TT DS PYTHON MODULE-21

Started on	Friday, 16 May 2025, 8:51 AM
State	Finished
Completed on	Friday, 16 May 2025, 8:57 AM
Time taken	5 mins 22 secs
Grade	**80.00** out of 100.00

Question **1**
Correct
Mark 20.00 out of 20.00
Flag question

Write a python program to implement knight tour problem

For example:

Input	Result
5 5	[1, 12, 25, 18, 3] [22, 17, 2, 13, 24] [11, 8, 23, 4, 19] [16, 21, 6, 9, 14] [7, 10, 15, 20, 5] [(0, 0), (1, 2), (0, 4), (2, 3), (4, 4), (3, 2), (4, 0), (2, 1), (3, 3), (4, 1), (2, 0), (0, 1), (1, 3), (3, 4), (Done!

Answer: (penalty regime: 0 %)

Reset answer

```python
import sys
class KnightsTour:
    def __init__(self, width, height):
        self.w = width
        self.h = height
        self.board = []
        self.generate_board()

    def generate_board(self):
        for i in range(self.h):
            self.board.append([0]*self.w)

    def print_board(self):

        for elem in self.board:
            print (elem)

    def generate_legal_moves(self, cur_pos):
        possible_pos = []
        move_offsets = [(1, 2), (1, -2), (-1, 2), (-1, -2),
                        (2, 1), (2, -1), (-2, 1), (-2, -1)]

```

	Input	Expected
	5 5	[1, 12, 25, 18, 3] [22, 17, 2, 13, 24] [11, 8, 23, 4, 19] [16, 21, 6, 9, 14] [7, 10, 15, 20, 5] [(0, 0), (1, 2), (0, 4), (2, 3), (4, 4), (3, 2), (4, 0), (2, 1), (3, 3), (4, 1), (2, 0), (0, 1), (1, 3), (3, Done!
	6 6	[1, 32, 9, 18, 3, 34] [10, 19, 2, 33, 26, 17] [31, 8, 25, 16, 35, 4] [20, 11, 36, 27, 24, 15] [7, 30, 13, 22, 5, 28] [12, 21, 6, 29, 14, 23] [(0, 0), (1, 2), (0, 4), (2, 5), (4, 4), (5, 2), (4, 0), (2, 1), (0, 2), (1, 0), (3, 1), (5, 0), (4, 2), (5, Done!

Passed all tests!

Correct
Marks for this submission: 20.00/20.00.

Question **2**
Not answered
Mark 0.00 out of 20.00
Flag question

Write a Python program to print the following pattern based on the given input.

input:6

output:

```
*
**
***
****
```

```
*****
```

For example:

Input	Result
6	* ** *** **** *****

Answer: (penalty regime: 0 %)

```
```

Question **3**

Correct

Mark 20.00 out of 20.00

Flag question

Write a Python program for Bad Character Heuristic of Boyer Moore String Matching Algorithm

For example:

Input	Result
ABAAAABCD ABC	Pattern occur at shift = 5

Answer: (penalty regime: 0 %)

Reset answer

```python
NO_OF_CHARS = 256
def badCharHeuristic(string, size):
    ########### Add your Code Here ###############
    #Start here
    badChar = [-1]*NO_OF_CHARS
    for i in range(size):
        badChar[ord(string[i])] = i;
    return badChar
    #End here
def search(txt, pat):
    m = len(pat)
    n = len(txt)
    badChar = badCharHeuristic(pat, m)
    s = 0
    while(s <= n-m):
        j = m-1
        while j>=0 and pat[j] == txt[s+j]:
            j -= 1
        if j<0:
            print("Pattern occur at shift = {}".format(s))
            s += (m-badChar[ord(txt[s+m])] if s+m<n else 1)
        else:
```

	Input	Expected	Got	
	ABAAAABCD ABC	Pattern occur at shift = 5	Pattern occur at shift = 5	

Passed all tests!

Correct
Marks for this submission: 20.00/20.00.

Question **4**
Correct
Mark 20.00 out of 20.00
Flag question

Write a python program to implement pattern matching on the given string using Brute Force algorithm.

For example:

Test	Input	Result
BF(a1,a2)	abcaaaabbbbcccabcbabdbcsbbbbbnnn ccabcba	12

Answer: (penalty regime: 0 %)

Reset answer

```

def BF(s1,s2):
##############  Add your code here #############
    #Start here
    i = 0
    j = 0
    while(i < len(s1) and j < len(s2)):
        if(s1[i] ==  s2[j]):
            i += 1
            j += 1
        else:
            i = i - j + 1
            j = 0
    if(j >= len(s2)):
        return i - len(s2)
    else:
        return 0
    #End here
if __name__ == "__main__":
    a1=input()
    a2=input()
    b=BF(a1,a2)
```

	Test	Input	Expected	Got	
	BF(a1,a2)	abcaaaabbbbcccabcbabdbcsbbbbbnnn ccabcba	12	12	

Passed all tests!

Correct
Marks for this submission: 20.00/20.00.

Question **5**
Correct
Mark 20.00 out of 20.00
Flag question

Create a python program to implement Hamiltonian circuit problem using Backtracking.
For example:

Result
Solution Exists: Following is one Hamiltonian Cycle 0 1 2 4 3 0

Answer: (penalty regime: 0 %)

Reset answer

```
class Graph():
    def __init__(self, vertices):
        self.graph = [[0 for column in range(vertices)]
                            for row in range(vertices)]
        self.V = vertices
    def isSafe(self, v, pos, path):
        if self.graph[ path[pos-1] ][v] == 0:
            return False
        for vertex in path:
            if vertex == v:
                return False

        return True
    def hamCycleUtil(self, path, pos):
        #####################Add your code here###################################
        #Start here
        if pos == self.V:
            if self.graph[ path[pos-1] ][ path[0] ] == 1:
                return True
            else:
                return False
        for v in range(1,self.V):
```

	Expected	Got	
	Solution Exists: Following is one Hamiltonian Cycle 0 1 2 4 3 0	Solution Exists: Following is one Hamiltonian Cycle 0 1 2 4 3 0	

Passed all tests!

Correct

Marks for this submission: 20.00/20.00.

Finish