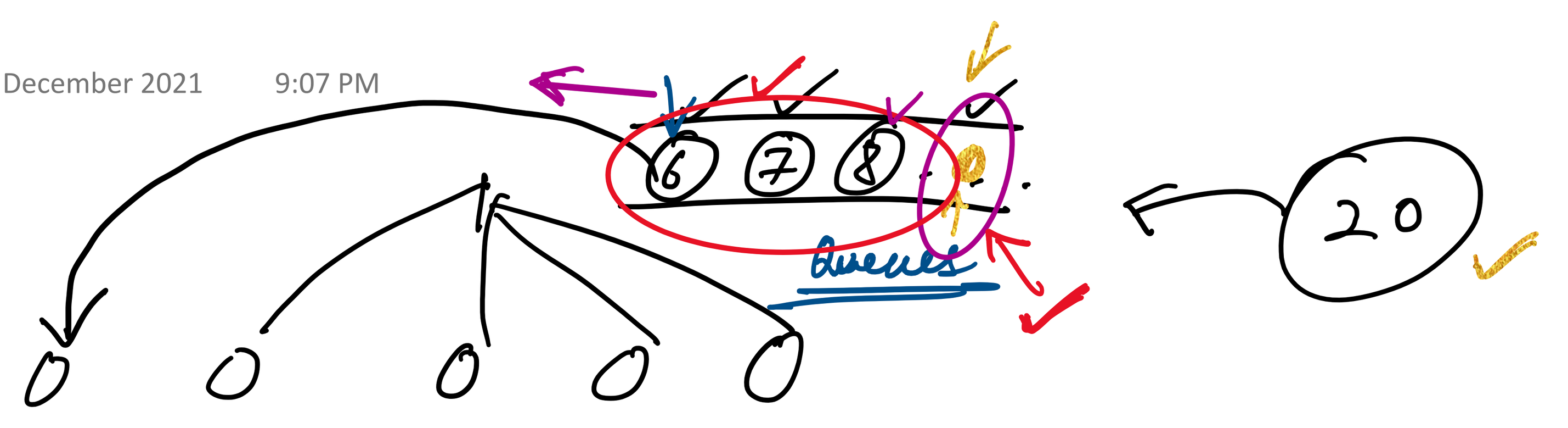
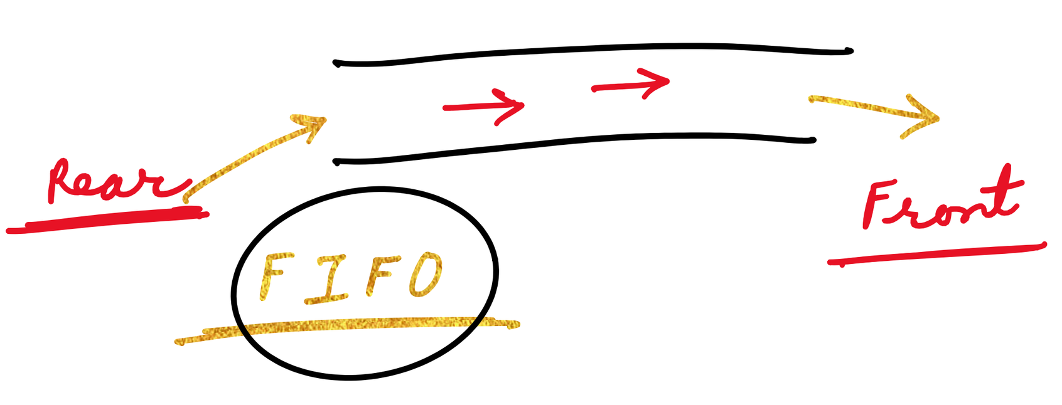
Queues

Wednesday, 22 December 2021 9:07 PM



Railway ticket counter
Airline checkin
Movie ticket counter



Operations in Queue

1) enqueue(x) → Add element x at rear end.
2) dequeue() → Remove element x from front end.
3) isEmpty() → checks if queue is empty.

Implement Queues using arrays → dynamic arrays

0 1 2 3 4 — —
| 3 | 7 | 5 | 4 | 6 |

f = ~~-1~~ ~~0~~ 1
r = ~~-1~~ ~~0~~ ~~1~~ ~~2~~ ~~3~~ 4

3 7 5 4 6 — — dequeue() dequeue()

f = -1 r = -1

```
enqueue(x) {
    a[++r] = x;
}

isEmpty() {
    return (f == r);
}

int size() {
    return r-f;
}

dequeue() {
    if(!isEmpty()) f++;
}
```

TC = O(1)

(f+1) ——— r

Front element → a[f+1]
Rear element → a[r]

Issues
1) Memory waste
2) How to undo an incorrect entry?

0 1 2 3 4 5 6 7 8 (F, R)
f = -1
r = -1 ... 0 1 ... 2 3 ... 4 ... 8

0 1 2 3 4 5
| 1 | 2 | 3 | 4 | 5 |
f = -1 0 1
r = -1 → 4

Dqueue (Double ended queue)
(Stacks + Queues)

```
enqueueR(x) {
    a[++r] = x;
}

dequeueF() {
    if(!isEmpty()) f++;
}
```

(f+1) ——— r

```
enqueueF(x) {
    a[f--] = x;
}

dequeueR() {
    if(!isEmpty())
        r--;
}
```

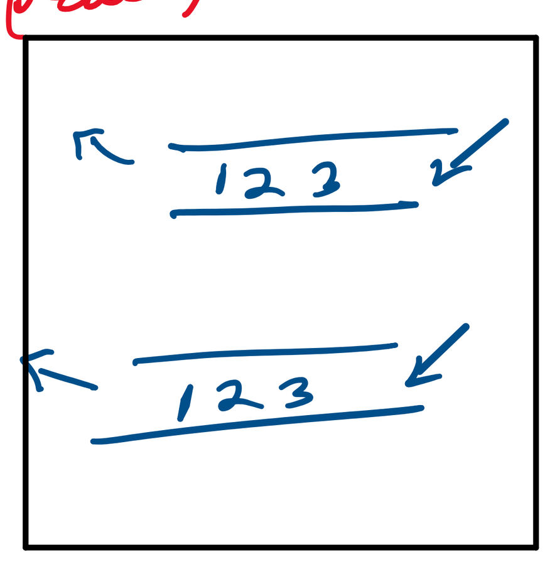
f = -1 dynamic array

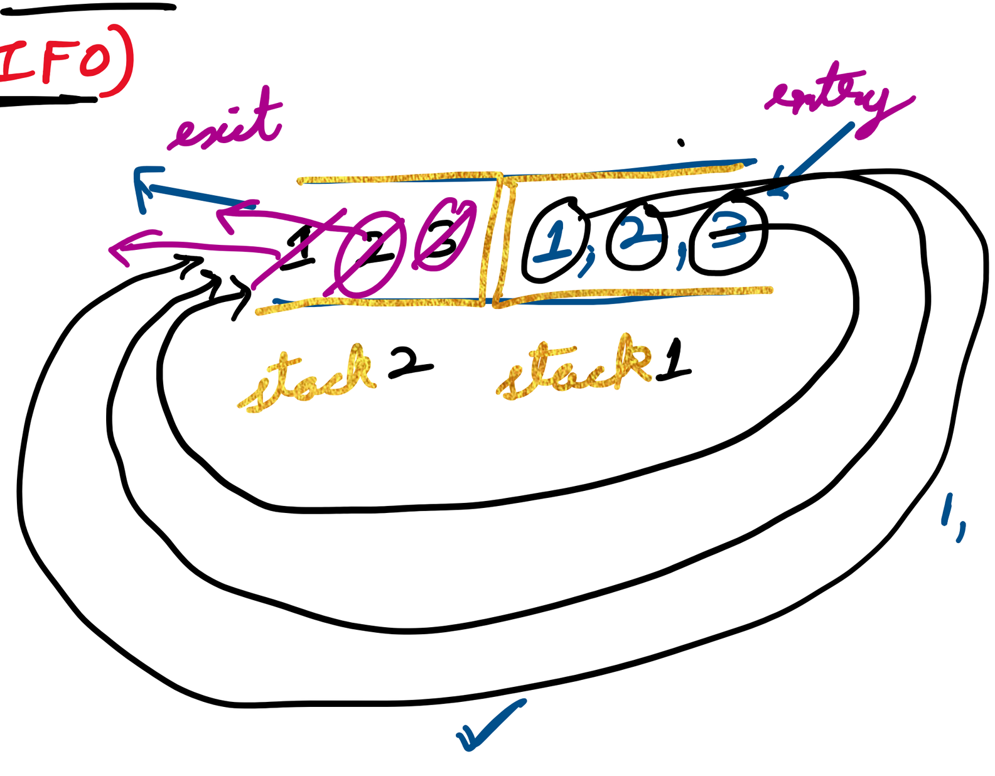
TC = O(1)

int a[4] → a[0] + 4 * intsize
4 * 4 = 16
1000
1016

Q → Implement Queues (FIFO) using stacks. (LIFO)

(Oracle)





1, 2, 3 - dequeue() (3 steps)
dequeue() (1 step)
dequeue() (1 step)

steps → 1 1 3 1 1 1 1 4 1
7 4 dq 6 3 dq 8 dq dq

s2 s1
1 extra step ∀ elements

```
enqueue(x) {
    s1.push(x);
}

dequeue() {
    if(!s2.isEmpty()) {
        return s2.pop();
    } else {
        while(!s1.isEmpty()) {
            s2.push(s1.pop());
        }
        if(!s2.isEmpty()) s2.pop();
        return -1;   // underflow
    }
}
```

s1 → s2

If Queue has N elements we are doing N extra steps in total.
⇒ extra step per element = 1

TC = O(1)

K elements
~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ | ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~
O(1)
s2 s1

dequeue() → K steps
For K-1 elements dq → 1 step

Q → Find N^{th} perfect number
↳ Even length pallindromic number with digits 1 & 2.

aa'
a' → reverse of a

Ans → 11, 22, 1111, 1221, 2112, 2222, 111111, 112211 ...
N → 1, 2, 3, 4, 5, 6, 7, 8
$O(10^5)$

N = 3 Ans = 1111
N = 5 Ans = 2112

check if no. is perfect no. starting from 11, 12, 13 ......
TC = O(Answer * len of number)

If a < b
⇒ aa' < bb'

12 < 21 12 < 112
12-- < 21-- 12-- < 112---

~~1~~, ~~2~~, ~~11~~, ~~12~~, 21, 22, 111, 112, 121, 122 N = 10

x → x*10+1
x*10+2

122 221 (Ans)

1
2
11
12
21
22
111
112
121
122
211
212
221
222
1111
1112
1121

TC = O(N)
SC = O(N)

N = 10

```
if(N%2 == 0) {
    prepend ← append 2
    N = N/2 - 1
} else {
    prepend ← append 1
    N = N/2
}
```

N = ~~17~~
~~8~~
~~3~~
~~1~~
0

1121

$O(\log_2(N))$