3,4,7,8,9,14,15); }
else if (WhiteMove == 3) {

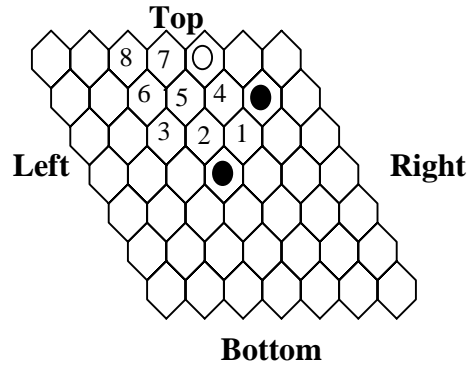
```

```

BlackMove = 2;
SumOfLocalGames = SumOfLocalGames - LocalPattern8
+ LocalPattern17(1,4,5,6,7,8,9,11,12,13,14,15); }

```

Local Pattern 9:

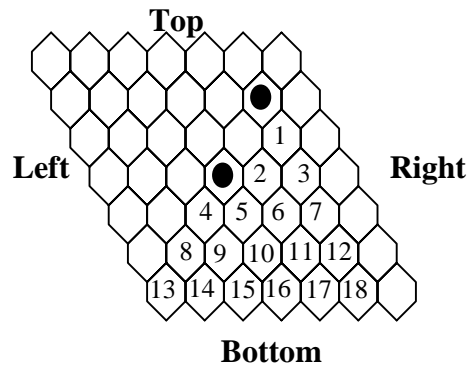


```

If (WhiteMove == 2 || 3 || 4 || 5 || 6 || 7 || 8) {
BlackMove = 1;
SumOfLocalGames = SumOfLocalGames - LocalPattern9; }
else if (WhiteMove == 1) {
BlackMove = 2;
SumOfLocalGames = SumOfLocalGames - LocalPattern9
+ LocalPattern18(3,4,5,6,7,8); }

```

Local Pattern 10:

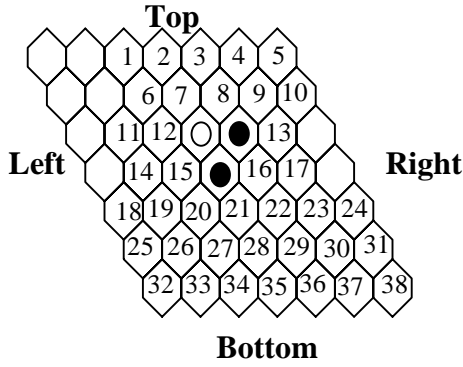


```

If (WhiteMove == 1 || 2 || 3 || 6 || 7 || 8 || 10 || 11 || 12 || 13 ||
16 || 17 || 18) {
BlackMove = 9;
SumOfLocalGames = SumOfLocalGames - LocalPattern10
+ LocalPattern2(4,5) + LocalPattern2(14,15); }
else if (WhiteMove == 4 || 9 || 14) {
BlackMove = 6;
SumOfLocalGames = SumOfLocalGames - LocalPattern10
+ LocalPattern2(2,5) +
LocalPattern13(7,10,11,12,15,16,17,18); }
else if (WhiteMove == 15) {
BlackMove = 8;
SumOfLocalGames = SumOfLocalGames - LocalPattern10
+ LocalPattern2(13,14) +
LocalPattern9(4,5,6,9,10,11,16,17); }
else if (WhiteMove == 5) {
BlackMove = 4;
SumOfLocalGames = SumOfLocalGames - LocalPattern10
+ LocalPattern19(1,2,3,6,7,8,9,10,11,13,14,15,16,17); }

```

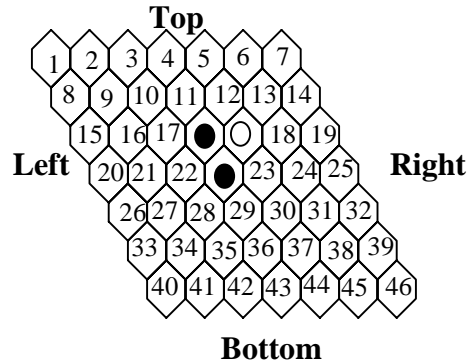
Local Pattern 11:



```
If (WhiteMove == 1 || 2 || 3 || 5 || 6 || 7 || 10 || 11 || 12 || 14 ||
15 || 18 || 19 || 25) {
BlackMove = 4;
SumOfLocalGames = SumOfLocalGames - LocalPattern11
+ LocalPattern2(8,9) +
LocalPattern5(13,16,17,20,21,22,23,26,27,28,29,30,32,33,
34,35,36,37); }
else if (WhiteMove == 8) {
BlackMove = 9;
SumOfLocalGames = SumOfLocalGames - LocalPattern11
+ LocalPattern2(4,5) + LocalPattern5(13,16,17,20,21,22,23,
26,27,28,29,30,32,33,34,35,36,37); }
else if (WhiteMove == 9) {
BlackMove = 8;
SumOfLocalGames = SumOfLocalGames - LocalPattern11
+ LocalPattern2(3,4) + LocalPattern5(13,16,17,20,21,22,23,
26,27,28,29,30,32,33,34,35,36,37); }
else if (WhiteMove == 13 || 16 || 17 || 22 || 23 || 24 || 28 || 29
|| 30 || 31 || 35 || 36 || 37 || 38) {
BlackMove = 27;
SumOfLocalGames = SumOfLocalGames - LocalPattern11
+ LocalPattern2(20,21) + LocalPattern2(33,34) +
LocalPattern14(32,26,25,19,18,15,14,12,11,9,8,7,6,5,4,3,2,
1); }
else if (WhiteMove == 26 || 32) {
BlackMove = 20;
SumOfLocalGames = SumOfLocalGames - LocalPattern11
+ LocalPattern4(16,21,22,23,27,28,29,30,33,34,35,36,37) +
LocalPattern16(19,15,14,12,11,9,8,7,6,5,4,3,2,1); }
else if (WhiteMove == 33) {
BlackMove = 19;
SumOfLocalGames = SumOfLocalGames - LocalPattern11
+ LocalPattern2(15,20) +
LocalPattern17(14,12,11,9,8,7,6,5,4,3,2,1) +
LocalPattern7(16,21,22,23,27,28,29,30,32,34,35,36,37); }
else if (WhiteMove == 34) {
BlackMove = 26;
SumOfLocalGames = SumOfLocalGames - LocalPattern11
+ LocalPattern2(32,33) +
LocalPattern9(20,21,22,27,28,29,35,36) +
LocalPattern17(14,12,11,9,8,7,6,5,4,3,2,1); }
else if (WhiteMove == 27) {
BlackMove = 20;
SumOfLocalGames = SumOfLocalGames - LocalPattern11
+
LocalPattern6(16,21,22,23,26,28,29,30,32,33,34,35,36,37)
```

```
+ LocalPattern16(19,15,14,12,11,9,8,7,6,5,4,3,2,1); }
else if (WhiteMove == 21) {
BlackMove = 20;
SumOfLocalGames = SumOfLocalGames - LocalPattern11
+
LocalPattern16(13,16,17,22,23,26,27,28,29,32,33,34,35,36)
+ LocalPattern16(19,15,14,12,11,9,8,7,6,5,4,3,2,1); }
else if (WhiteMove == 20) {
BlackMove = 21;
SumOfLocalGames = SumOfLocalGames - LocalPattern11
+ LocalPattern20(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,
18,19,22,23,24,25,26,27,28,29,30,31,32,33,34,35,
36,37,38); }
else if (WhiteMove == 4) {
BlackMove = 8;
SumOfLocalGames = SumOfLocalGames - LocalPattern11
+ LocalPattern35(1,2,3,6,7,11,12,13,14,15,16,17,18,19,20,
21,22,23,24,25,26,27,28,29,30,31,32,33,34,35,36,37,38); }
```

Local Pattern 12:



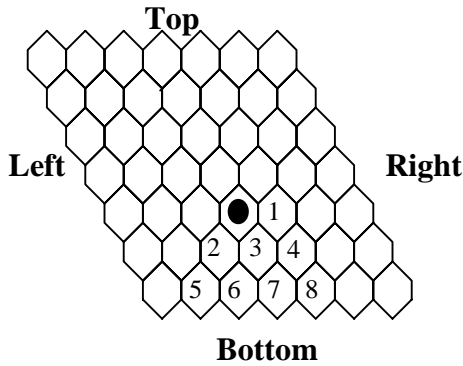
```
If (WhiteMove == 1 || 2 || 7 || 8 || 9 || 13 || 14 || 15 || 16 || 18 ||
19 || 20 || 21 || 22 || 23 || 24 || 25 || 26 || 27 || 30 || 31 || 32 || 33
|| 34 || 36 || 37 || 38 || 39 || 40 || 43 || 44 || 45 || 46) {
BlackMove = 35;
SumOfLocalGames = SumOfLocalGames - LocalPattern12
+ LocalPattern2(28,29) + LocalPattern2(41,42) +
LocalPattern13(17,12,11,10,6,5,4,3); }
else if (WhiteMove == 28 || 35 || 41) {
BlackMove = 30;
SumOfLocalGames = SumOfLocalGames - LocalPattern12
+ LocalPattern2(23,29) +
LocalPattern13(31,36,37,38,42,43,44,45) +
LocalPattern13(17,12,11,10,6,5,4,3); }
else if (WhiteMove == 3 || 4 || 10 || 11 || 17) {
BlackMove = 12;
SumOfLocalGames = SumOfLocalGames - LocalPattern12
+ LocalPattern2(5,6) +
LocalPattern3(7,13,14,18,19,23,24,28,29,30,31,32,34,35,36,
37,38,39,40,41,42,43,44,45, 46); }
else if (WhiteMove == 42) {
BlackMove = 34;
SumOfLocalGames = SumOfLocalGames - LocalPattern12
+ LocalPattern2(40,41) +
LocalPattern9(28,29,30,35,36,37,43,44) +
LocalPattern13(17,12,11,10,6,5,4,3); }
else if (WhiteMove == 29) {
BlackMove = 28;
```

```

SumOfLocalGames = SumOfLocalGames - LocalPattern12
+ LocalPattern20(44,43,42,41,40,37,36,35,34,33,31,30,27,
24,23,22,21,19,18,17,16,15,14,13,12,11,10,9,8,7,6,5,4,3,2,1
); }
else if (WhiteMove == 6 || 12) {
BlackMove = 28;
SumOfLocalGames = SumOfLocalGames - LocalPattern12
+ LocalPattern13(29,34,35,36,40,41,42,43) +
LocalPattern22(27,22,21,17,16,15,11,10,9,8,5,4,3,2,1); }
else if (WhiteMove == 5) {
BlackMove = 12;
SumOfLocalGames = SumOfLocalGames - LocalPattern12
+ LocalPattern23(17,11,10,6,4,3) +
LocalPattern3(7,13,14,18,19,23,24,28,29,30,31,32,34,35,36
,37,38,39,40,41,42,43,44,45, 46); }

```

Local Pattern 13:

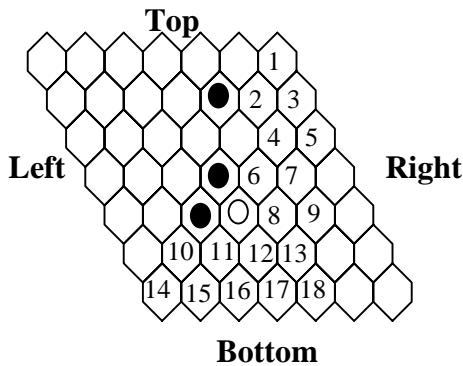


```

If (WhiteMove == 1 || 3 || 4 || 7 || 8) {
BlackMove = 2;
SumOfLocalGames = SumOfLocalGames - LocalPattern13
+ LocalPattern2(5,6); }
else if (WhiteMove == 2 || 5 || 6) {
BlackMove = 4;
SumOfLocalGames = SumOfLocalGames - LocalPattern13
+ LocalPattern2(1,3) + LocalPattern2(7,8); }

```

Local Pattern 14:



```

If (WhiteMove == 1 || 2 || 3 || 4 || 5 || 6 || 7 || 8 || 9 || 12 || 13 ||
14 || 16 || 17 || 18) {
BlackMove = 15;
SumOfLocalGames = SumOfLocalGames - LocalPattern14
+ LocalPattern2(10,11); }
else if (WhiteMove == 10) {
BlackMove = 11;

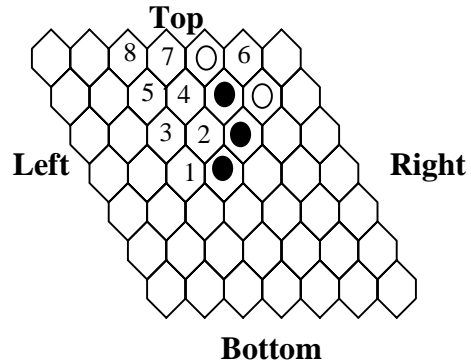
```

```

SumOfLocalGames = SumOfLocalGames - LocalPattern14
+ LocalPattern2(15,16); }
else if (WhiteMove == 11) {
BlackMove = 10;
SumOfLocalGames = SumOfLocalGames - LocalPattern14
+ LocalPattern2(14,15); }
else if (WhiteMove == 15) {
BlackMove = 11;
SumOfLocalGames = SumOfLocalGames - LocalPattern14
+ LocalPattern24(1,2,3,4,5,6,7,8,9,12,13,16,17,18); }

```

Local Pattern 15:

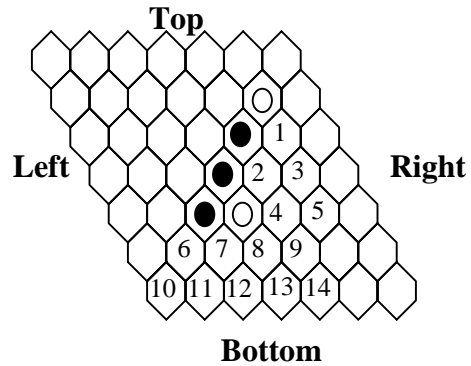


```

If (WhiteMove == 1 || 2 || 3 || 4 || 5 || 7 || 8) {
BlackMove = 6;
SumOfLocalGames = SumOfLocalGames -
LocalPattern15; }
else if (WhiteMove == 6) {
BlackMove = 5;
SumOfLocalGames = SumOfLocalGames - LocalPattern15
+ LocalPattern2(7,8) + LocalPattern25(1,2,3,4); }

```

Local Pattern 16:



```

If (WhiteMove == 1 || 2 || 3 || 4 || 5 || 8 || 9 || 10 || 12 || 13 ||
14) {
BlackMove = 11;
SumOfLocalGames = SumOfLocalGames - LocalPattern16
+ LocalPattern2(6,7); }
else if (WhiteMove == 6) {
BlackMove = 7;
SumOfLocalGames = SumOfLocalGames - LocalPattern16
+ LocalPattern2(11,12); }
else if (WhiteMove == 7) {
BlackMove = 6;
SumOfLocalGames = SumOfLocalGames - LocalPattern16
+ LocalPattern2(10,11); }

```

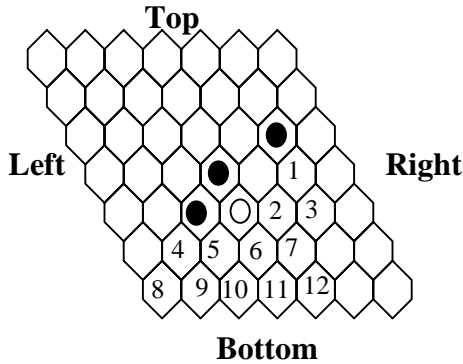


```

else if (WhiteMove == 11) {
BlackMove = 7;
SumOfLocalGames = SumOfLocalGames - LocalPattern16
+ LocalPattern26(1,2,3,4,5,8,9,12,13,14); }

```

Local Pattern 17:

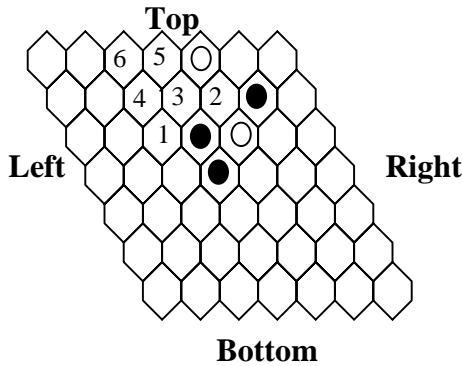


```

If (WhiteMove == 1 || 2 || 3 || 6 || 7 || 8 || 10 || 11 || 12) {
BlackMove = 9;
SumOfLocalGames = SumOfLocalGames - LocalPattern17
+ LocalPattern2(4,5); }
else if (WhiteMove == 4) {
BlackMove = 5;
SumOfLocalGames = SumOfLocalGames - LocalPattern17
+ LocalPattern2(9,10); }
else if (WhiteMove == 5) {
BlackMove = 4;
SumOfLocalGames = SumOfLocalGames - LocalPattern17
+ LocalPattern2(8,9); }
else if (WhiteMove == 9) {
BlackMove = 5;
SumOfLocalGames = SumOfLocalGames - LocalPattern17
+ LocalPattern27(1,2,3,6,7,10,11,12); }

```

Local Pattern 18:

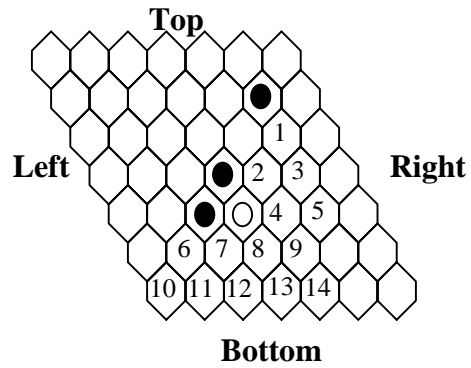


```

If (WhiteMove == 1 || 3 || 4 || 5 || 6) {
BlackMove = 2;
SumOfLocalGames = SumOfLocalGames -
LocalPattern18; }
else if (WhiteMove == 2) {
BlackMove = 4;
SumOfLocalGames = SumOfLocalGames -
LocalPattern18+ LocalPattern2(1,3) + LocalPattern2(5,6);

```

Local Pattern 19:

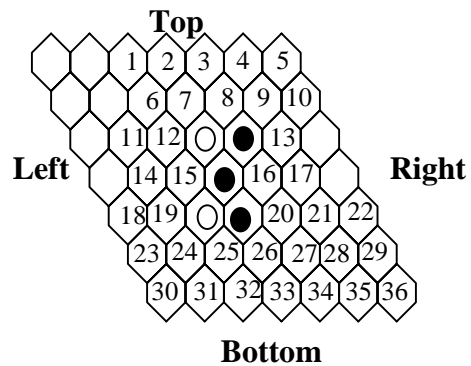


```

If (WhiteMove == 1 || 2 || 3 || 4 || 5 || 8 || 9 || 10 || 12 || 13 ||
14) {
BlackMove = 11;
SumOfLocalGames = SumOfLocalGames - LocalPattern19
+ LocalPattern2(6,7); }
else if (WhiteMove == 6) {
BlackMove = 7;
SumOfLocalGames = SumOfLocalGames - LocalPattern19
+ LocalPattern2(11,12); }
else if (WhiteMove == 7) {
BlackMove = 6;
SumOfLocalGames = SumOfLocalGames - LocalPattern19
+ LocalPattern2(10,11); }
else if (WhiteMove == 11) {
BlackMove = 7;
SumOfLocalGames = SumOfLocalGames - LocalPattern19
+ LocalPattern28(1,2,3,4,5,8,9,12,13,14); }

```

Local Pattern 20:



```

If (WhiteMove == 1 || 2 || 3 || 5 || 6 || 7 || 10 || 11 || 12 || 13 ||
14 || 15 || 16 || 17 || 18 || 19 || 21 || 22 || 23 || 24 || 28 || 29 || 30
|| 35 || 36) {
BlackMove = 4;
SumOfLocalGames = SumOfLocalGames - LocalPattern20
+ LocalPattern2(8,9) +
LocalPattern13(20,25,26,27,31,32,33,34); }
else if (WhiteMove == 8) {
BlackMove = 9;
SumOfLocalGames = SumOfLocalGames - LocalPattern20
+ LocalPattern2(4,5) +
LocalPattern13(20,25,26,27,31,32,33,34); }
else if (WhiteMove == 9) {
BlackMove = 8;

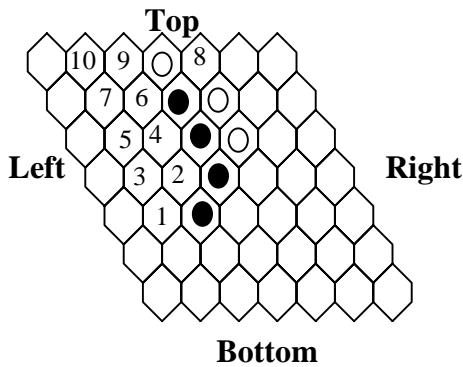
```

```

SumOfLocalGames = SumOfLocalGames - LocalPattern20
+ LocalPattern2(3,4) +
LocalPattern13(20,25,26,27,31,32,33,34); }
else if (WhiteMove == 20 || 26 || 27 || 33 || 34) {
BlackMove = 25;
SumOfLocalGames = SumOfLocalGames - LocalPattern20
+ LocalPattern2(31,32) + LocalPattern14(30,24,23,19,18,
15,14,12,11,9,8,7,6,5,4,3,2,1); }
else if (WhiteMove == 25 || 31) {
BlackMove = 4;
SumOfLocalGames = SumOfLocalGames - LocalPattern20
+ LocalPattern2(8,9) + LocalPattern22(13,16,17,20,21,22,
26,27,28,29,32,33,34,35,36); }
else if (WhiteMove == 32) {
BlackMove = 25;
SumOfLocalGames = SumOfLocalGames - LocalPattern20
+ LocalPattern23(20,26,27,31,33,34) + LocalPattern14(30,
24,23,19,18,15,14,12,11,9,8,7,6,5,4,3,2,1); }
else if (WhiteMove == 4) {
BlackMove = 8;
SumOfLocalGames = SumOfLocalGames - LocalPattern20
+ LocalPattern29(1,2,3,6,7,11,12,13,14,15,16,17,18,19,20,
21,22,23,24,25,26,27,28,29,30, 31,32,33,34,35,36); }

```

Local Pattern 21:

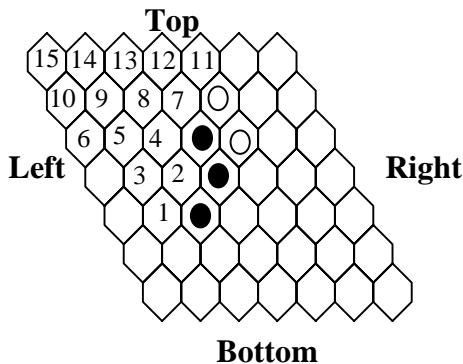


```

If (WhiteMove == 1 || 2 || 3 || 4 || 5 || 6 || 7 || 9 || 10) {
BlackMove = 8;
SumOfLocalGames = SumOfLocalGames -
LocalPattern21; }
else if (WhiteMove == 8) {
BlackMove = 7;
SumOfLocalGames = SumOfLocalGames - LocalPattern21
+ LocalPattern2(9,10) + LocalPattern30(1,2,3,4,5,6); }

```

Local Pattern 22:

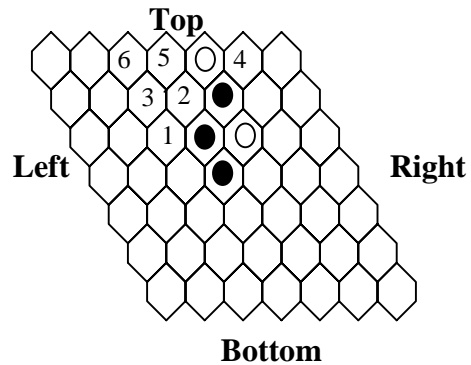


```

If (WhiteMove == 1 || 2 || 3 || 4 || 5 || 6 || 8 || 9 || 10 || 13 || 14
|| 15) {
BlackMove = 7;
SumOfLocalGames = SumOfLocalGames - LocalPattern22
+ LocalPattern2(11,12); }
else if (WhiteMove == 7 || 11) {
BlackMove = 5;
SumOfLocalGames = SumOfLocalGames - LocalPattern22
+ LocalPattern13(6,8,9,10,12,13,14,15) +
LocalPattern25(1,2,3,4); }
else if (WhiteMove == 12) {
BlackMove = 7;
SumOfLocalGames = SumOfLocalGames - LocalPattern22
+ LocalPattern21(1,2,3,4,5,8,9,11,13,14); }

```

Local Pattern 23:

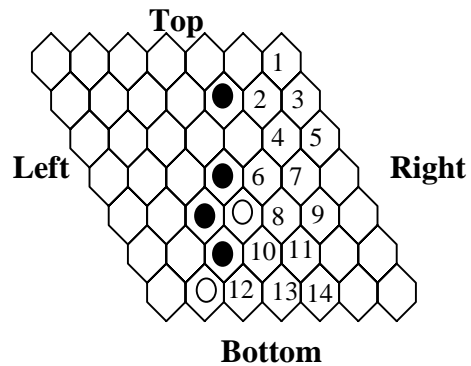


```

If (WhiteMove == 1 || 2 || 3 || 5 || 6) {
BlackMove = 4;
SumOfLocalGames = SumOfLocalGames -
LocalPattern23; }
else if (WhiteMove == 4) {
BlackMove = 3;
SumOfLocalGames = SumOfLocalGames - LocalPattern23
+ LocalPattern2(1,2) + LocalPattern2(5,6); }

```

Local Pattern 24:



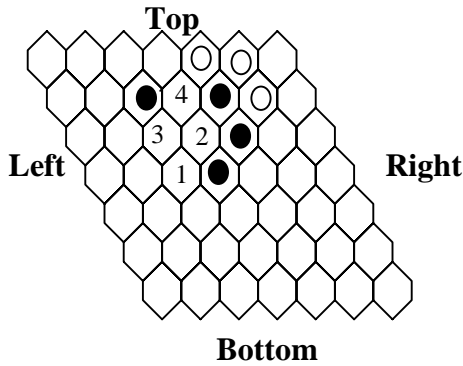
```

If (WhiteMove == 1 || 2 || 3 || 4 || 5 || 6 || 7 || 8 || 9 || 10 || 11 ||
13 || 14) {
BlackMove = 12;
SumOfLocalGames = SumOfLocalGames -
LocalPattern24; }
else if (WhiteMove == 12) {
BlackMove = 11;
SumOfLocalGames = SumOfLocalGames - LocalPattern24

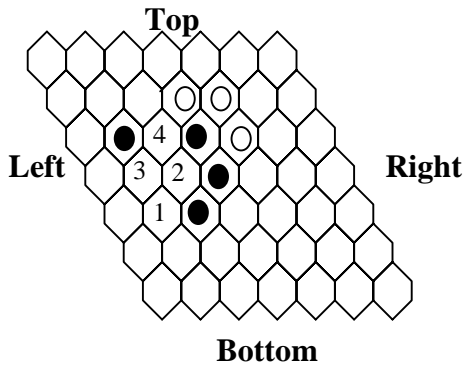
```

```
+ LocalPattern2(13,14) +
LocalPattern31(1,2,3,4,5,6,7,8,9,10); }
```

Local Pattern 25(a):

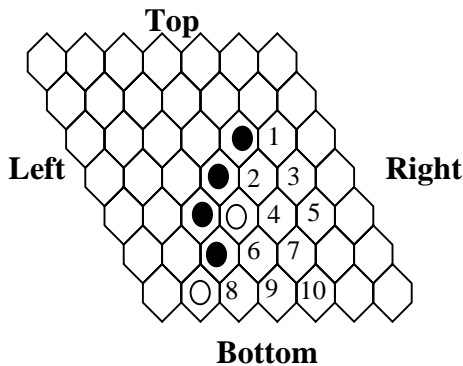


Local Pattern 25(b):



```
If (WhiteMove == 1 || 2 || 3) {
BlackMove = 4;
SumOfLocalGames = SumOfLocalGames -
LocalPattern25; }
else if (WhiteMove == 4) {
BlackMove = 3;
SumOfLocalGames = SumOfLocalGames - LocalPattern25
+ LocalPattern2(1,2); }
```

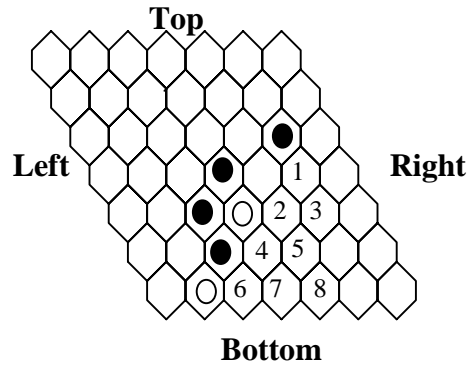
Local Pattern 26:



```
If (WhiteMove == 1 || 2 || 3 || 4 || 5 || 6 || 7 || 9 || 10) {
BlackMove = 8;
SumOfLocalGames = SumOfLocalGames -
LocalPattern26; }
else if (WhiteMove == 8) {
BlackMove = 7;
SumOfLocalGames = SumOfLocalGames - LocalPattern26; }
```

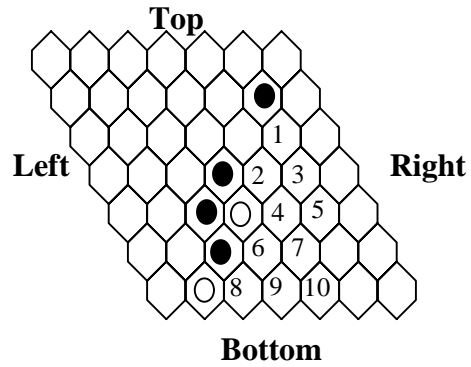
```
SumOfLocalGames = SumOfLocalGames - LocalPattern26
+ LocalPattern2(9,10) + LocalPattern32(1,2,3,4,5,6); }
```

Local Pattern 27:



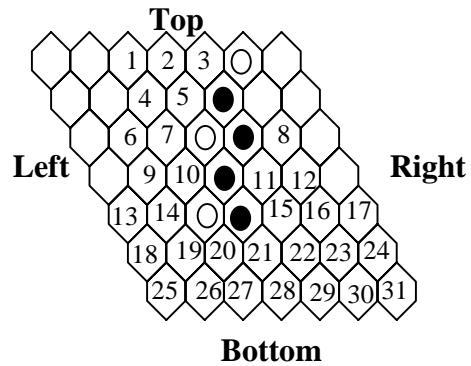
```
If (WhiteMove == 1 || 2 || 3 || 4 || 5 || 7 || 8) {
BlackMove = 6;
SumOfLocalGames = SumOfLocalGames -
LocalPattern27; }
else if (WhiteMove == 6) {
BlackMove = 5;
SumOfLocalGames = SumOfLocalGames - LocalPattern27
+ LocalPattern2(7,8) + LocalPattern33(1,2,3,4); }
```

Local Pattern 28:



```
If (WhiteMove == 1 || 2 || 3 || 4 || 5 || 6 || 7 || 9 || 10) {
BlackMove = 8;
SumOfLocalGames = SumOfLocalGames -
LocalPattern28; }
else if (WhiteMove == 8) {
BlackMove = 7;
SumOfLocalGames = SumOfLocalGames - LocalPattern28
+ LocalPattern2(9,10) + LocalPattern34(1,2,3,4,5,6); }
```

Local Pattern 29:

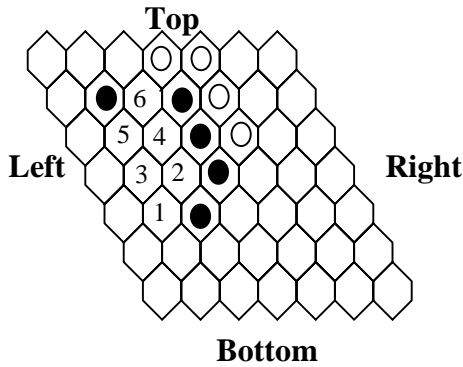


```

If (WhiteMove == 1 || 2 || 4 || 5 || 6 || 7 || 8 || 9 || 10 || 11 || 12
|| 13 || 14 || 16 || 17 || 18 || 19 || 23 || 24 || 25 || 30 || 31) {
BlackMove = 3;
SumOfLocalGames = SumOfLocalGames - LocalPattern29
+ LocalPattern13(15,20,21,22,26,27,28,29); }
else if (WhiteMove == 15 || 21 || 22 || 28 || 29) {
BlackMove = 20;
SumOfLocalGames = SumOfLocalGames - LocalPattern29
+ LocalPattern2(26,27) +
LocalPattern24(25,19,18,14,13,10,9,7,6,5,4,3,2,1); }
else if (WhiteMove == 20 || 26) {
BlackMove = 3;
SumOfLocalGames = SumOfLocalGames - LocalPattern29
+ LocalPattern22(8,11,12,15,16,17,21,22,23,24,27,28,29,
30,31); }
else if (WhiteMove == 27) {
BlackMove = 20;
SumOfLocalGames = SumOfLocalGames - LocalPattern29
+ LocalPattern23(15,21,22,26,28,29) +
LocalPattern24(25,19,18,14,13,10,9,7,6,5,4,3,2,1); }
else if (WhiteMove == 3) {
BlackMove = 4;
SumOfLocalGames = SumOfLocalGames - LocalPattern29
+ LocalPattern2(1,2) + LocalPattern36(5,6,7,8,9,10,11,12,13
,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31); }

```

Local Pattern 30:

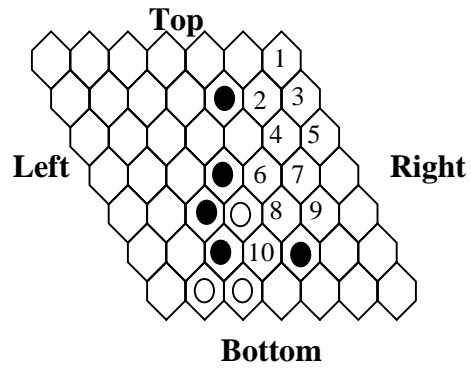


```

If (WhiteMove == 1 || 2 || 3 || 4 || 5) {
BlackMove = 6;
SumOfLocalGames = SumOfLocalGames -
LocalPattern30; }
else if (WhiteMove == 6) {
BlackMove = 5;
SumOfLocalGames = SumOfLocalGames - LocalPattern30
+ LocalPattern25(1,2,3,4); }

```

Local Pattern 31:

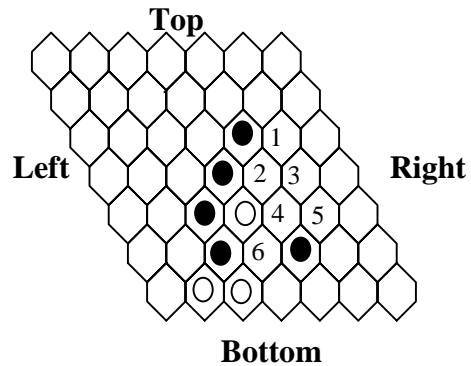


```

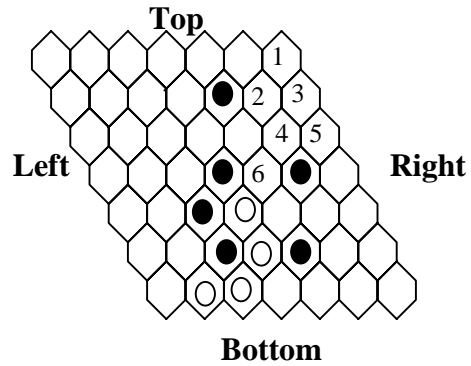
If (WhiteMove == 1 || 2 || 3 || 4 || 5 || 6 || 7 || 8 || 9) {
BlackMove = 10;
SumOfLocalGames = SumOfLocalGames -
LocalPattern31; }
else if (WhiteMove == 10) {
BlackMove = 7;
SumOfLocalGames = SumOfLocalGames - LocalPattern31
+ LocalPattern2(8,9) + LocalPattern32(1,2,3,4,5,6); }

```

Local Pattern 32(a):



Local Pattern 32(b):

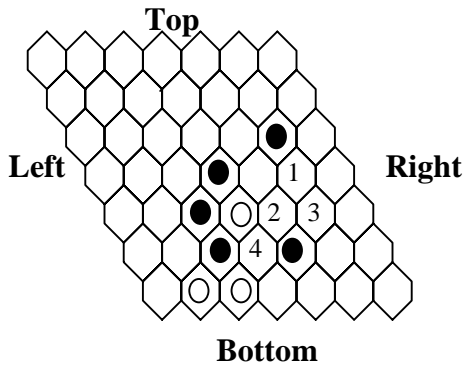


```

If (WhiteMove == 1 || 2 || 3 || 4 || 5) {
BlackMove = 6;
SumOfLocalGames = SumOfLocalGames -
LocalPattern32; }
else if (WhiteMove == 6) {
BlackMove = 3;
SumOfLocalGames = SumOfLocalGames - LocalPattern32
+ LocalPattern2(1,2) + LocalPattern2(4,5); }

```

Local Pattern 33:

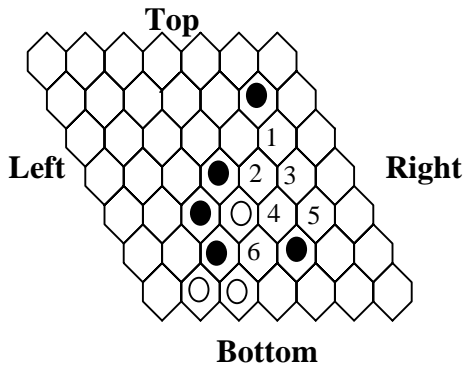


```

If (WhiteMove == 1 || 2 || 3) {
  BlackMove = 4;
  SumOfLocalGames = SumOfLocalGames -
  LocalPattern33; }
else if (WhiteMove == 4) {
  BlackMove = 1;
  SumOfLocalGames = SumOfLocalGames - LocalPattern33
  + LocalPattern2(2,3); }

```

Local Pattern 34:

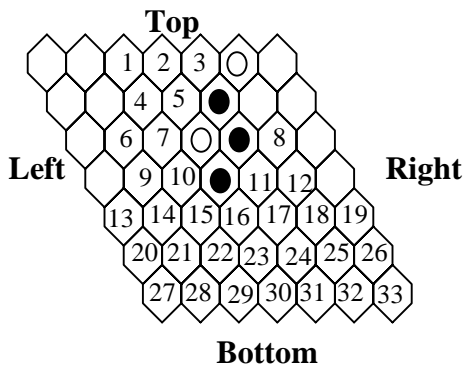


```

If (WhiteMove == 1 || 2 || 3 || 4 || 5) {
  BlackMove = 6;
  SumOfLocalGames = SumOfLocalGames -
  LocalPattern34; }
else if (WhiteMove == 6) {
  BlackMove = 3;
  SumOfLocalGames = SumOfLocalGames - LocalPattern34
  + LocalPattern2(1,2) + LocalPattern2(4,5); }

```

Local Pattern 35:

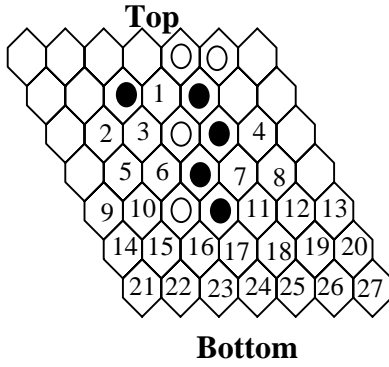


```

If (WhiteMove == 1 || 2 || 4 || 5 || 6 || 7 || 9 || 10 || 13 || 14 ||
20) {
  BlackMove = 3;
  SumOfLocalGames = SumOfLocalGames - LocalPattern35
  + LocalPattern5(8,11,12,15,16,17,18,21,22,23,24,25,27,28,
29,30,31,32); }
else if (WhiteMove == 8 || 11 || 12 || 17 || 18 || 19 || 23 || 24 ||
25 || 26 || 30 || 31 || 32 || 33) {
  BlackMove = 22;
  SumOfLocalGames = SumOfLocalGames - LocalPattern35
  + LocalPattern2(15,16) + LocalPattern2(28,29) +
  LocalPattern24(27,21,20,14,13,10,9,7,6,5,4,3,2,1); }
else if (WhiteMove == 21 || 27) {
  BlackMove = 15;
  SumOfLocalGames = SumOfLocalGames - LocalPattern35
  + LocalPattern26(14,10,9,7,6,5,4,3,2,1) +
  LocalPattern4(11,16,17,18,22,23,24,25,28,29,30,31,32); }
else if (WhiteMove == 28) {
  BlackMove = 14;
  SumOfLocalGames = SumOfLocalGames - LocalPattern35
  + LocalPattern2(10,15) +
  LocalPattern7(11,16,17,18,20,21,23,24,25,27,29,30,31,32)
  + LocalPattern27(9,7,6,5,4,3,2,1); }
else if (WhiteMove == 29) {
  BlackMove = 21;
  SumOfLocalGames = SumOfLocalGames - LocalPattern35
  + LocalPattern2(27,28) +
  LocalPattern9(15,16,17,22,23,24,30,31) +
  LocalPattern28(14,10,9,7,6,5,4,3,2,1); }
else if (WhiteMove == 22) {
  BlackMove = 15;
  SumOfLocalGames = SumOfLocalGames - LocalPattern35
  + LocalPattern26(14,10,9,7,6,5,4,3,2,1) +
  LocalPattern6(11,16,17,18,21,23,24,25,27,28,29,30,31,32);
}
else if (WhiteMove == 15) {
  BlackMove = 16;
  SumOfLocalGames = SumOfLocalGames - LocalPattern35
  + LocalPattern29(1,2,3,4,5,6,7,8,9,10,11,12,13,14,17,18,19,
20,21,22,23,24,25,26,27,28,29,30,31,32,33); }
else if (WhiteMove == 16) {
  BlackMove = 15;
  SumOfLocalGames = SumOfLocalGames - LocalPattern35
  + LocalPattern26(14,10,9,7,6,5,4,3,2,1) +
  LocalPattern16(8,11,12,17,18,21,22,23,24,27,28,29,30,31);
}
else if (WhiteMove == 3) {
  BlackMove = 4;
  SumOfLocalGames = SumOfLocalGames - LocalPattern35
  + LocalPattern2(1,2) +
  LocalPattern40(5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,
21,22,23,24,25,26,27,28,29,30,31,32,33); }

```

Local Pattern 36:

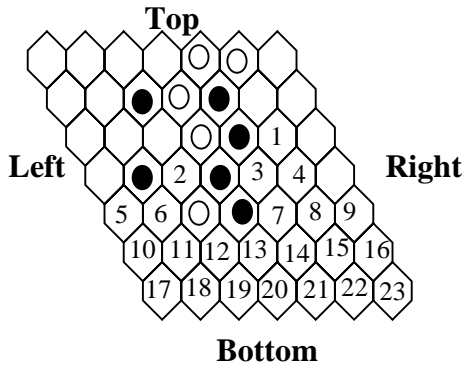


```

If (WhiteMove == 2 || 3 || 4 || 5 || 6 || 7 || 8 || 9 || 10 || 12 || 13
|| 14 || 15 || 19 || 20 || 21 || 26 || 27) {
BlackMove = 1;
SumOfLocalGames = SumOfLocalGames - LocalPattern36
+ LocalPattern13(11,16,17,18,22,23,24,25); }
else if (WhiteMove == 11 || 17 || 18 || 24 || 25) {
BlackMove = 16;
SumOfLocalGames = SumOfLocalGames - LocalPattern36
+ LocalPattern2(22,23) +
LocalPattern31(21,15,14,10,9,6,5,3,2,1); }
else if (WhiteMove == 16 || 22) {
BlackMove = 1;
SumOfLocalGames = SumOfLocalGames - LocalPattern36
+ LocalPattern22(4,7,8,11,12,13,17,18,19,20,23,24,25,26,
27); }
else if (WhiteMove == 23) {
BlackMove = 16;
SumOfLocalGames = SumOfLocalGames - LocalPattern36
+ LocalPattern23(11,17,18,22,24,25) +
LocalPattern31(21,15,14,10,9,6,5,3,2,1); }
else if (WhiteMove == 1) {
BlackMove = 5;
SumOfLocalGames = SumOfLocalGames - LocalPattern36
+ LocalPattern2(2,3) +
LocalPattern37(4,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,
21,22,23,24,25,26,27); }

```

Local Pattern 37:



```

If (WhiteMove == 1 || 3 || 4 || 5 || 6 || 8 || 9 || 10 || 11 || 15 ||
16 || 17 || 22 || 23) {
BlackMove = 2;
SumOfLocalGames = SumOfLocalGames - LocalPattern37
+ LocalPattern13(7,12,13,14,18,19,20,21); }

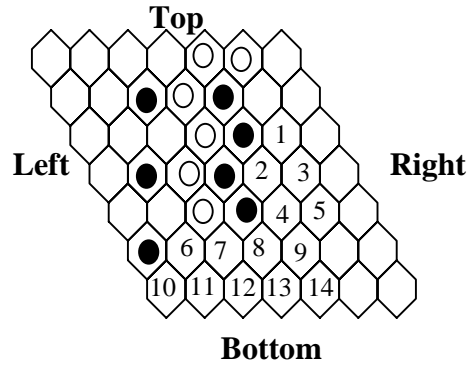
```

```

else if (WhiteMove == 7 || 13 || 14 || 20 || 21) {
BlackMove = 12;
SumOfLocalGames = SumOfLocalGames - LocalPattern37
+ LocalPattern2(18,19) + LocalPattern32(17,11,10,6,5,2); }
else if (WhiteMove == 12 || 18) {
BlackMove = 2;
SumOfLocalGames = SumOfLocalGames - LocalPattern37
+ LocalPattern22(1,3,4,7,8,9,13,14,15,16,19,20,21,22,23); }
else if (WhiteMove == 19) {
BlackMove = 12;
SumOfLocalGames = SumOfLocalGames - LocalPattern37
+ LocalPattern32(17,11,10,6,5,2) +
LocalPattern23(7,13,14,18,20,21); }
else if (WhiteMove == 2) {
BlackMove = 10;
SumOfLocalGames = SumOfLocalGames - LocalPattern37
+ LocalPattern2(5,6) +
LocalPattern38(1,3,4,7,8,11,12,13,14,17,18,19,20,21); }

```

Local Pattern 38:

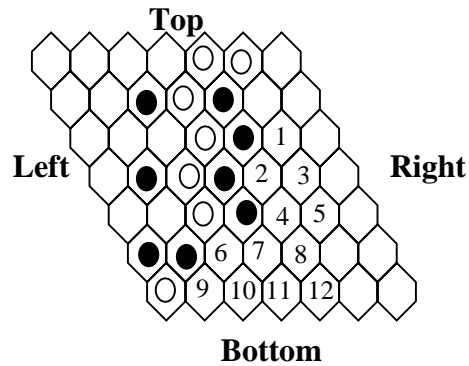


```

If (WhiteMove == 1 || 2 || 3 || 4 || 5 || 6 || 7 || 8 || 9 || 11 || 12 ||
13 || 14) {
BlackMove = 10;
SumOfLocalGames = SumOfLocalGames -
LocalPattern38; }
else if (WhiteMove == 10) {
BlackMove = 6;
SumOfLocalGames = SumOfLocalGames - LocalPattern38
+ LocalPattern39(1,2,3,4,5,7,8,9,11,12,13,14); }

```

Local Pattern 39:



```

If (WhiteMove == 1 || 2 || 3 || 4 || 5 || 6 || 7 || 8 || 10 || 11 || 12)
{
BlackMove = 9;
SumOfLocalGames = SumOfLocalGames -

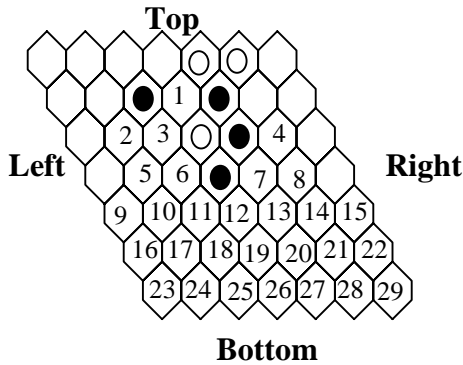
```

```

LocalPattern39; }
else if (WhiteMove == 9) {
BlackMove = 6;
SumOfLocalGames = SumOfLocalGames - LocalPattern39
+ LocalPattern26(1,2,3,4,5,7,8,10,11,12); }

```

Local Pattern 40:



```

If (WhiteMove == 2 || 3 || 5 || 6 || 9 || 10 || 16) {
BlackMove = 1;
SumOfLocalGames = SumOfLocalGames - LocalPattern40
+ LocalPattern5(4,7,8,11,12,13,14,17,18,19,20,21,23,24,25,
26,27,28); }
else if (WhiteMove == 4 || 7 || 8 || 13 || 14 || 15 || 19 || 20 || 21
|| 22 || 26 || 27 || 28 || 29) {
BlackMove = 18;
SumOfLocalGames = SumOfLocalGames - LocalPattern40
+ LocalPattern2(11,12) + LocalPattern2(24,25) +
LocalPattern31(23,17,16,10,9,6,5,3,2,1); }
else if (WhiteMove == 17 || 23) {
BlackMove = 11;
SumOfLocalGames = SumOfLocalGames - LocalPattern40
+ LocalPattern32(10,6,5,3,2,1) +
LocalPattern4(7,12,13,14,18,19,20,21,24,25,26,27,28); }
else if (WhiteMove == 24) {
BlackMove = 10;
SumOfLocalGames = SumOfLocalGames - LocalPattern40
+ LocalPattern2(6,11) +
LocalPattern7(7,12,13,14,16,17,19,20,21,23,25,26,27,28) +
LocalPattern33(5,3,2,1); }
else if (WhiteMove == 25) {
BlackMove = 17;
SumOfLocalGames = SumOfLocalGames - LocalPattern40
+ LocalPattern2(23,24) +
LocalPattern9(11,12,13,18,19,20,26,27) +
LocalPattern34(10,6,5,3,2,1); }
else if (WhiteMove == 18) {
BlackMove = 11;
SumOfLocalGames = SumOfLocalGames - LocalPattern40
+ LocalPattern32(10,6,5,3,2,1) +
LocalPattern6(7,12,13,14,17,19,20,21,23,24,25,26,27,28); }
else if (WhiteMove == 11) {
BlackMove = 12;
SumOfLocalGames = SumOfLocalGames - LocalPattern40
+ LocalPattern36(1,2,3,4,5,6,7,8,9,10,13,14,15,16,17,18,19,
20,21,22,23,24,25,26,27,28,29); }
else if (WhiteMove == 12) {
BlackMove = 11;

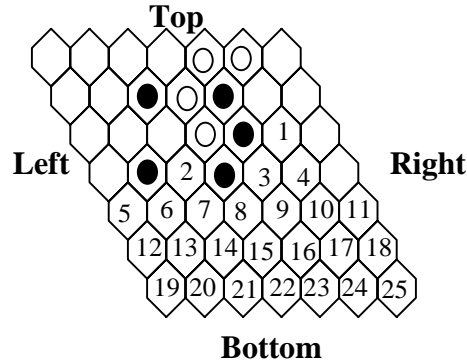
```

```

SumOfLocalGames = SumOfLocalGames - LocalPattern40
+ LocalPattern32(10,6,5,3,2,1) +
LocalPattern16(4,7,8,13,14,17,18,19,20,23,24,25,26,27); }
else if (WhiteMove == 1) {
BlackMove = 5;
SumOfLocalGames = SumOfLocalGames - LocalPattern40
+ LocalPattern2(2,3),
LocalPattern41(4,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,
21,22,23,24,25,26,27,28,29); }

```

Local Pattern 41:



```

If (WhiteMove == 5 || 6 || 12) {
BlackMove = 2;
SumOfLocalGames = SumOfLocalGames - LocalPattern41
+ LocalPattern5(1,3,4,7,8,9,10,13,14,15,16,17,19,20,21,22,
23,24); }
else if (WhiteMove == 1 || 3 || 4 || 9 || 10 || 11 || 15 || 16 || 17
|| 18 || 22 || 23 || 24 || 25) {
BlackMove = 14;
SumOfLocalGames = SumOfLocalGames - LocalPattern41
+ LocalPattern2(7,8) + LocalPattern2(20,21) +
LocalPattern32(19,13,12,6,5,2); }
else if (WhiteMove == 13 || 19) {
BlackMove = 7;
SumOfLocalGames = SumOfLocalGames - LocalPattern41
+ LocalPattern2(6,2) +
LocalPattern4(3,8,9,10,14,15,16,17,20,21,22,23,24); }
else if (WhiteMove == 20) {
BlackMove = 6;
SumOfLocalGames = SumOfLocalGames - LocalPattern41
+ LocalPattern2(2,7) +
LocalPattern7(3,8,9,10,12,13,15,16,17,19,21,22,23,24); }
else if (WhiteMove == 21) {
BlackMove = 13;
SumOfLocalGames = SumOfLocalGames - LocalPattern41
+ LocalPattern2(19,20) + LocalPattern2(2,6) +
LocalPattern9(7,8,9,14,15,16,22,23); }
else if (WhiteMove == 14) {
BlackMove = 7;
SumOfLocalGames = SumOfLocalGames - LocalPattern41
+ LocalPattern2(2,6) +
LocalPattern6(3,8,9,10,13,15,16,17,19,20,21,22,23,24); }
else if (WhiteMove == 7) {
BlackMove = 8;
SumOfLocalGames = SumOfLocalGames - LocalPattern41
+ LocalPattern37(1,2,3,4,5,6,9,10,11,12,13,14,15,16,17,18,
19,20,21,22,23,24,25); }

```

```

else if (WhiteMove == 8) {
BlackMove = 7;
SumOfLocalGames = SumOfLocalGames - LocalPattern41
+ LocalPattern2(2,6) +
LocalPattern16(1,3,4,9,10,13,14,15,16,19,20,21,22,23); }
else if (WhiteMove == 2) {
BlackMove = 12;
SumOfLocalGames = SumOfLocalGames - LocalPattern41
+ LocalPattern2(5,6) + LocalPattern2(7,8) +
LocalPattern38(1,3,4,9,10,13,14,15,16,19,20,21,22,23); }

```

IV. CONCLUSION

In this paper, we use heuristics to decompose the 7×7 Hex board into a sum of disjunctive small "local games" because moves made in one part of the positions do not often affect that of other parts. Based on this method, we found the winning strategy on 7×7 Hex board. According to [5], this is the largest Hex board for which a winning strategy is actually known.

We believe that the decomposition method developed in this research can be used to find winning strategies for larger Hex boards and extended to explore other combination explosion problems in computer games.

V. REFERENCES

- [1] L. Victor Allis, "Searching for Solutions in Games and Artificial Intelligence", 1995.
- [2] Cameron Browne, "Hex Strategy: Making the Right Connections", 2000.
- [3] J. H. Conway, "On Numbers and Games", 1976.
- [4] Martin Muller, "Computer Go as a Sum of Local Games: An Application of Combinatorial Game Theory", Ph.D. Dissertation, 1995.
- [5] Ian Stewart, "Hex Marks the Spot", Scientific American, Sept. 2000.