Data Structures
Queue – Types of Queue

Team Emertxe

Types of Queue

Types of Queue

Types

- Linear
- Circular

Types of Queue

Types

Linear

Circular

- In Linear Queue ,the data is arranged in Sequential manner
- In Circular Queue,the data is arranged in circular manner.

Linear Queue

- int arr[SIZE]

SIZE = 4

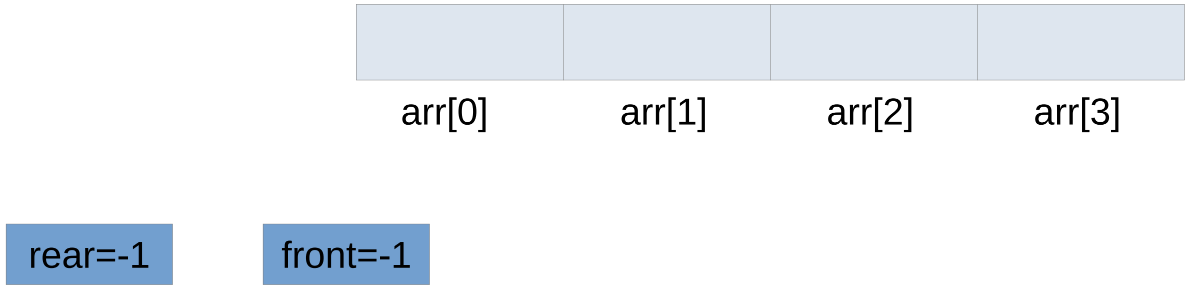
Data Structure –Queue

Linear Queue

- int arr[SIZE]

SIZE = 4

Enqueue Operation



Data Structure –Queue

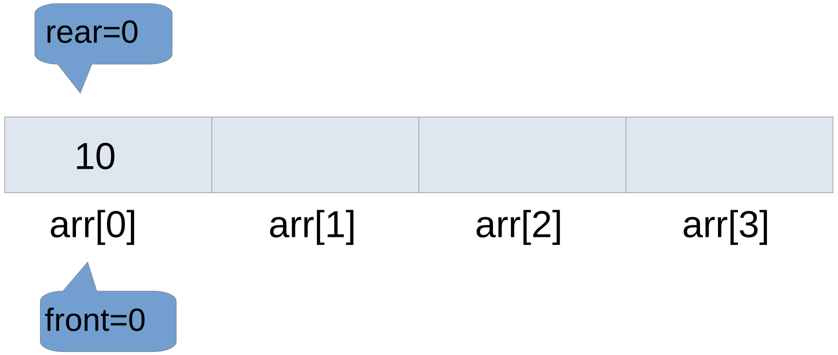
Linear Queue

- int arr[SIZE]

SIZE = 4

Enqueue Operation

enqueue(10)



Data Structure –Queue

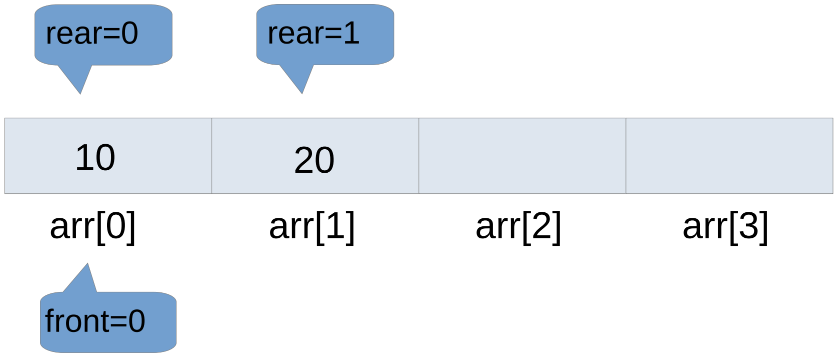
Linear Queue

- int arr[SIZE]

SIZE = 4

Enqueue Operation

enqueue(20)



Data Structure –Queue

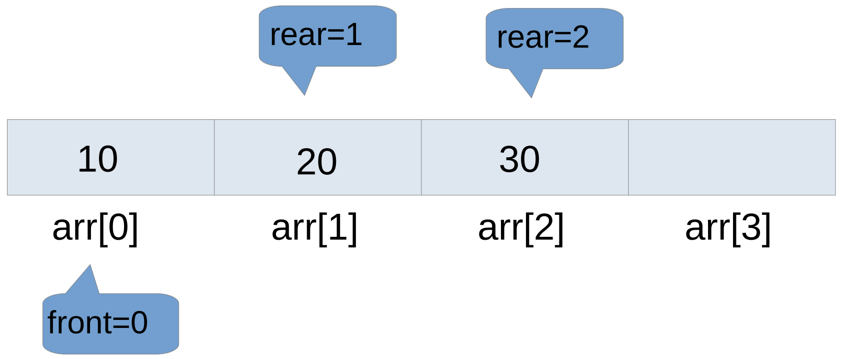
Linear Queue

- int arr[SIZE]

SIZE = 4

Enqueue Operation

enqueue(30)



Data Structure –Queue

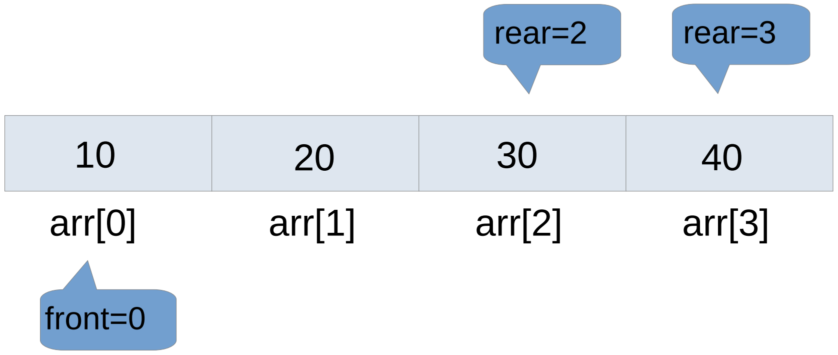
Linear Queue

- int arr[SIZE]

SIZE = 4

Enqueue Operation

enqueue(40)



Data Structure –Queue

Linear Queue

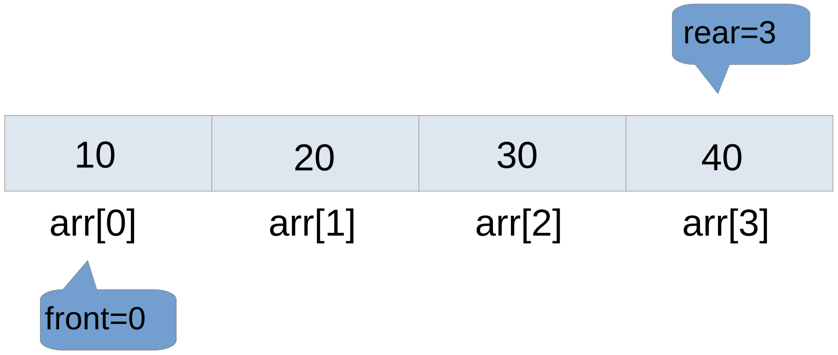
- int arr[SIZE]

SIZE = 4

Queue is full

Enqueue Operation

enqueue(50)



Data Structure –Queue

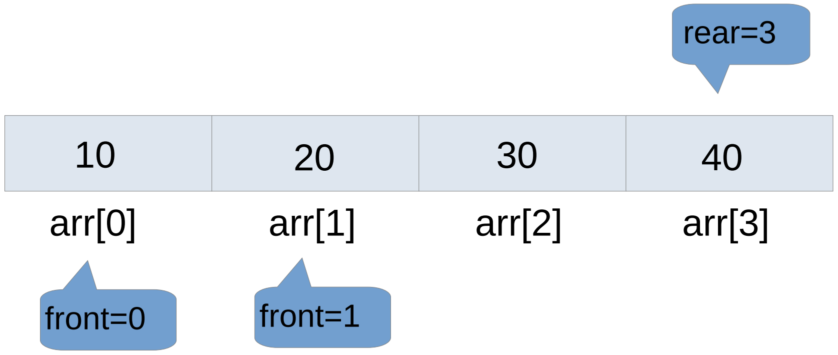
Linear Queue

- int arr[SIZE]

SIZE = 4

Dequeue Operation

dequeue()



Data Structure –Queue

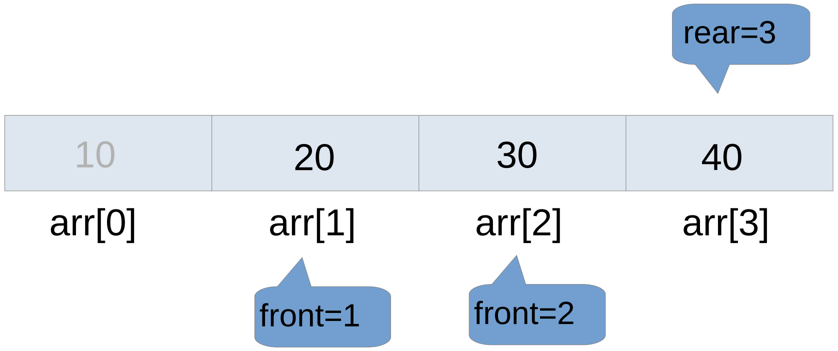
Linear Queue

- int arr[SIZE]

SIZE = 4

Dequeue Operation

dequeue()



Data Structure –Queue

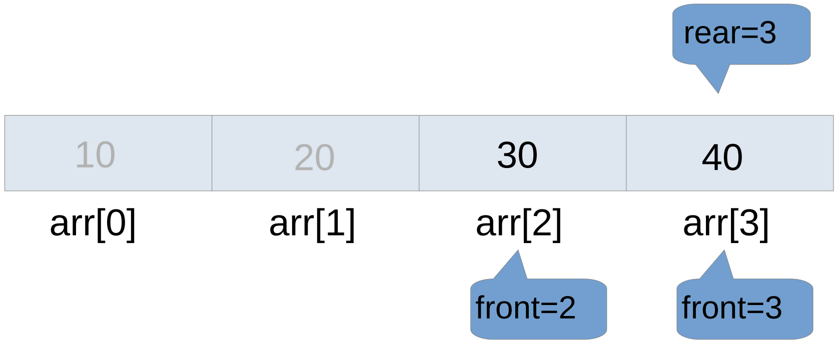
Linear Queue

- int arr[SIZE]

SIZE = 4

Dequeue Operation

dequeue()



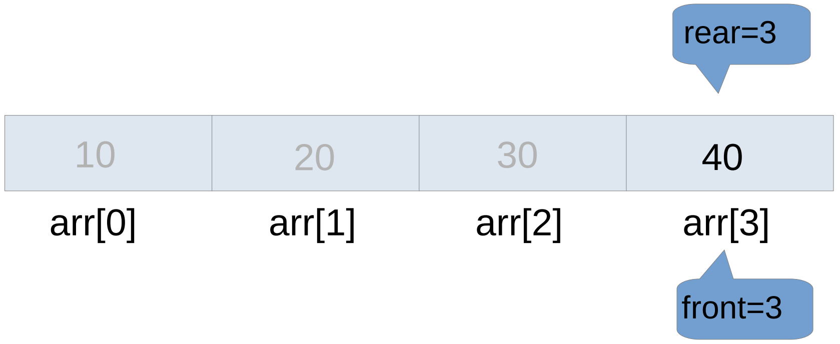
Linear Queue

- int arr[SIZE]

SIZE = 4

Dequeue Operation

dequeue()



Data Structure –Queue

Linear Queue

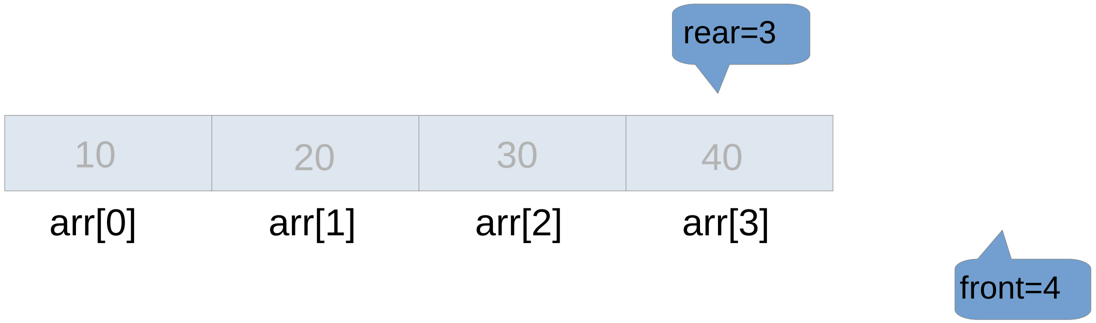
- int arr[SIZE]

SIZE = 4

Queue is Empty

Dequeue Operation

dequeue()



Data Structure –Queue

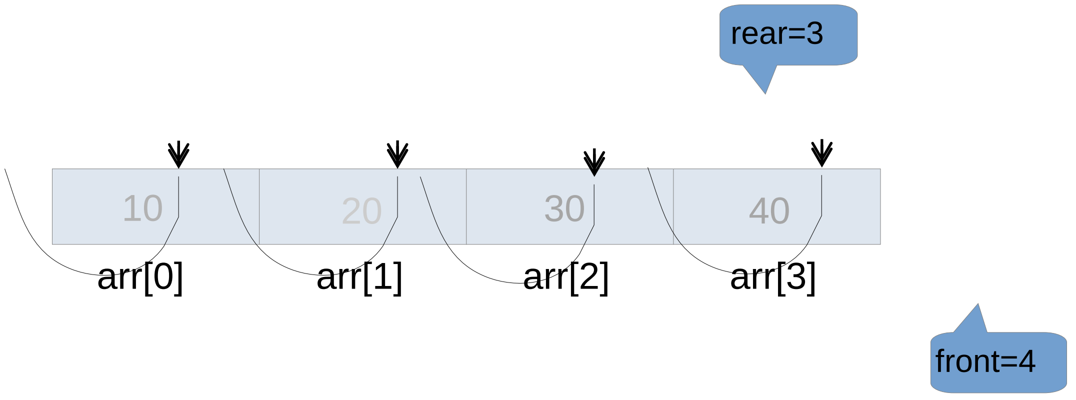
Linear Queue

- int arr[SIZE]

SIZE = 4

Enqueue Operation

enqueue(60)

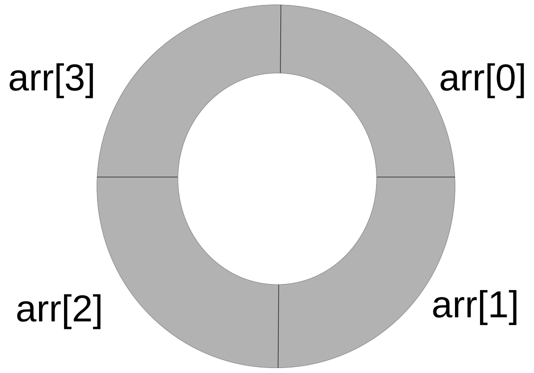


Circular Queue

Circular Queue

- int arr[SIZE]

SIZE = 4



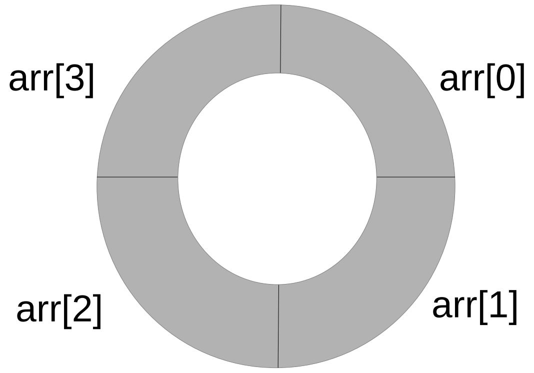
Data Structure –Queue

Circular Queue

- int arr[SIZE]

SIZE = 4

Enqueue Operation



rear = -1

front = -1

count = 0

Data Structure –Queue

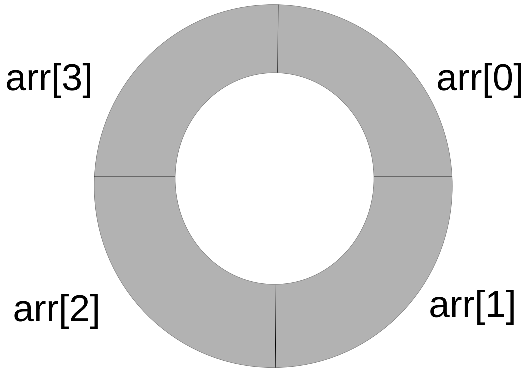
Circular Queue

- int arr[SIZE]

SIZE = 4

Enqueue Operation

enqueue(10)



rear = -1

front = -1

count = 0

rear = (rear+1) % SIZE

front = (front+1) % SIZE

Data Structure –Queue

Circular Queue

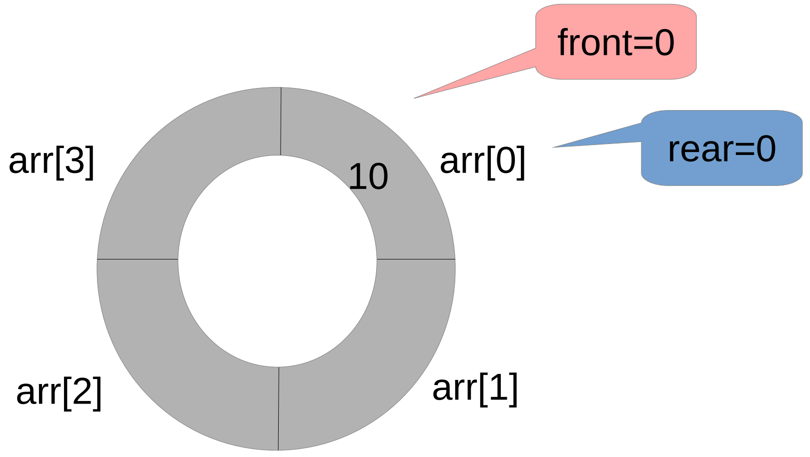
- int arr[SIZE]

SIZE = 4

count = 1

Enqueue Operation

enqueue(10)



rear = (rear+1) % SIZE

front = (front+1) % SIZE

EMERTXE

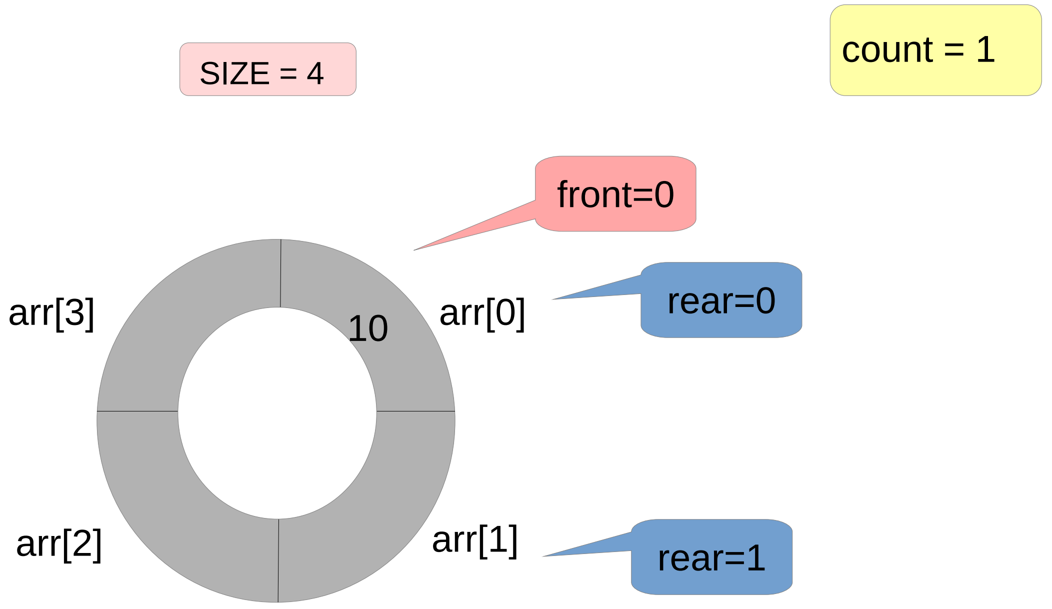
Data Structure –Queue

Circular Queue

- int arr[SIZE]

Enqueue Operation

enqueue(20)



rear = (rear+1) % SIZE

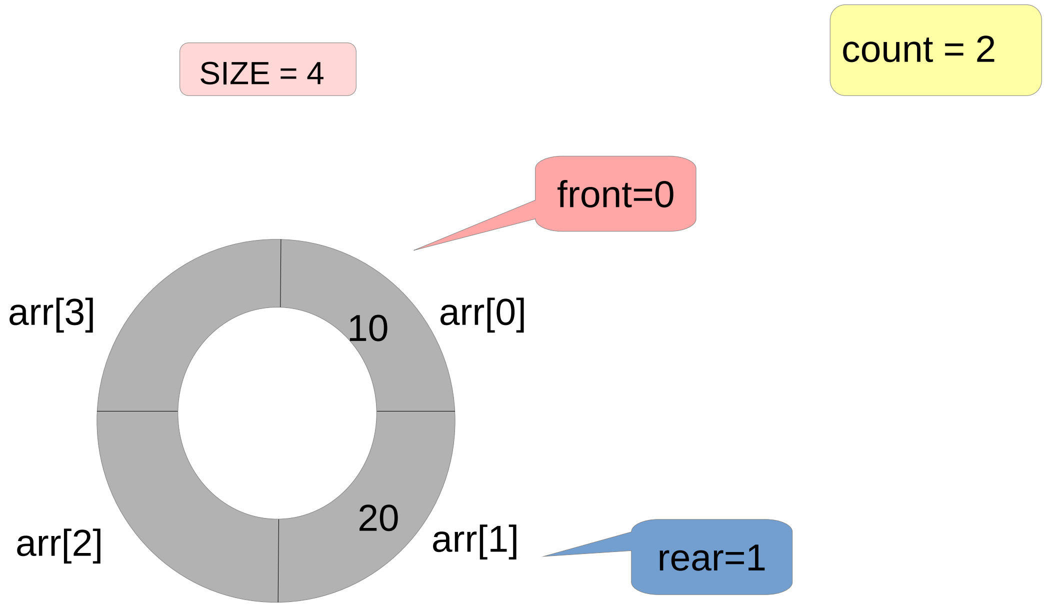
Data Structure –Queue

Circular Queue

- int arr[SIZE]

Enqueue Operation

enqueue(20)



Data Structure –Queue

Circular Queue

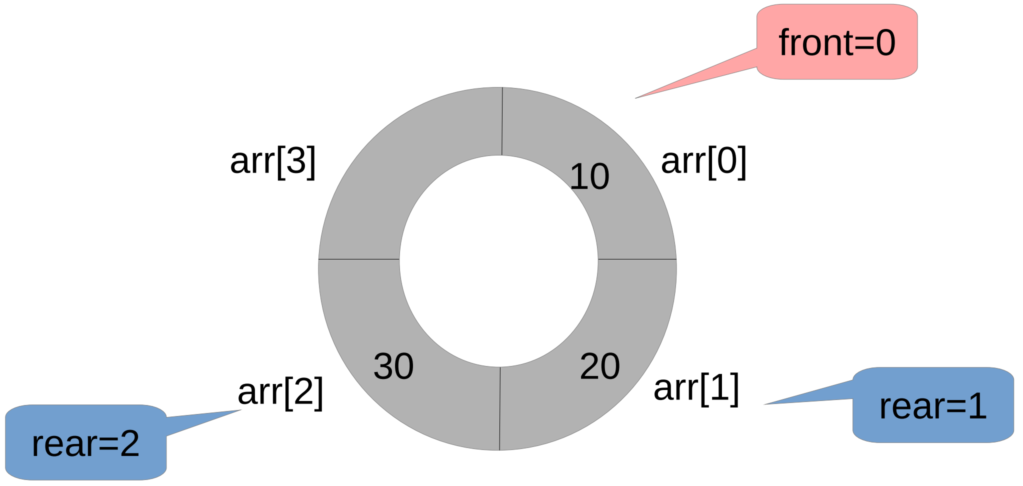
- int arr[SIZE]

SIZE = 4

count = 3

Enqueue Operation

enqueue(30)



rear = (rear+1) % SIZE

Data Structure –Queue

Circular Queue

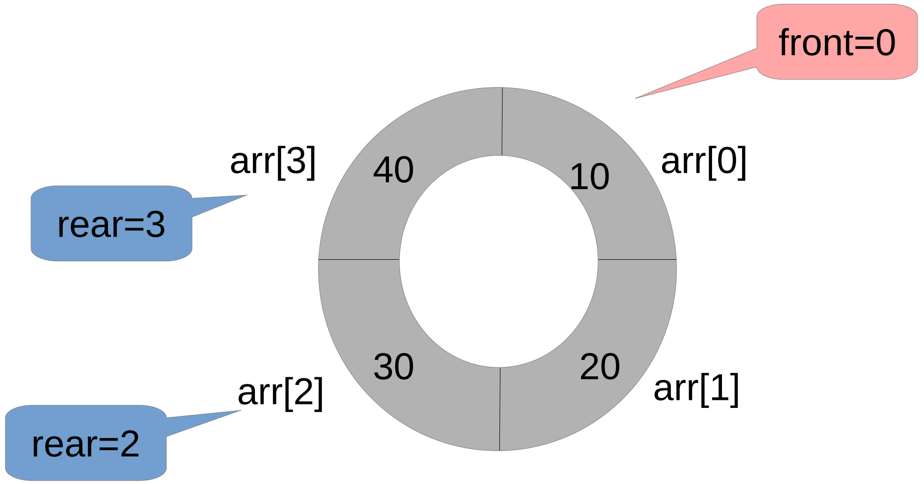
- int arr[SIZE]

SIZE = 4

count = 4

Enqueue Operation

enqueue(40)



rear = (rear+1) % SIZE

Data Structure –Queue

Circular Queue

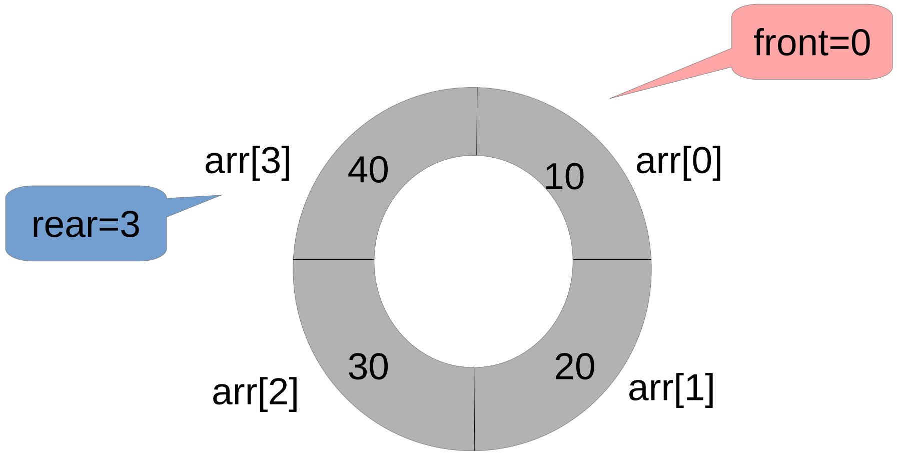
- int arr[SIZE]

SIZE = 4

count = 4

Enqueue Operation

enqueue(50)



Queue is full

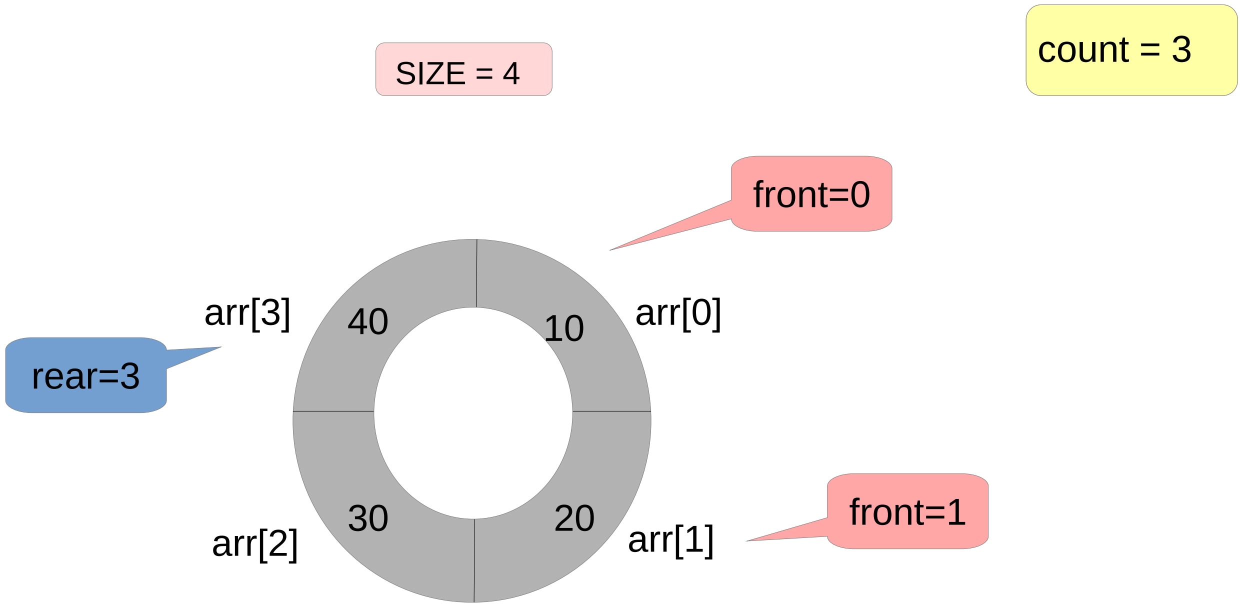
Data Structure –Queue

Circular Queue

- int arr[SIZE]

Dequeue Operation

dequeue()



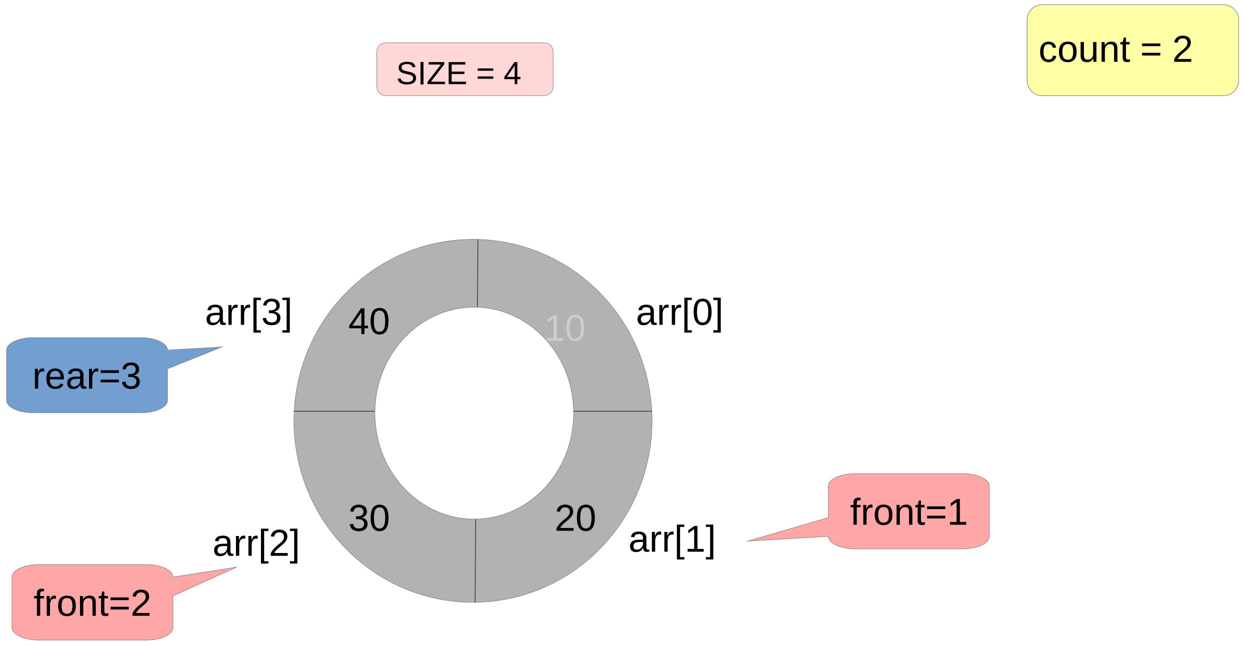
front = (front+1) % SIZE

Circular Queue

- int arr[SIZE]

Dequeue Operation

dequeue()



front = (front+1) % SIZE

Data Structure –Queue

Circular Queue

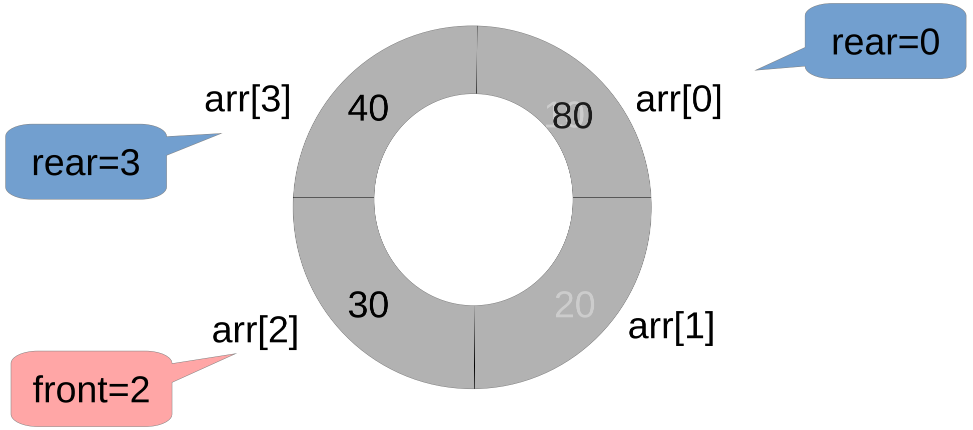
- int arr[SIZE]

SIZE = 4

count = 2

Enqueue Operation

enqueue(80)



rear = (rear+1) % SIZE

Circular Queue -Algorithm