ExpNo: 05
Pate;

MinMax Algorithm.

(a) aithy look

oleg neighbour (note).

shot : Kix

10.0) = 100/2

1012-De Abon

(Filed) times

- tugting

Aim:

To implement winner algorithm problem Using Python.

Source code:

from math import infas infinity
from random import choice
import platform

Import time

from os import system

HUMAN=-1

COMP 2+1

board = [C0,010], [01010], [01010]]

def craliate (state):

if wins (state, (OMP): 2) (12)

Score+1

elif wins (state, HUMAN).

Score = -1

Thus the At search program is issued the

sure so well a verified a tique

Letter score

def wins (State, player):

wm_state = [

Cotate CoJCo7, state CoJCi7, state CoJC2JJ,

Cotate [2](07, state CoJCi7, state CiJC2JJ,

Cotate [2](07, state CoJCi), state CoJCi),

Cotate CoJCo7, state CiJCo1, state CiJCoJ,

Cotate CoJCi , state CiJCo1, state CiJCij

Cotate CoJCi , state CiJCi , state CiJCij

Cotate CoJCi , state CiJCij

Cotate CoJCi , state CiJCi , state CiJCi , state CiJCij

Cotate CoJCi , state CiJCi , stat

```
if [player, player, player) in win_state:
    doff. In (empty-cells love)): BUST news
         if depth on spane over (loand):
    return FALSE
                                   o always.
 der game-over (state):
     letter win (state, HUMAN) or wins (state, comp)
 def empty-cells ( state):
       Cells = [7
   for x, now in enumerate ( state):
       for y, cell in enumerate (now):
         if cell = : 0;
             cells append ((x,y))
        Roturn celle.
    def valid_more (niy):
       Pf [n.y] in empty-cells (board):
             Settun TRUE
              letter FALSE
       des render ( state, (-chaire, h-chaire):
           chars= &
              -1: h-choice
             H: c-Choice
              0;11
                                              77023以
   Lus Litings phopped and sull'
          print( 'In" + Atribine) was trybus all
          for row in state:
              for cell in row:
                   Symbol = chars [cells ]
                  print(f1/ ssymboly 11, end="")
                  pant ("In "+ sh_line)
```

```
def ou-turn (c-choise h-choise):
    depth - len (empty-cells (board)):
       if depth . = o or game-over (board):
                                   grant muise
         setur o
                         : ( state ) 1900 - sinop per
          clean ()
  letter win (state, numan) or who (state, court)
                             def emply relle ( state):
                                   rethand 1
                 for or stow in commonate ( state):
                dos y, cell in enumerate (new):
                          is allos of
                cells offered (cary)
                              Rotun celle.
                        del valid-more (nit);
             of (or y) in empty -cells (board):
                           Busy ands.
                        eluc:
Robun FALSE
       def render ( Alba, rechard, heddie):
                              3 = 210Hs
                         prodot 1
                         H: c choice
RESULT:
   Thus the program was successfully executed and
the output was verified.
                      for row in state:
                   : wor is not
      Symbol chara Frelle 1
   (" " = tore, " Pladings ( 17) tring
        (and the 1.1 at ) thing
```