Vivekanand Education Society's Institute of Technology

Department of Computer Engineering

Subject: AI

Class :- CMPN Semester:-6 Div :- D12A

Roll No: 08	Name: Varnit Batheja		
Exp No: 08	Title: To implement an adversarial searching algorithm.		
DOP:	24-03-2022	DOS:	31-03-2022
GRADE:	LAB OUTCOMES:	SIGNATURE:	

AI EXPERIMENT NO: 8

AIM:
Implement an adversial searching algorithm.

THEORY:
Terminology

Game Tree:
It is a structure in the form at a tree consisting of all possible moves which allow you to move from state of game to next state

A game can be defined as a search problem with following components

Initial state:
It comprises the position of the board and showing whose move it is.

Successor function:
It defines what the legal moves a player can make are

Terminal State
It is the position of board when the game gets over

Utility function
It is a function which assigns a numeric value for the outcome of a game. For instance in chess or

Teacher's Sign. _____

tic-tac-toe the outcome is either a win a loss or a draw and these can be represented by values +1 -1 or 0 respectively. There are games that have a much larger range of possible outcomes for instance the utilities in backgroound varies from +192 to -192. A utility function can also be called play of function.

Alpha:
Alpha is the best choice or the highest value that we have found at any instance along the path of maximize the initial value of alpha is $-\infty$

Beta:
Beta is the best choice or the lowest value that we have found at any instance along path of minimizer. The initial value for beta is $+\infty$

Condition for alpha Beta pruning is $\alpha \geq \beta$

Each node has to keep track of it's alpha and Beta values Alpha can be updated only when it's max's turn and similarly ther can be updated when it's Min's chance

- Max will update only alpha values and Min player will update only beta value
- Node values will be passed to upper nodes instead of values of alpha and beta during go into reverse of tree.

Teacher's Sign:

Page No.
Date

- Alpha and Beta values only be passed to child nodes

Advantages :-

1) Allows elimination of the search tree branches
2) Limited the search time to more promising sub trees which enables a deeper search
3) Reduces computation and searching during the minmax algorithm
4) Prevents the use of additional computational time making the process more responsive and fast

Disadvantage

1) It does not save all the problem associated with original minimax algorithm
2) Requires a set depth limit as in most cases it is not feasible to search the entire game tree.
3) Though designed to calculate the good move it also calculates the value of all the legal moves

CONCLUSION

In this experiment We have successfully implemented an adversial searching algorithm alpha beta pruning

Teacher's Sign : _____

EXPERIMENT-08

Code:

```
MAX, MIN = 1000, -1000


def minimax(depth, nodeIndex, maximizingPlayer, values):
    if depth == 2:
        return values[nodeIndex]

    if maximizingPlayer:
        best = MIN
        for i in range(0, 2):
            val = minimax(depth + 1, nodeIndex * 2 + i, False, values)
            best = max(best, val)
        return best
    else:
        best = MAX
        for i in range(0, 2):
            val = minimax(depth + 1, nodeIndex * 2 + i, True, values)
            best = min(best, val)
        return best


if __name__ == "__main__":
    values = [3,5,10,2,8,19,2,7,3]
    print("The optimal value is :", minimax(0, 0, True, values))
```

Output:

```
The optimal value is : 3


...Program finished with exit code 0
Press ENTER to exit console.
```