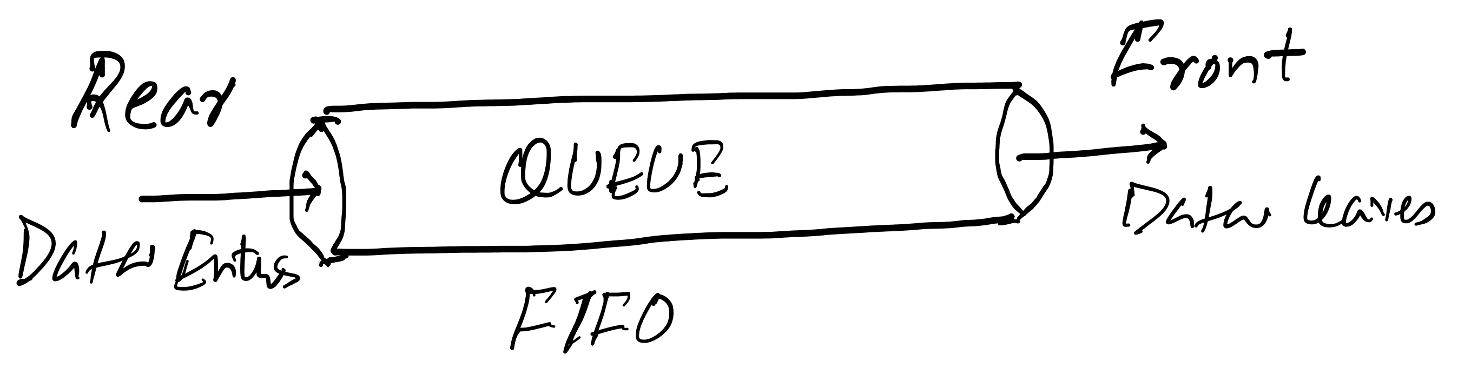
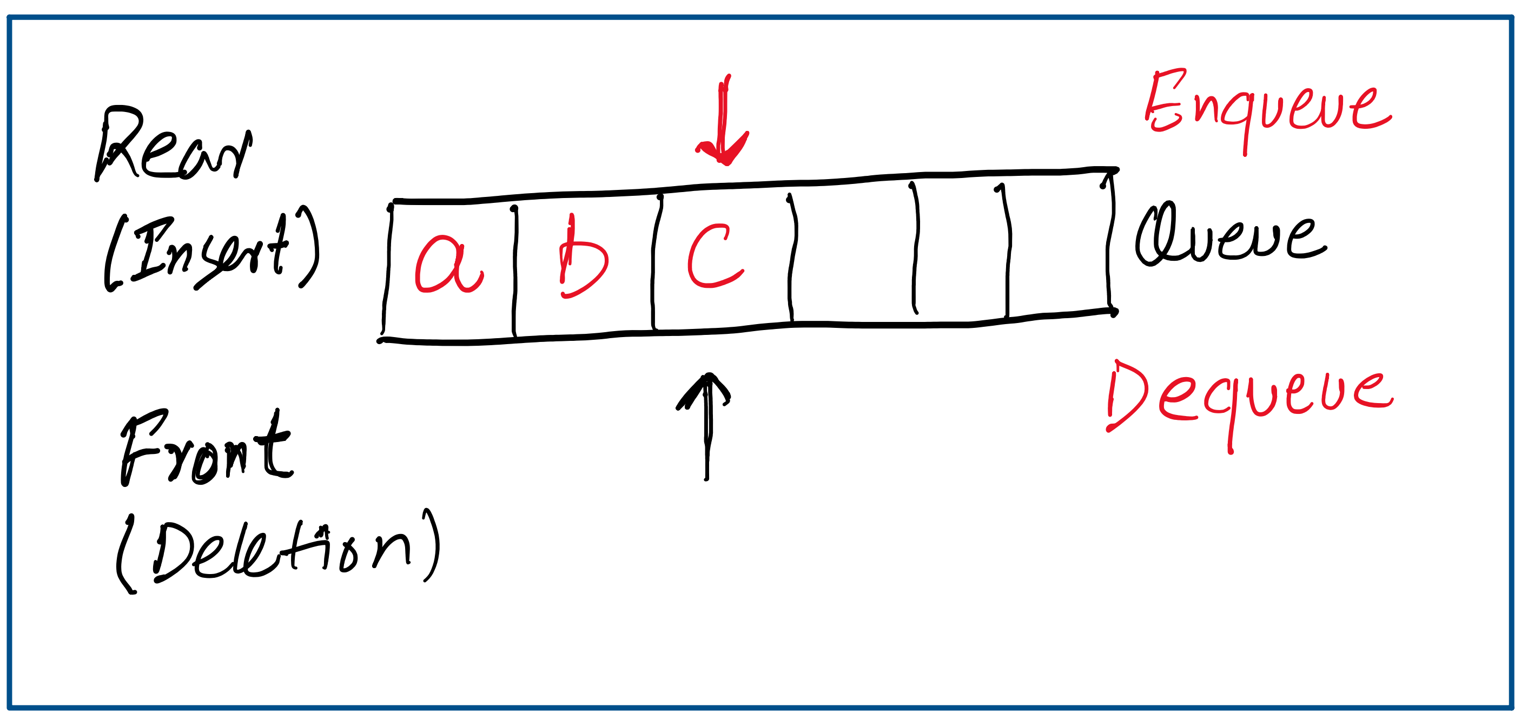
Queue ADT

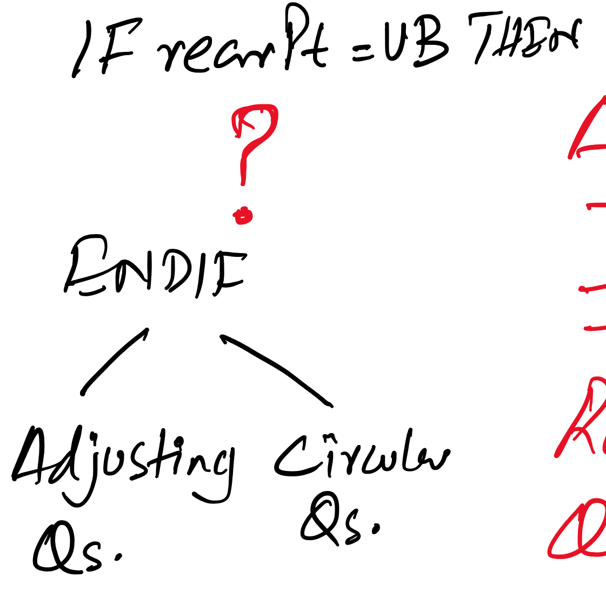
Sunday, 20 December 2020 1:11 PM



Setup Q

```
Rear ← null (-1)
front ← null (-1)
```





Add (Enqueue)

```
INPUT item
RearPtr ← RearPt + 1
Queue[RearPt] ← item
```

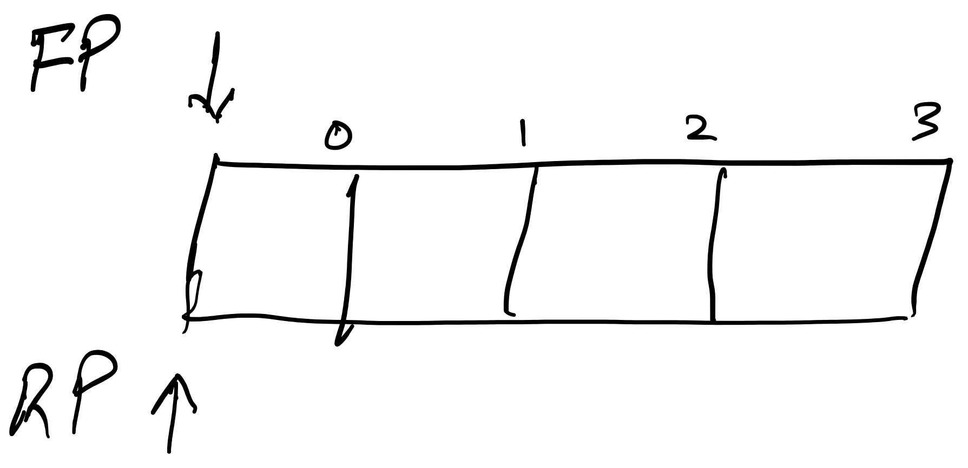
Delete (Dequeue)

```
FrontPt ← FrontP + 1
x ← Queue[FrontPt]
```

```
IF FrontPt = RearPt THEN
    QUEUE is empty
ENDIF
```

Exception

Underflow



Overflow

```
IF RP = UB THEN
    OVERFLOW.
ENDIF
```

Underflow

```
IF RP = null THEN
    UNDERFLOW
ENDIF
```

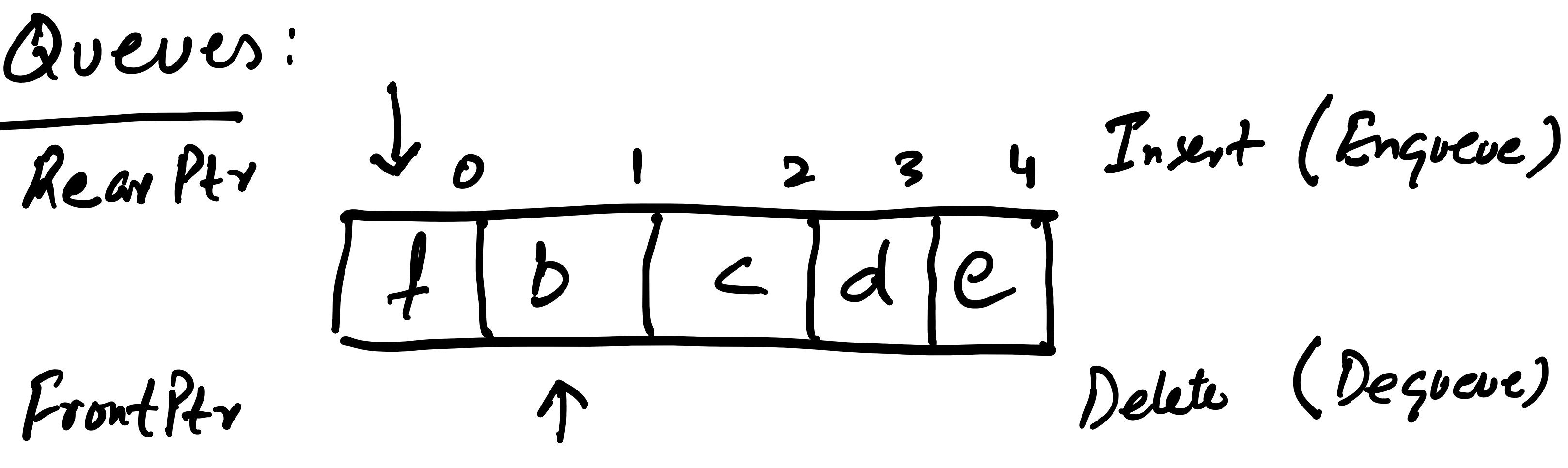
Enqueue

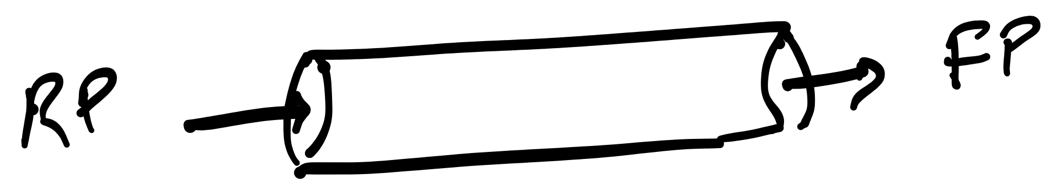
```
INPUT item
IF RP = UB THEN
    OUTPUT "Overflow"
    End
ELSE
    RP ← RP + 1
    Queue[RP] ← item
ENDIF.
```

Dequeue

```
IF RP = null THEN
    OUTPUT "Underflow"
    End.
ELSE
    FP ← FP + 1
    x ← Queue[FP]
    For i ← 0 TO (RP-1)
        Queue[i] ← Queue[i+1]
    Next i
    Queue[RP] ← None
    RP ← RP - 1
    FP ← FP - 1
```

Circular Queues:





Enqueue

```
INPUT data
IF QSize = ArraySize
    THEN
        OUTPUT "Overflow, data can't
                be added."
    ELSE
        IF RearPtr = UB
            THEN
                RearPtr = 0
            ELSE
                RearPtr = RearPtr + 1
        ENDIF
        Queue[RearPtr] = Data
        QSize = QSize + 1
ENDIF
```

Dequeue

```
IF QSize = 0
    THEN
        OUTPUT "Underflow."
    ELSE
        IF FrontPtr = UB
            THEN
                FrontPtr = 0
            ELSE
                FrontPtr = FrontPtr + 1
        ENDIF
        OUTPUT QUEUE[FrontPtr]
        QSize = QSize - 1
ENDIF.
```