

Started on Wednesday, 16 July 2025, 3:12 PM

State Finished

Completed on Wednesday, 16 July 2025, 3:29 PM

Time taken 17 mins 13 secs

Grade **80.00** out of 100.00

Question 1

Correct

Mark 20.00 out of 20.00

Develop a python program to add only the even unique numbers using appendleft() from n given numbers

For example:

Input	Result
5 2 5 8 2 4	deque([4, 8, 2])
6 3 5 2 8 2 5	deque([8, 2])

Answer: (penalty regime: 0 %)

```

1 | from collections import deque
2 | class Queue:
3 |     def __init__(self):
4 |         self.queue = deque()
5 |     def add_element(self,val):
6 |         if val%2==0 and val not in self.queue:
7 |             self.queue.appendleft(val)
8 |             return True
9 |         return False
10 | TheQueue = Queue()
11 | n=int(input())
12 | for i in range(n):
13 |     TheQueue.add_element(int(input()))
14 | print(TheQueue.queue)

```

	Input	Expected	Got	
✓	5 2 5 8 2 4	deque([4, 8, 2])	deque([4, 8, 2])	✓
✓	6 3 5 2 8 2 5	deque([8, 2])	deque([8, 2])	✓

Passed all tests! ✓

Correct

Marks for this submission: 20.00/20.00.

Question 2

Not answered

Mark 0.00 out of 20.00

Write a Python program to get the name, attendance and Id of a student and check they are eligible for Placement using multiple inheritance

Note: grade >90 eligible student else Not Eligible student

For example:

Input	Result
saveetha 21 98	saveetha 21 Eligible for Placement
sachin 22 71	sachin 22 Not Eligible for Placement

Answer: (penalty regime: 0 %)

1 ||

	Input	Expected	Got	
✓	saveetha 21 98	saveetha 21 Eligible for Placement	saveetha 21 Eligible for Placement	✓
✗	sachin 22 71	sachin 22 Not Eligible for Placement	saveetha 21 Eligible for Placement	✗

Your code must pass all tests to earn any marks. Try again.

Show differences

Incorrect

Marks for this submission: 0.00/20.00.

Question 3

Correct

Mark 20.00 out of 20.00

Write a python program to delete two neighboring non-identical letters(lower case and upper case) .

Example: AbBbA

lowercase b and uppercase B will get removed

For example:

Input	Result
leEeetcode	leetcode

Answer: (penalty regime: 0 %)

```

1 def makeGood(s):
2     stack = []
3     for i in s:
4         if stack and stack[-1] != i and stack[-1].lower() == i.lower():
5             stack.pop()
6         else:
7             stack.append(i)
8     return "".join(stack)
9
10
11 s = input()
12 print(makeGood(s))

```

	Input	Expected	Got	
✓	leEeetcode	leetcode	leetcode	✓

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 create a stack with a maximum size of 7 using Lifo Queue. Get the input from the user and check whether the stack is full and then display the stack values in reverse order

For example:

Input	Result
4 Maths Physics Chemistry Biology	False Biology Chemistry Physics Maths
7 Maths Physics Chemistry Biology History Economics English	True English Economics History Biology Chemistry Physics Maths

Answer: (penalty regime: 0 %)**Reset answer**

```

1 |from queue import LifoQueue
2 |stack = LifoQueue(maxsize=7)
3 |n= int(input())
4 |for i in range(n):
5 |    stack.put(input())
6 |print(stack.full())
7 |for i in range(n):
8 |    print(stack.get())

```

	Input	Expected	Got	
✓	4 Maths Physics Chemistry Biology	False Biology Chemistry Physics Maths	False Biology Chemistry Physics Maths	✓
✓	7 Maths Physics Chemistry Biology History Economics English	True English Economics History Biology Chemistry Physics Maths	True English Economics History Biology Chemistry Physics Maths	✓

Passed all tests! ✓

Correct

Marks for this submission: 20.00/20.00.

Question 5

Correct

Mark 20.00 out of 20.00

Develop a python program to get string values from the user and display the values using circular queue

For example:

Input	Result
4 Python Java C C++	Python Java C C++
5 Java C# C Python C++	Java C# C Python C++

Answer: (penalty regime: 0 %)

[Reset answer](#)

```

1 class MyCircularQueue():
2     def __init__(self, k):
3         self.k = k
4         self.queue = [None] * k
5         self.head = self.tail = -1
6     def enqueue(self, data):
7         if ((self.tail + 1) % self.k == self.head):
8             print("The circular queue is full\n")
9         elif (self.head == -1):
10             self.head = 0
11             self.tail = 0
12             self.queue[self.tail] = data
13         else:
14             self.tail = (self.tail + 1) % self.k
15             self.queue[self.tail] = data
16         #
17     def printCQueue(self):
18         if(self.head == -1):
19             print("No element in the circular queue")
20         elif (self.tail >= self.head):
21             for i in range(self.head, self.tail + 1):
22                 print(self.queue[i], end=" ")

```

	Input	Expected	Got	
✓	4 Python Java C C++	Python Java C C++	Python Java C C++	✓
✓	5 Java C# C Python C++	Java C# C Python C++	Java C# C Python C++	✓

Passed all tests! ✓

Correct

Marks for this submission: 20.00/20.00.

