

|  |  |  |
| --- | --- | --- |
| **Subject NAME** | **:** | **Data Structure & Algorithm LAb** |
| **DATE OF EXPERIMENT** | **:** | **25th October, 2021** |
| **Instructor NAME** |  | **Prof. Nadeem** |
| **NAME** | **:** | **Muhammad Ahmed** |
| **ROLL NO** | **:** | **200901101** |

**LAB TASK 1**

Implement queue with numpy:

CODE:

import numpy as np

class queue:

    def \_init\_ (self , size):

        self.size = size

        self.elements = np.empty([self.size])

        self.front=0

        self.rear = 0

        self.size = size

    def queue\_Enqueue(self,data):

        if (self.size == self.rear):

            return "Queue is full (Overflow)"

        else:

            self.elements[self.rear] = data

            self.rear = self.rear +1

# \

 def queue\_dequeue(self):

        if (self.front == self.rear):

            return "Queue is Empty (Underflow)"

        else:

            self.front = self.front + 1

    def queue\_display(self):

        if(self.front == self.rear):

            return "Queue is empty underflow "

        for x in self.elements:

            print(x)

    def queue\_Front(self):

        if(self.front == self.rear):

            print("Queue is Empty underflow")

        print("Front Element is:",

          self.queue[self.front])

q = queue()

q.queue\_dequeue()

q.queue\_Enqueue(83)

q.queue\_Enqueue(2)

q.queue\_Enqueue(23)

q.queue\_Enqueue(5)

LAB TASK 2:

from collection import deque

append(a)

-->> append(a) method adds an 'a' to the end of the list

appendleft(a)

-->> appendleft() function Add 'a' to the left side of the deque

clear()

-->> Remove all elements from the deque.

copy()

-->> Create a shallow copy of the deque

count(a)

-->>Count the number of deque elements equal to 'a'.

extend(iterable)

-->>Extend the right side of the deque by appending elements from the iterable argument.

extendleft(iterable)

-->>Extend the left side of the deque by appending elements from iterable.

index()

-->> Return the position of element in the deque

insert()

-->> Insert element into the deque at specific index.

maxlen()

-->> This funcion is used for finding Maximum size of a deque

pop()

-->>Remove an element from the right side of the deque. popleft()

-->>Remove an element from the left side of the deque.

remove()

-->>Remove the first occurrence of value.

reverse()

-->> This method is used for Reverse the elements of the deque.

rotate(n=1)

-->> Rotate the deque n steps to the right. If n is negative, rotate to the left