

23/02/26

Date \_\_\_\_\_  
Page \_\_\_\_\_

॥ गणपती वाप्या मोरया ॥

#### 4. deque (double-ended queue)

- It is a special data structure in Python that allows fast insertion and deletion from both ends.

from collections import deque

##### • Creation of deque

from collections import deque  
d = deque()

or

q = deque([1, 2, 3])

##### • Basic operations

###### i. Insert at back

O(1)

q.append(10)

Before : [ ]

After : [10]

###### ii. Insert at front

O(1)

q.appendleft(20)

o/p: [20, 10]

iii. Remove from back

`q.pop()`

$O(1)$

o/p: [20]

iv. Remove from front

`q.popleft()`

$O(1)$

o/p: []

Imp: deque is not good for random access.

access by index  $\Rightarrow O(n)$

• Where deque is used?

1. BFS (Breadth First Search)

from collections import deque

`q = deque()`

`q.append(start_node)`

`while q:`

`node = q.popleft()`

It is used in : graphs, trees, shortest path  
(unweighted)

## 2. Sliding Window problems

- (i) maximum of all subarrays of size k
- (ii) first -ve no. in window
- (iii) longest substring problems

## 3. Queue problems

- (i) process in order
- (ii) first come first serve
- (iii) level order traversal

## 4. Monotonic Queue (Advanced)

- (i) Sliding window maximum
- (ii) Next greater element variants
- (iii) Optimization problems

Imp: While a standard queue follows the FIFO principle, a Deque is more flexible power tool that allows  $O(1)$  insertions and deletions from both the front and back.