Started on	Monday, 28 April 2025, 3:14 PM
State	Finished
Completed on	Monday, 28 April 2025, 3:33 PM
Time taken	19 mins 17 secs
Grade	**80.00** out of 100.00

Question **1**

Incorrect

Mark 0.00 out of 20.00

Write a Python Program to find minimum number of swaps required to sort an array given by the user.

For example:

Test	Input	Result
`minSwaps(arr)`	5 1 5 4 3 2	2
`minSwaps(arr)`	6 1 24 36 21 20 3	3

Answer: (penalty regime: 0 %)

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

```
def fun(int a){

}
```

Syntax Error(s)

```
  File "__tester__.python3", line 1
    def fun(int a){
                ^
SyntaxError: invalid syntax
```

Incorrect

Marks for this submission: 0.00/20.00.

Question **2**

Correct

Mark 20.00 out of 20.00

Write a python program to implement knight tour problem using warnsdorff's algorithm

For example:

Test	Input	Result
a.warnsdroff((x,y))	8 8 3 3	board: [21, 32, 17, 30, 39, 36, 15, 42] [18, 29, 20, 35, 16, 41, 54, 37] [33, 22, 31, 40, 53, 38, 43, 14] [28, 19, 34, 1, 44, 49, 60, 55] [23, 2, 27, 52, 61, 56, 13, 50] [8, 5, 24, 45, 48, 51, 62, 59] [3, 26, 7, 10, 57, 64, 47, 12] [6, 9, 4, 25, 46, 11, 58, 63]

Answer: (penalty regime: 0 %)

Reset answer

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

```
KNIGHT_MOVES = [(2, 1), (1, 2), (-1, 2), (-2, 1), (-2, -1), (-1, -2), (1, -2), (2, -1)]
class KnightTour:
    def __init__(self, board_size):
        self.board_size = board_size  # tuple
        self.board = []
        for i in range(board_size[0]):
            temp = []
            for j in range(board_size[1]):
                temp.append(0)
            self.board.append(temp) # empty cell
        self.move = 1

    def print_board(self):
        print('board:')
        for i in range(self.board_size[0]):
            print(self.board[i])

    def warnsdroff(self, start_pos, GUI=False):
```

	Test	Input	Expected	Got	
✓	a.warnsdroff((x,y))	8 8 3 3	board: [21, 32, 17, 30, 39, 36, 15, 42] [18, 29, 20, 35, 16, 41, 54, 37] [33, 22, 31, 40, 53, 38, 43, 14] [28, 19, 34, 1, 44, 49, 60, 55] [23, 2, 27, 52, 61, 56, 13, 50] [8, 5, 24, 45, 48, 51, 62, 59] [3, 26, 7, 10, 57, 64, 47, 12] [6, 9, 4, 25, 46, 11, 58, 63]	board: [21, 32, 17, 30, 39, 36, 15, 42] [18, 29, 20, 35, 16, 41, 54, 37] [33, 22, 31, 40, 53, 38, 43, 14] [28, 19, 34, 1, 44, 49, 60, 55] [23, 2, 27, 52, 61, 56, 13, 50] [8, 5, 24, 45, 48, 51, 62, 59] [3, 26, 7, 10, 57, 64, 47, 12] [6, 9, 4, 25, 46, 11, 58, 63]	✓

Passed all tests! ✓

Correct

Marks for this submission: 20.00/20.00.

Question **3**

Correct

Mark 20.00 out of 20.00

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

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

```
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
```

	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

Write a python program to check whether Hamiltonian path exits in the given graph.

For example:

Test	Result
Hamiltonian_path(adj, N)	YES

Answer: (penalty regime: 0 %)

Reset answer

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

```
def Hamiltonian_path(adj, N):
    dp = [[False for i in range(1 << N)] for j in range(N)]
    for i in range(N):
         dp[i][1 << i]=True
    for i in range(1 << N):
        for j in range(N):
            if ((i & (1 << j))!=0):
                for k in range(N):
                    if((i & (1 << k)) != 0 and
                            adj[k][j] == 1 and
                                    j != k and
                          dp[k][i ^ (1 << j)]):
                       dp[j][i]=True
                       break
    for i in range(N):
        if (dp[i][(1 << N)-1]):
            return True
    return False
```

	Test	Expected	Got	
✓	Hamiltonian_path(adj, N)	YES	YES	✓

Passed all tests! ✓

Correct

Marks for this submission: 20.00/20.00.

Question **5**

Correct

Mark 20.00 out of 20.00

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

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

```
def preprocess_strong_suffix(shift, bpos, pat, m):
    ###############  Add your Code here #####################
    #Start here
    i = m
    j = m + 1
    bpos[i] = j
    while i > 0:
        while j <= m and pat[i - 1] != pat[j - 1]:
            if shift[j] == 0:
                shift[j] = j - i
            j = bpos[j]
        i -= 1
        j -= 1
        bpos[i] = j
    #End here
def preprocess_case2(shift, bpos, pat, m):
    j = bpos[0]
    for i in range(m + 1):
```

	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.