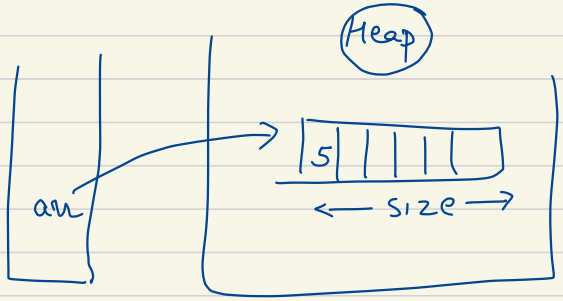
2D Arrays

ArrayLists

2D Arrays

Arrays → Primitives → Memory Difference

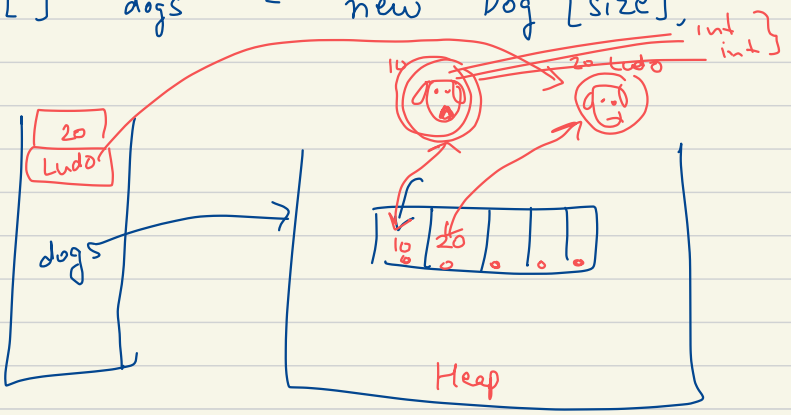
→ objects




```
int[] arr = new int[size];

arr[0] = 5;
```

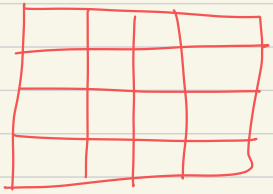
Dog [] dogs = new Dog [size];



Dog ludo = new Dog();

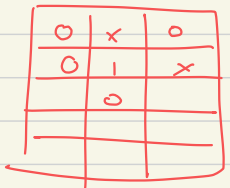
dogs[0] = new Dog();

dogs[1] = ludo;



Matrix

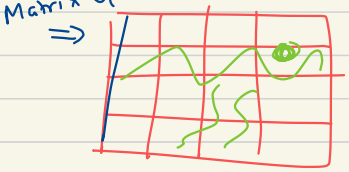
Puzzle - Sudoku / Tic-Tac-Toe



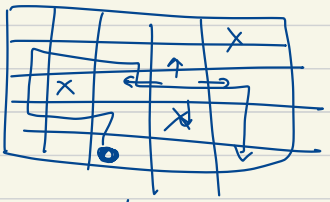
"Game"

id	P	C	M
0	10	20	30
1	40	50	60
2	-	-	-
3	-	-	-

Matrix of Pixel values (RGB)



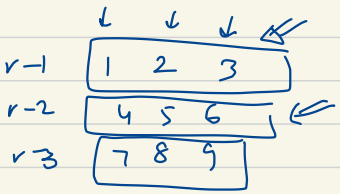
Images



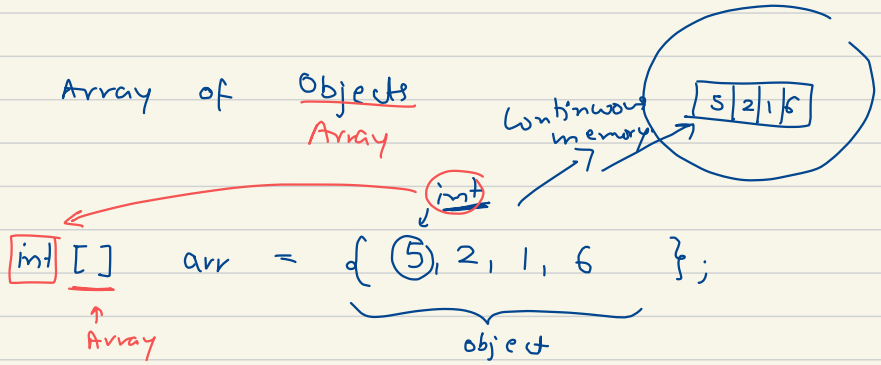
Graph (implicit graphs)

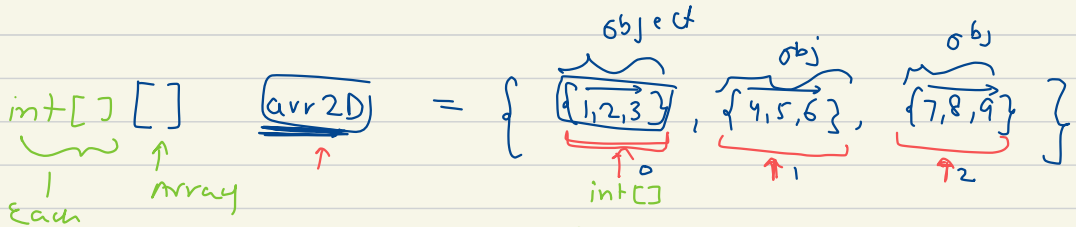
Memory Level

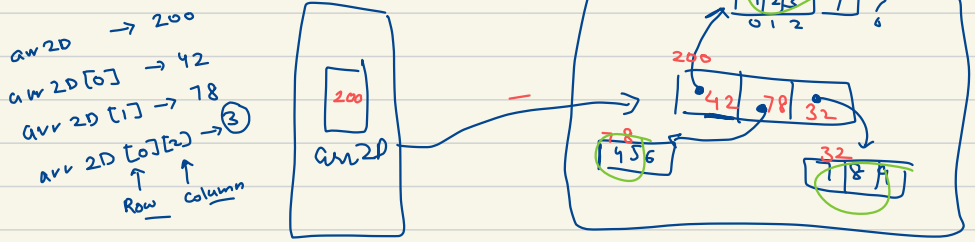
Create an Array of Objects

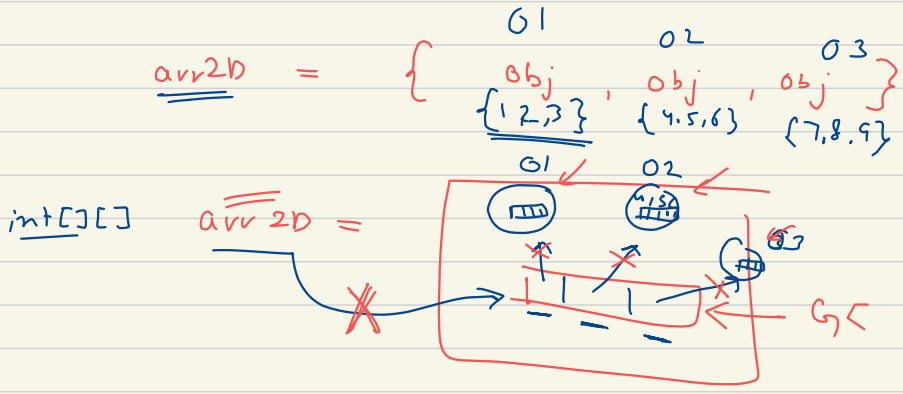


Array







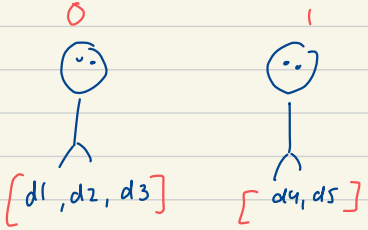


arr 2D [0] ⇒ Add of 0th Array (obj Ref)

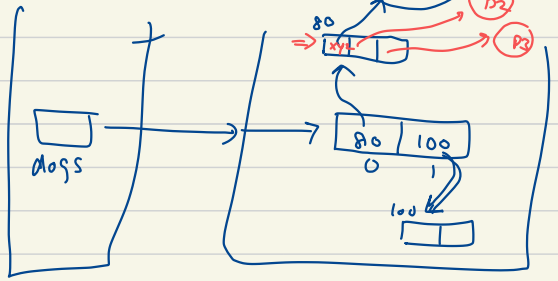
2D Array of Dogs

```
Dog [] [] dogs = { {d1, d2, d3},
                   {d4, d5} };
```

dogs[0][0] = d1

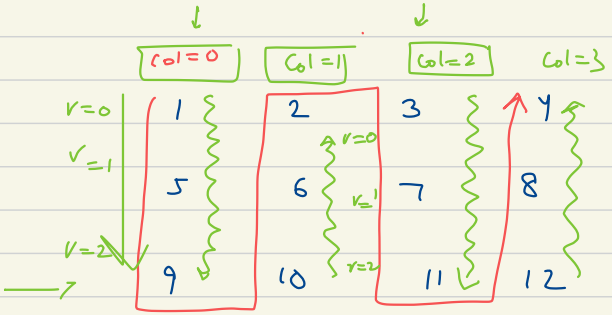



```
d1 = new Dog();
d2
d3
d4
d5
```

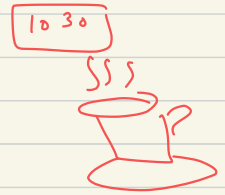


Time To TRY (Fixed cols) Wave Print

Live Assgn

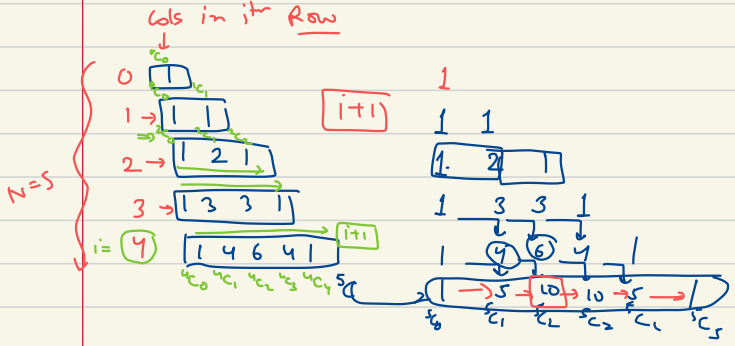


5 Mins + 10 Break. :)



Output: Print

1, 5, 9, 10, 6, 2, 3, 7, 11, 12, 8, 4

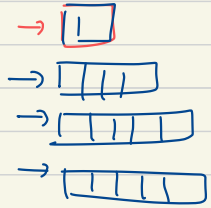


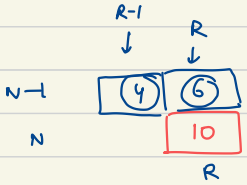
① Fact ↑

② 2D Array

③ formula $^{N}C_{R}$ ↷ k. $^{N}C_{R+1}$ =

$^{N}C_{R}$ colm





arr(n,r) = arr(n-1,r)

+ arr(n-1,r-1)

"ARRAY LISTS"

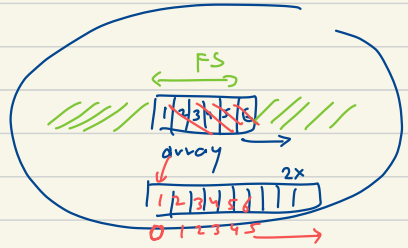
Array that can Dynamic ⇒ grow & shrink in size.

Java Collections framework

↓

Library of Data structures, predefined classes.

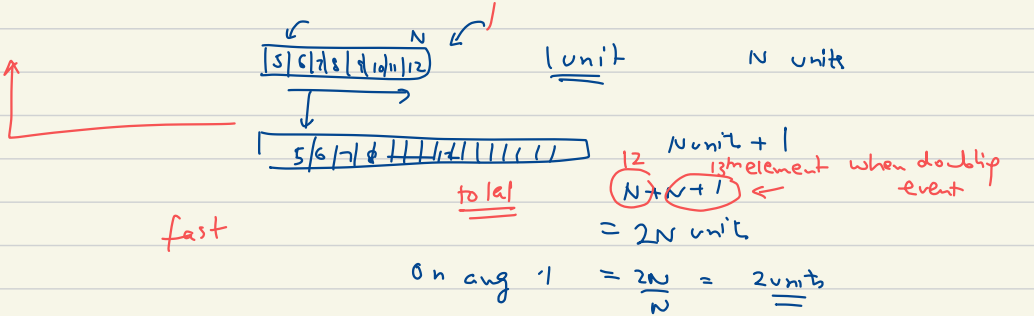
Internals



[1.5 x Java
2 x C++

Reserve some init Capacity to avoid freq doubling

Doubling is a expensive op.



Space Complexity

→ Additional space as fn of N.

Give an array

| 1 | 3 | 5 | 2 | 6 | 8 |

→

O(1) Constant

Given a stream of numbers ⇒ 1, 3, 5, 2, 6, 8

→ O(1)

largest → 8

| 1 | 3 | 5 | 2 | 6 | 8 |

→ 8 O(N)

O(N²)