Page No.
Date

Sumanth.K.V.
AI lab test-1
1BM18CS112
09/11/2020

Q) Implement IDS algorithm using suitable heuristic function where d should not be greater than 4

Sol:-

#code.

```
def dfs(src, target, visited_states, limit):
        if src == target
            return True

        if limit == 0
            return False

        visited_states.append(src)
        poss_moves = []
         poss_moves = possible_states(src,
                                   visited_states);
        all_cost = [], index=0
        for move in poss_moves:
            all_cost[index] = h(move, target);
            index = index+1
        poss = [], ~~index=0~~ index=0
        min_cost = min(all_cost)
        for move in poss_moves:
            ~~poss[index~~ if allcost[index] == min_cost
            poss.append(move)
            index += 1
```

(1) Sumanth.K.V.

Page No.
Date

```
for move in   poss:
    if (dfs(move, target, visited_states, limit-1))
            return True.

~~Fa~~ return False

def possible_states ( src, visited_states):
    blank= src.index (-1)

    d=[]

    if blank ~~is~~ not in [0,1,2]:
        d.append('u')

    if blank not in [6,7,8];
        d.append('d')

    if blank not in [0,3,6];
        d.append('l')

    if blank not in [2,5,8];
        d.append('r')

    all_states=[]
    for i in d:
        all_states.append (get_possible (src, i,blank))

    return [move for move in all_states if
        move not in visited_states]
```

©Sumanth.kv.

Page No.
Date

```
def get_possible (src, action, blank):

 temp = src.copy()
 if action=='u':
   temp[blank], temp[blank-3]=
          temp(blank-3], temp(blank]

 if action == 'r':
   temp[blank], temp(blank+1]=
     temp(blank+1], temp (blank]

 if action=='l':
   temp[blank], temp[blank-1]=
     temp[blank-1], temp[blank]

 if action=='d':
   temp(blank], temp (blank+3]=
   temp[blank+3], temp(blank]

 return temp
```

Ⓢ Sumanth.ku

Page No.
Date

```
if __name__ == "__main__":

        for i in range(4):
            if (dfs(src, target, i))
                return True.

    return False.
```

(4) Sumanth.KU

```
def h(src, tar):
        empty_slot = tar.index(-1)
        count=0

        for index in range(len(src)):
            if empty_slot != index and
                src[index] != tar[index]:
                count += 1

    return count
```

(4) Sumanth.KU