

TT DS PYTHON MODULE-21

Started on Tuesday, 24 September 2024, 8:25 AM**State** Finished**Completed on** Tuesday, 24 September 2024, 8:31 AM**Time taken** 6 mins 28 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 KMP (Knuth Morris Pratt).

For example:

Input	Result
ABABDABACDABABCABAB ABABCABAB	Found pattern at index 10

Answer: (penalty regime: 0 %)

Reset answer

```

1 def KMPSearch(pat, txt):
2     ##### Add your code here #####
3     #Start here
4     M = len(pat)
5     N = len(txt)
6     lps = [0]*M
7     j = 0
8     computeLPSArray(pat, M, lps)
9     i = 0
10    while (N - i) >= (M - j):
11        if pat[j] == txt[i]:
12            i += 1
13            j += 1
14        if j == M:
15            print ("Found pattern at index " + str(i-j))
16            j = lps[j-1]
17        elif i < N and pat[j] != txt[i]:
18            if j != 0:
19                j = lps[j-1]
20            else:
21                i += 1
22    #End here

```

	Input	Expected	Got	
	ABABDABACDABABCABAB ABABCABAB	Found pattern at index 10	Found pattern at index 10	
	SAVEETHAENGINEERING VEETHA	Found pattern at index 2	Found pattern at index 2	

Passed all tests!

Correct

Marks for this submission: 20.00/20.00.

Question 2

Correct

Mark 20.00 out of 20.00

Flag question

Write a python program to implement Boyer Moore Algorithm with Good Suffix heuristic to find pattern in given text string.

For example:

Input	Result
ABAAABAACD ABA	pattern occurs at shift = 0 pattern occurs at shift = 4

Answer: (penalty regime: 0 %)

Reset answer

```

1 def preprocess_strong_suffix(shift, bpos, pat, m):
2     ##### Add your Code here #####
3     #Start here
4     i = m
5     j = m + 1
6     bpos[i] = j
7     while i > 0:
8         while i < m and pat[i] != pat[i-1]:

```

```

8         while j <= m and pat[i - 1] != pat[j - 1]:
9             if shift[j] == 0:
10                 shift[j] = j - i
11                 j = bpos[j]
12             i -= 1
13             j -= 1
14         bpos[i] = j
15     #End here
16 def preprocess_case2(shift, bpos, pat, m):
17     j = bpos[0]
18     for i in range(m + 1):
19         if shift[i] == 0:
20             shift[i] = j
21         if i == j:
22             j = bpos[j]

```

	Input	Expected	Got	
	ABAAABAACD ABA	pattern occurs at shift = 0 pattern occurs at shift = 4	pattern occurs at shift = 0 pattern occurs at shift = 4	
	SaveethaEngineering Saveetha veetha	pattern occurs at shift = 2 pattern occurs at shift = 22	pattern occurs at shift = 2 pattern occurs at shift = 22	

Passed all tests!

Correct

Marks for this submission: 20.00/20.00.

Question 3

Not answered

Mark 0.00 out of 20.00

Flag question

Write a python to implement Quick sort using the first element as pivot value

For example:

Input	Result
5 61 24 3 50 8	Pivot: 61 Pivot: 8 Pivot: 24 Sorted array: [3, 8, 24, 50, 61]
6 2 3 54 10 28 94	Pivot: 2 Pivot: 3 Pivot: 54 Pivot: 28 Sorted array: [2, 3, 10, 28, 54, 94]

Answer: (penalty regime: 0 %)

1

Syntax Error(s)

File "__tester__.python3", line 1
 Write a python program to implement knight tour problem using warnsdorff's algorithm
 ^
 SyntaxError: invalid syntax

Incorrect

Marks for this submission: 0.00/20.00.

Question 4

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

```

1 class Graph():
2     def __init__(self, vertices):
3         self.graph = [[0 for column in range(vertices)]
4                       for row in range(vertices)]
5         self.V = vertices
6     def isSafe(self, v, pos, path):
7         if self.graph[ path[pos-1] ][v] == 0:
8             return False
9         for vertex in path:
10             if vertex == v:
11                 return False
12
13         return True
14     def hamCycleUtil(self, path, pos):
15         #####Add your code here#####
16         #Start here
17         if pos == self.V:
18             if self.graph[ path[pos-1] ][ path[0] ] == 1:
19                 return True
20             else:
21                 return False
22         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.

Question 5

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), (0, 4)] Done!

Answer: (penalty regime: 0 %)

Reset answer

```

1 import sys
2 class KnightsTour:
3     def __init__(self, width, height):
4         self.w = width
5         self.h = height
6         self.board = []
7         self.generate_board()
8
9     def generate_board(self):
10        for i in range(self.h):
11            self.board.append([0]*self.w)
12
13    def print_board(self):

```

Correct

Marks for this submission: 20.00/20.00.