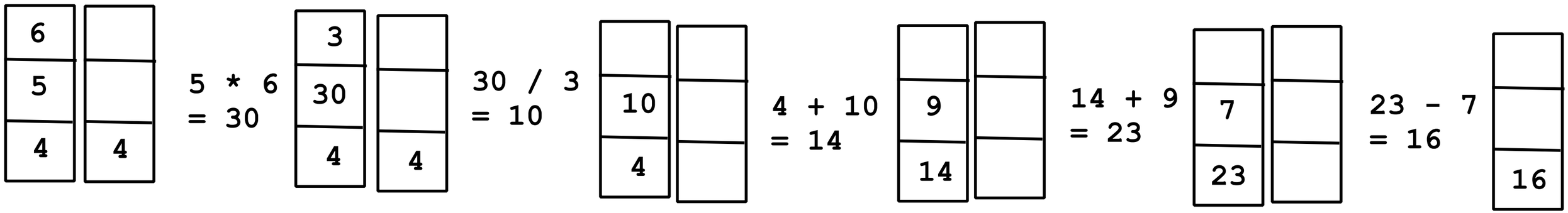
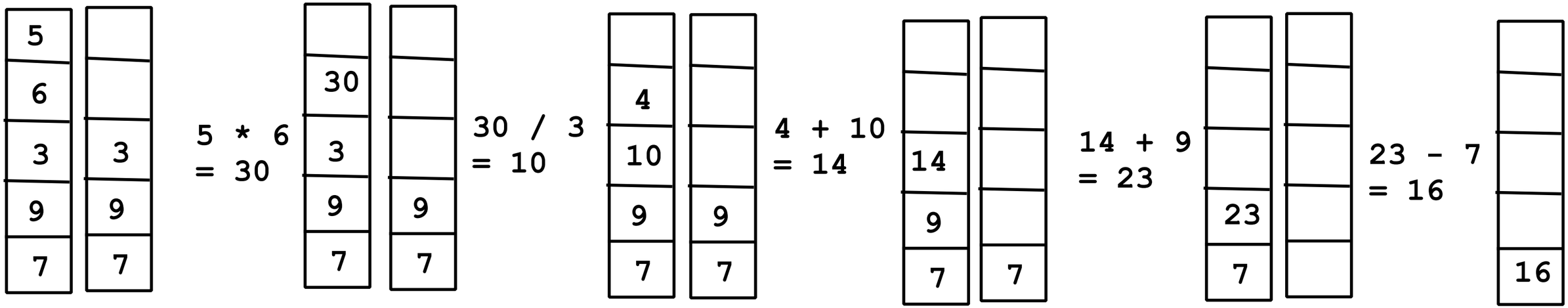
Stack Applications - Postfix Evaluation

Postfix Expression : 4 5 6 * 3 / + 9 + 7 -

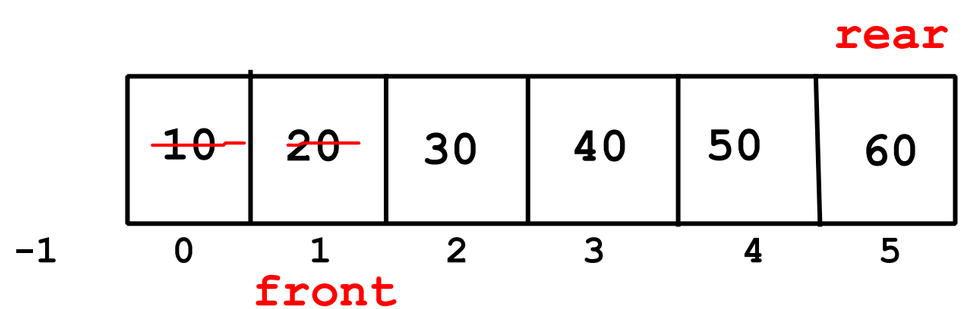


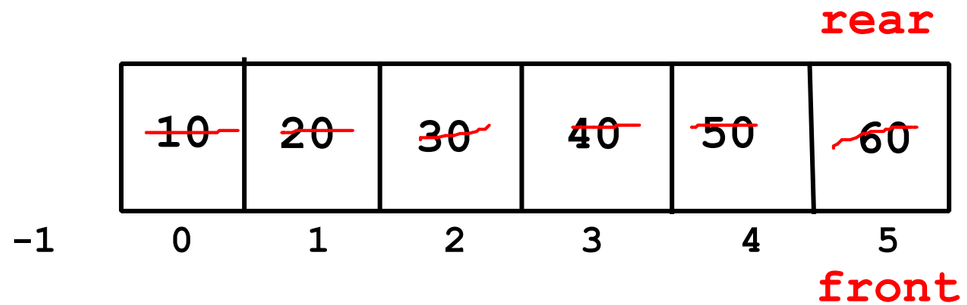
Stack Applications - Prefix Evaluation

Prefix Expression : - + + 4 / * 5 6 3 9 7



Linear Queue

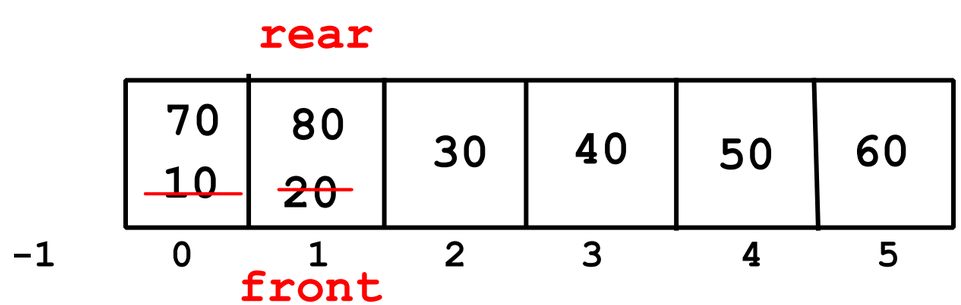




SIZE = 6

1. Empty
   front == rear
2. Full
   rear == SIZE - 1
3. Add
   a. increament rear
   b. add element/data at rear index
4. Delete
   increment front
5. peek
   return data of front + 1 index

Circular Queue

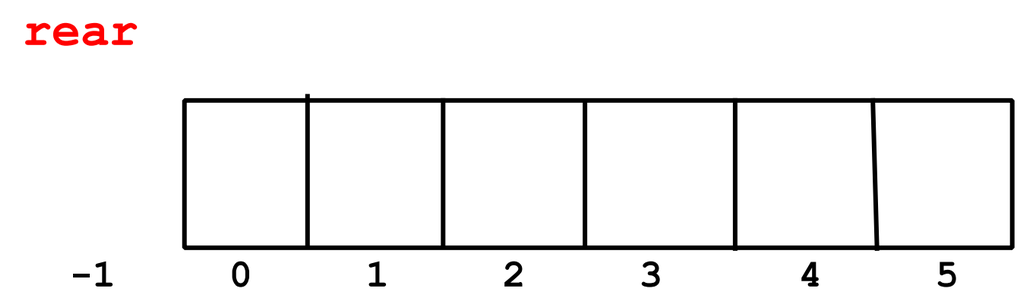


SIZE = 6

```
rear = (rear + 1) % SIZE
     = (-1 + 1) % 6 = 0
     = (0 + 1) % 6 = 1
     = (1 + 1) % 6 = 2
     = (2 + 1) % 6 = 3
     = (3 + 1) % 6 = 4
     = (4 + 1) % 6 = 5
     = (5 + 1) % 6 = 0
```

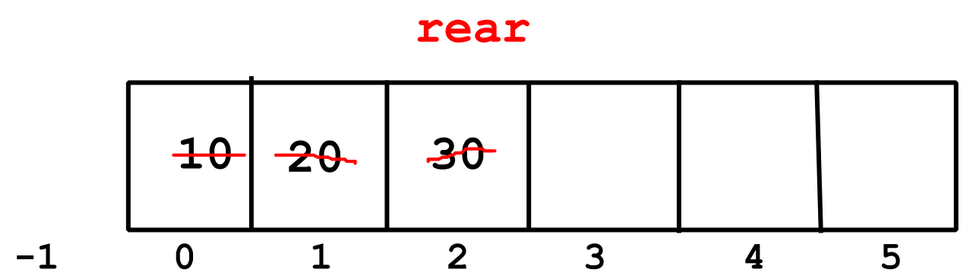
```
front = (front + 1) % SIZE
      = (-1 + 1) % 6 = 0
      = (0 + 1) % 6 = 1
      = (1 + 1) % 6 = 2
      = (2 + 1) % 6 = 3
      = (3 + 1) % 6 = 4
      = (4 + 1) % 6 = 5
      = (5 + 1) % 6 = 0
```

Circular Queue



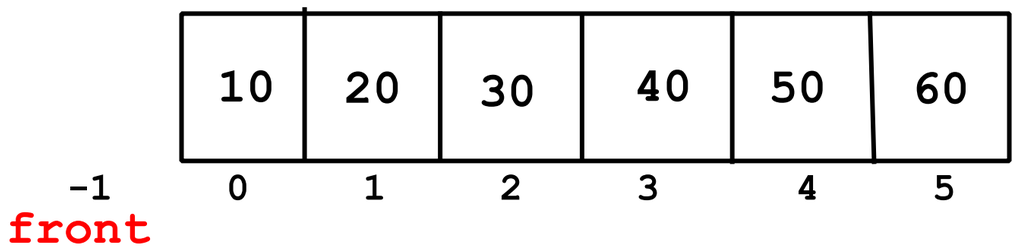
1. Empty

```
front == rear && rear == -1
```



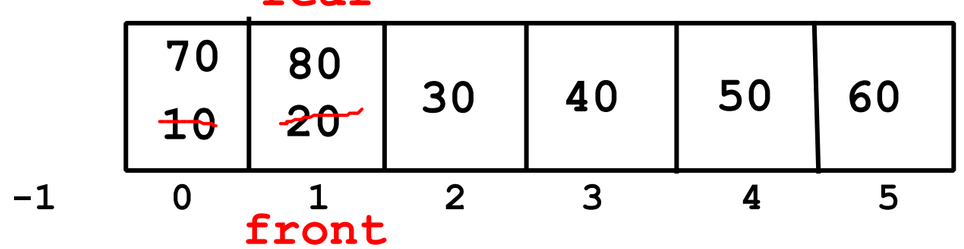
After every delete

```
if(front == rear)
    front = rear = -1
```



2. Full

```
rear == SIZE -1 && front == -1
```



2. Full

```
front == rear && rear != -1
```

```
(rear == SIZE -1 && front == -1) || (front == rear && rear != -1)
```