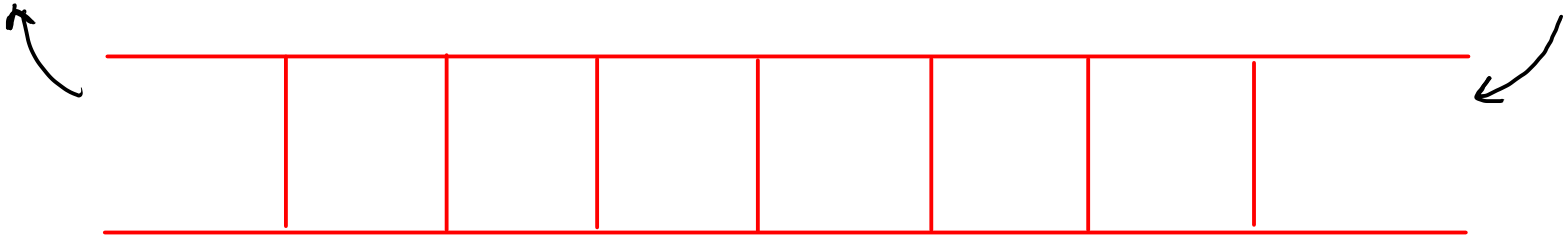
⇒ Queue (FIFO)

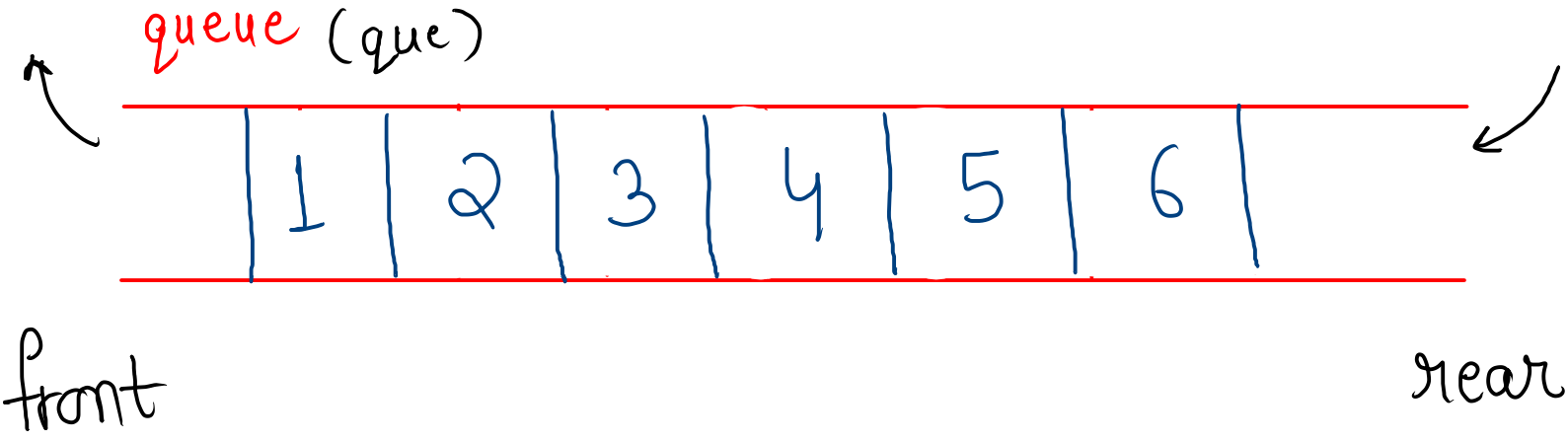
first in, first out



front

rear

Note :- In queue we add elements from rear and we remove elements from front



Syntex :-

```
Queue< DataType> que = new LinkedList<>();
```

functions:-

- que.add(x); // to add element from rear
- que.remove(); // to remove ele. from front
- que.peek(); // to tell value of front ele.
- que.poll(); // to return the value from front and remove it as well.
- que.size();
- que.isEmpty();

Queue Syntax Learning

1)

```
public static Queue<Integer> que;
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);

    que = new LinkedList<>();

    int t = scn.nextInt();
    while (t-- > 0) {
        int c = scn.nextInt();
        if (c == 1) {
            printSize();
        } else if (c == 2) {
            removeElement();     // from front
        } else if (c == 3) {
            int x = scn.nextInt();
            addElement(x);  // from rear
        } else if (c == 4) {
            printFrontElement();
        } else {
            System.out.println("Invalid Input");
        }
    }
}
```

2)

```
public static void printSize() {
    int s = que.size();
    System.out.println(s);
}

public static void removeElement() {
    if ( que.size() == 0 ) {
        System.out.println("-1");
    } else {
        que.remove();
    }
}

public static void addElement(int x) {
    que.add(x);
}

public static void printFrontElement() {
    if ( que.size() == 0 ) {
        System.out.println("-1");
    } else {
        int ans = que.peek();
        System.out.println(ans);
    }
}
```

all fn T.C is O(1)

Print Binary

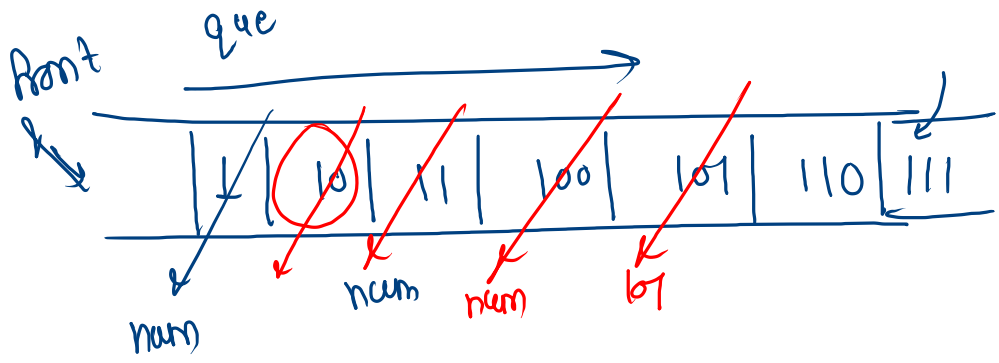
2

decimal (10) binary (2)

0	0
1	1
2	10
3	11
4	100
5	101
6	110
7	111
8	1000
9	1001
	1010

10

n

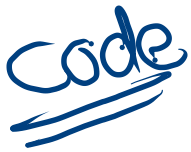


num + "0"

num + "1"

1000 | 1001 | 1010

1011



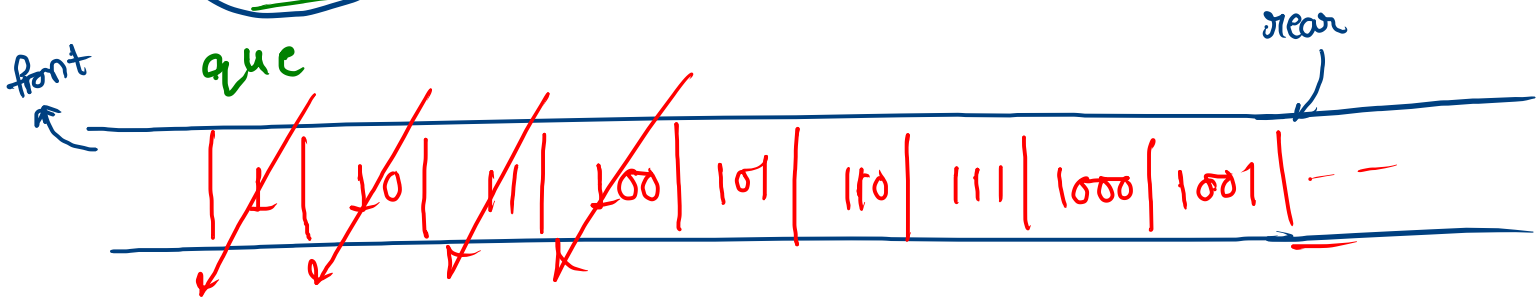

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();

    Queue<String> que = new LinkedList<>();
    que.add("1");
    for (int i = 1; i <= n; i++) {
        String str = que.poll();
        System.out.print(str + " ");

        String str1 = str + "0";
        que.add(str1);

        String str2 = str + "1";
        que.add(str2);
    }
}
```

Ex :- n = 13 → 26 = 2 × 13 = 2 × N



4

str = "1"
str1 = "10"
str2 = "11"

str = "10"
str1 = "100"
str2 = "101"

str = "11"
str1 = "110"
str2 = "111"

str = 100
str1 = 1000
str2 = 1001

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();

    Queue<String> que = new LinkedList<>();
    que.add("1");
    for (int i = 1; i <= n; i++) {
        String str = que.poll();
        System.out.print(str + " ");

        String str1 = str + "0";
        que.add(str1);

        String str2 = str + "1";
        que.add(str2);
    }
}
```

T.C = O(N)

S.C = O(N)

→ memory = 2 * N