```
nege = 21801A0524 (Automotive)
code :-
from contributs impost default dict, deque
der bis (graph, bears, godg):
      vibiled + seech
      queue = sequeue([(staxt, (staxt)])
      while queue:
           note, Path + queve poplette
           if node == goal:
                  selvan path
            if node not in visited:
                   visited . add (nodes)
                   for neighbor in graph[node]:
                         queue . a ppend (theighboux, path . (ne i ghood))
      *cturn None
graph - defaultdict (list)
num-nodes = int(input("enter the number of nodes: "))
num-edges - int (input ("enter the number of edges:"))
for - in range (num-edges):
        U, V = input( Enter edge (U V): "). splif()
        graph [u]. append(v)
        groph [v]. append (v)
nodes = 1161 (graph keys (3)
paint (" Available nodes : ", nodes)
Start-node = input( enter the start node: ")
goal node = inpull enjet the goal node:")
il start note not in nodes or goal node not in nodes:
         Paint L' Invalid nodes (entered.")
e18e :
      Path = bps (graph, start node, goal node)
      if Path:
            print ( Path from ", start-node, "to", goal-node, ";
                    · join (Path)).
      eise :
            print (" No path found from", start-node, "to", goal-node)
```

Path

grom

```
OULPUL!-
                          nodes: 25
                     20
             numbes
        the
 ENLED
                         edges: 27
                     20
            number
Gnt 69
       the
       (2) (U V): A B
enles
       edge (0 V) : A
enter
      eage LUVJ: B
                       D
enter
      eage (U V): B 6
ence
      edge (UV); C
enves
      edge (U V) : C
                      G
enles
      edge LO NJ : D
enter
      eage (0 v) ; D
cruer
      edge (U V)
                       J
coulce
      edge (UV) ; E
erves
      edge (UV): F
enter
      edge (U V); F
                      M
erves
      edge (UV): G
                       N
EMOR
                       0
      edge (UV): G
entor
                       ρ
      edge (U V): H
Envor
      edge (U V) : H
                       Q
enter
      edge (UV) : I
                        R
GNION
      edge (UV): I
                       9
enles
     edge (UV): J
                       T
enen
      edge (UV): J
                       U
entes
          (U V) : K
                       V
      edge
entes
      edge (UV): K
                       W
entes
                        X
            (U V): L
      edge
                        У
enter
            (U U) : L
      eage
            (U V): M
enter
                        X
      6906
           (U V); ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', '5', 'k',
enler
      edge
Gruen
      6986
                     `L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V',
enter
Available
                     'w', 'x', 'Y']
            node : A
e mer
      Start
             node: V
       9001
enter
             A to Y: A->(-)F-)L->Y
```

node : 2

entered

9004

node

trivatio

```
nodes: 25
                  08
      the number
enter
                     edges: 21
                  10
     the number
Enles
                     B
      edge (U V): A
Inter
     edge (U V): A C
egues
     EAGE (D V): B
nies
                     E
      edge (UV): B
ner
                     F
     edge (U V); (
rues
     edge (U V); C
nuer
                     H
     egde (n n) : D
=nue2
                      I
     eage (U V) : D
enter
      eage (U V) : E
                      5
entor
     edge (UV) : E
ener
     edge (UV): F
GNYOR
      edge (U V): F
                      M
enter
      edge (UV):6
                      N
enter
      edge (UV) : G
                      0
6ryer
      edge (U V): H
                       P
Enler
      edge LU V): H
                      (2)
enter
      6986 (n n) ? I
                       R
e nuer
      edge Cuvi: I
                       5
errer
      edge (U V) : 5
                       T
Enter
       edge (UV): J
                       U
ENTER
       eage (D V): K
enter
       edge (U V) : K
GILLER
       edge CU V) : L
                        ×
enter
       edge (UV) ; L
 enter
       6986 (n n) : W
                        X
 enver
       edge (UV);
 ence
 Available nodes : ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J'
            (UU)
 enter
                       עי , יאי , יאי , יאי , יטי
  enter
         Start node: A
  G 1402
```

```
VASIREDDY VENKATADRI INSTITUTE OF TECHNOLOGY VVIT
  Regd. No. 218 GIAO524 (Autonomous)
c08e'-
from collections import defaultice
der bis (graph, current-node, goal-node, visited, path):
    if cussent node == goal node:
           relush path
     visited. ad (cossent_node)
     for neighbor in graph[cussery.node]:
               neighbox not in visited:
                   new-path = dfs (graph, neighbor, goal-node, visited, path+ [neighbor]
            18
                    if new-path:
                           retion new-path
      return None
grouph . defoult dick (list)
num-nodes intlinguil' Enter the number of nodes: "))
num. edges = in (input ('Enter the number of edges; "))
 for _ in ronge (num - edges):
          U,V = input ( Enter egge (0 N): ). Split()
           graph [v]. append (v)
           graph[v] append (v)
 nodes = list (gooph, keys())
 Pairu l' Available nodes; ", nodes)
 Start. node = input (" Enter the start node ; ")
 goal node : inpul ("Enled the goal node;")
 if Start node not in nodes or goal node not in nodes:
           Pairu ('Invalid nodes envesed')
 eise :
       Path = 295 (9 80Ph, Stort pode, Mep, goal-node, sell), [start-node])
```

prirul Path from, slart node, "(0", goal node, ","," -> "soin (Path))

Print ( No Path found from ", start node, 'to", goal node)

if path:

eise:

Path

from

FO

```
Regd. No. 21801A0521
 CULPUL!
                       rodes:29
                   OF
           number
      the
cares
                        edges : 21
                    00
           nomber
      ne
ENTOR
       eage (U V): A
                       B
cruor
       CHGE (U V): A
ENLES
       edge (U V); B
                       0
eneo
      edge (UV) : B
                       Ŀ
enles
       eage (U V) : C
colle o
       eage (U V) : C
erven
      edge (U V) : D
enter
      edge (U V) ; D
                       I
enter
      edge LU V) : E
                       J
enter
       edge (U V) ; E
                       K
enteo
       edge (UV): F
enter
       edge (UV) : F
                       M
enter
       edge (U V): G
Eules
       edge (UV): G
                        O
entes
      eage (U V) : 14
                        P
enten
       eage (U V); 17
                        Q
eneo
       edge (UV): I
                        R
enter
       edge (V V): I
                        5
erver
       edge (UV): 5
                       T
enter
       edge (UV): J
                       U
entor
       edge (UV) : X
erver
      edge (UU): K
enter
       edge (U V) : L
                       X
ener
       edge (U V) : L
enter
       edge (U V): M
enter
       eage (U V): M
enter
         nodes: ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J',
       edge
envor
Avoilable
                  'L', 'M', 'N', 'O', 'P', 'G', 'R', 'S',
                  יעי, 'שי, 'א', 'צ']
                   node : A
             Start
enter
       the
                   node: V
             9001
enter
       the
                   V: A->C->F->L->X->M->Y
```

```
nodes: 25
 Criter the number
                   of
                         Cd968:27
                    of
       the number
 cates
      6936 (n n) : V
 colle
 enter eage to vo : A
                       0
       eage (U V): B
 entes
      eage (U V); B
                       E
 entes
       eage (0 v): c
ences
      696 (0 N) : C
 enter
      edge (U V) : D
                       H
enles
       edge (U V) : D
enter
enter eage LU V) : E
enter edge LU VJ : E
                       K
enter edge (U V): F
      edge (U V) : F
Enver
      edge (UV): G
- ruca
Enler edge (UV):G
Enter edge (UV): H
enter edge (UV): H
enter edge (D V): I
enven edge (v v): I
enter edge (UV): J
                     T
enver eage (v v): 5
                     U
ENTER CAGE LU V): K
enter edge (b v): 12
enter edge (U U); L
enuer edge (UV); L
ener edge (0 v); M
errer edge (UV): M
cintex edge (UV): N
                                               16', 14', 17', 75', 1k', 1L',
Avoilable nodes: ['A', 18', 16', 10', 1et
                                      1a', 'R', 'S', IT', 'U', 'V',
```

enter the start node: A cruen the good node: Z

Invalid nodes energed

```
VASIREDDY VENKATADRI INSTITUTE OF TECHNOLOGY WIT
   Regd. No. 21BQ1A0524 (Autonomous)
Code : -
impost sandom
des Tictactoe ();
        Print ("Welcome Lo TicTacToe game!")
       del baintBoord ():
               Psini ('', boasd[2], "1", boasd[2])
               Point (" --- 1 --- ")
               Print (1, poara[3], "1", poora[1], "1", poara[2])
               Print(" --- 1 --- ")
               briuf (,, poory[e]' "1" ' poory[1] ' 1," poory[3])
               pxiru ()
       def GelRowld:
                    YOW = inpull choose you (1,2,3): ")
              while True:
                     if sow in ['1', '2', '3']:
                              בפנטשח וער (אסמי)
                            Printl'In Invalid input. Choose sow between 1 and 3.7
                       e186:
            Gelcolomn():
       def
               while True !
                     column : inpull'choose column(1,2,3):")
                      if column in ['1', '2', '3']:
                                setusa inicionan)
                           printl. In Invalid inhal choose colomu permen 1 and 3.7
                      C196 :
      des GelNumber ():
               while Txue:
                        number = inpulls
                        184
                            number = in(number)
                             if wamper in roude (1, 10);
                                     seluso number
                             eise :
                                  printin Number not in board?
                      except voice evice;
                                   Print ( 'In that's not a number . 134 again )
                                   continue
```

```
Ser Boorg Copy ( po ass);
de 8
         dupeBoasd :[]
         fox i in board:
                  dupe Board append(i)
          reform gobe boorg
der
      is space Face ( board , move):
           scrosu poasqi[more-i] == ,,
      makemove ( boasd ), lettes, move):
336
             boasd , [move - 1] + letter
      choose Random Move From List (board, moves List):
 126
           Possible Moves + []
           for i in movestist:
                    ie isspace Free (board, i):
                              Possible Moves. append(i)
            if lentpossible moves) ! = 0 :
                      yeluso sandom. Choice ( possi ble moves)
             eise :
                   selvan None
        computer choice ():
  des
            Pos: in songe (1,10):
                      COPY = ge1800xd(opy (600xd)
                       if isspace Free ( copy, i):
                                 make Move (copy, 'o', i):
                                  if checkwin (copy, 'o'):
                                          setazu:
             for : in songe (1, 10):
                       coba = defBoorg coba (pooga)
                        if is space Free (copy, i):
                                   makemove (copy, 'x', i)
                                       Checkwin (copy, 'x'):
                                              selusn i
                 is space Free (boord, 5):
              12
                          veturn 5
                          move = chooseRandom Move From List (board, [1,3,7,9])
                          if move; = none:
                                   reform move
              return choose Rondom Move From List ( 50 as d, [2,4,6,1])
```

```
der Turn(player):
      1 - ( ) radmonia . Gernomber ) - 1
      if poass[placing_intex] . . , x, on poass[blacing_intex] ... of
              Print('In Box already occupied Try another one")
              Turn (Player)
        cise .
              boasd[Placing_index] = Player
del Toxnilmoves:
         poozs[more -1] = ,0,
 get checkmin (poozgi bladez);
         for x in range(9):
                Por y in range (9):
                     for z in range(a):
                          if X != y and Y!= 2 and Z!= x:
                                  if poorqi[x] = = blader and poorqi[x]
                                  = = blades and poosqi[s] = = blades:
                                          if magic square [x] + Magic square
                                           [y] + Magicsquase = 15 :
                                                   BURL LARD
 der isboasdfuil(boasd):
         COURT = 0
          for a in sange (9):
               it poass(a) = = "x" or poass(a) = = .0.;
                         court +=1
          if count ==9:
                  Prince the game ends in a tie (n')
                  return True
   Print (" who plays first? (c for computer) It for Human):")
    fixst_ player = input(). upper()
    while True:
           Magicsquare = [4,9,2,3,5,7,8,1,6]
            if fixst_player == 'C':
                   while Truc;
                          Print Board()
                           end , checkwin (board, "O")
```

```
if end:
      print ( computer wins the gome )
      break
 eise:
       ie is Boox d Full (booxd);
              PAGOK
 move = compute & choice ()
  TUSD I (MOVE)
  Dring BOOSE ()
   end = checkwin( booxd, "X")
    if end:
          Paint ( you win the game")
           break
      cise:
           if is Boord Full (poard):
                     break
       princt" choose a box player x")
        if poary[ (xom-1) *3 + (olomu-1] = = "X" or poary[(xom-1) *
       row = Get Row()
          3 + (010mn - 1] == "0";
                     Print(10 Box already occupied try another one )
                600xd[(x0w-1) *3 + column-1] = "X"
          eise:
elif fixst player == 'H':
          while True:
                 printBoord()
                 SOW = GGROW()
                  column = Get (olumni)
                  if poosy[(xom -1) *3 + column -1] = = "x" ox poosy[(
                   80ω - 1) ×3 + column -1] == "0":
                              PSITUL("In Box alsearly occupied. Try another one")
                               conkinue.
                     erse:
                          poor of [(xom - 1) *3 + colomn -1] = "X"
                     end = Checkwin (boosd, "Y")
                     if end:
                            Print (" you win the game")
                           pacor.
```

```
: 5815
           if is Boosdfull boosd):
                  baeak
      move = computerchaice()
       rush (move)
       PrintBoard()
       end , checkwin (booxd, "O")
        if end:
               print ( computer wing the game )
               break
          e15e:
                if is Boox Full (boosd):
Play again = input("Do you wont to play again? (YIN):") . upport)
                       bscak
if play again ! = 'y':
        bxcak
```

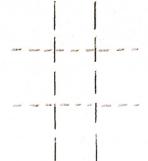
TicTacToeLs

OU FLOF: -

welcome to TicTocToe game!

muo blade tikef (C tok combatek) it tok Hamon):

H



choose sow (1,2,3):1

choose column(1,2,3):1

Choose 800 (1,2,3); 1

choose column(1,2,3):2

Choose 8000 (1,2,3):3

choose column (1,2,3):1

0101

Choose 8000 (1,2,3):2 Choose column (1,2,3);3

Choose 8000 (1,2,3);3 Choose column (1,2,3):2

the game ends in a tie

Do you work to Play again (YIN): y

who plays fixel? (c fox computer [H fox Human):

choose column (1,2,3):1

X -----

choose column(1,2,3): 1

X 0 0

X1010

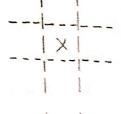
choose sow (1,2,3):3

bo you want to play again (VIN): N

-----

choose 80w(1,2,3):2

choose column(1,2,3): 2

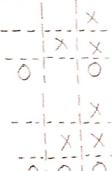


Choose 3000 (1,2,3):1

choose column (1,2,3):3

choose 80w(1,2,3):2

Chaose column (1,2,3):3



01010

Computer wins the game Bo you want to Play again? (YIN): N