

Java
+
DSA

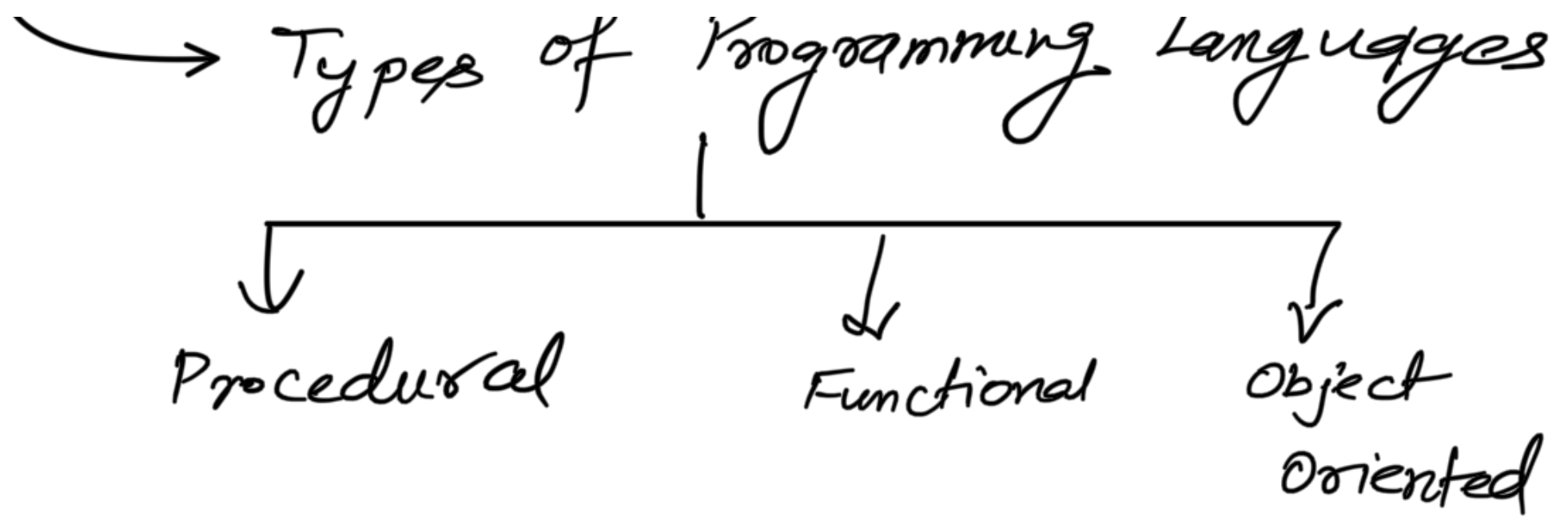
Lecture 1 - Introduction

What is Programming?

Human 1 English Human 2
Abhishek ← Ashok
 Hindi

Human 3 How are you? Computer → Binary
Abhay Java/C++ "How are you!"; (0, 1)

Programming is a way to instruct the
Computer to perform various task.



Procedural

- well-structured ✓
- order (systematic) ✓
- functions and commands ✓

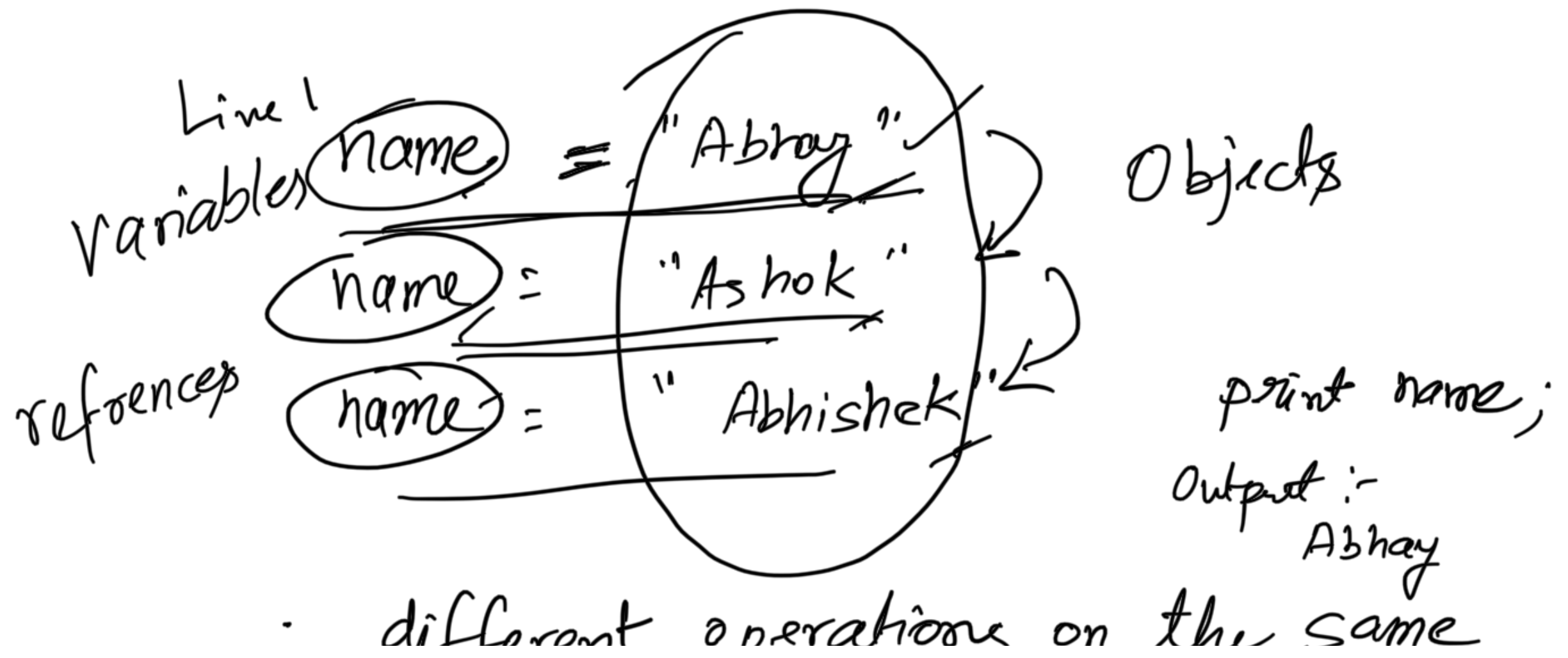
Functional

- pure functions ✓

- never modify variables
- only create new ones


Abhay
Ashok
Abhishek

→ variable : name



set of data

3. Object Oriented ✓

- revolves around objects.
- Code + Data. 
- develop, debug, reuse, maintain
→ easy -

Q1. Can you name a language which is of all these three types?

- Python

Java → Procedural + OOP

Static → still,
not changing
or
fixed

Dynamic → not still,
changing,
not fixed

"Abhay"

datatypes declare before
use.
↳ String, int, char

datatype ← ~~String~~ name;

pl. int a;
a = 10;

String
^ name = "Abhay";

No need to declare
datatype of
variables

a = 10;

a = "Abhay"

172. $a = 10 \rightarrow$ Error:

String datatype
required
no datatype
is found

is it a dynamic
or
Static?

$a = 10$

print a .

Error

~~int a;~~

$\text{int } a = 10$

s.o.p(a);

More information
required

W/W

1. Perform type checking
compile time

Perform type
checking at
runtime

3. More Control

Save time in
writing code.

function
definition

function add(a, b) {

³
c = ¹a + ²b;
print c;

Function call add(1, 2);

Output :-

3

Add 4242 and 2634

$\text{add}(4242, 2634)$

Memory Management

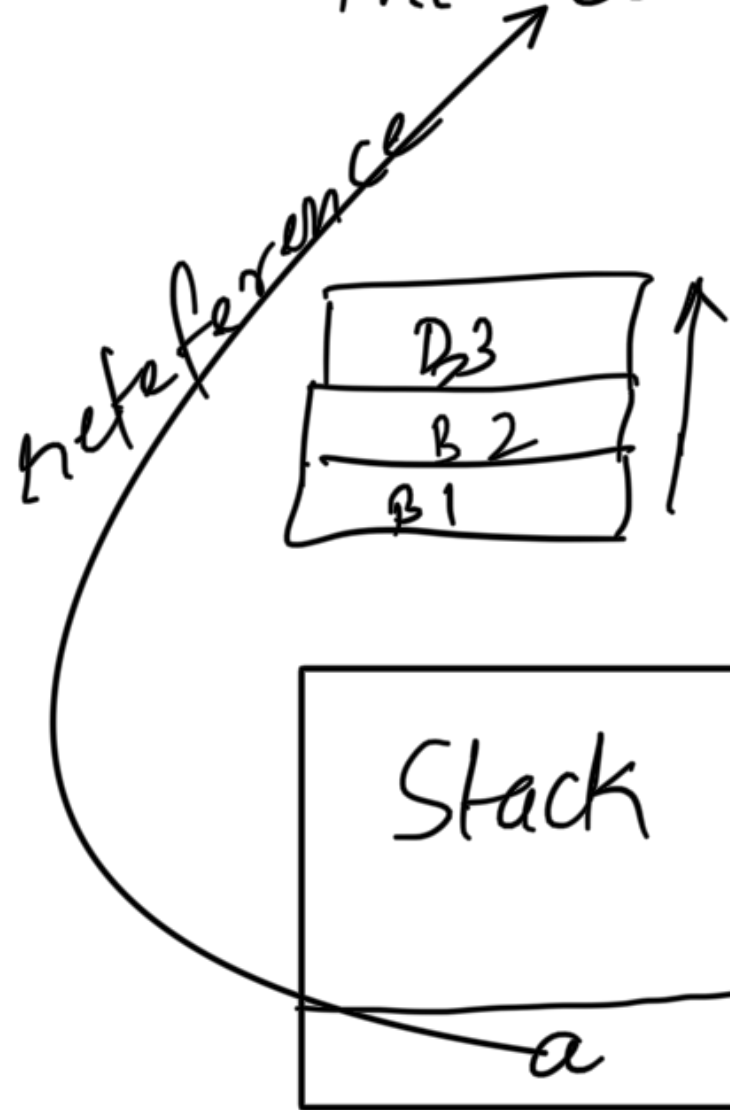
2 types of memory

1. Stack

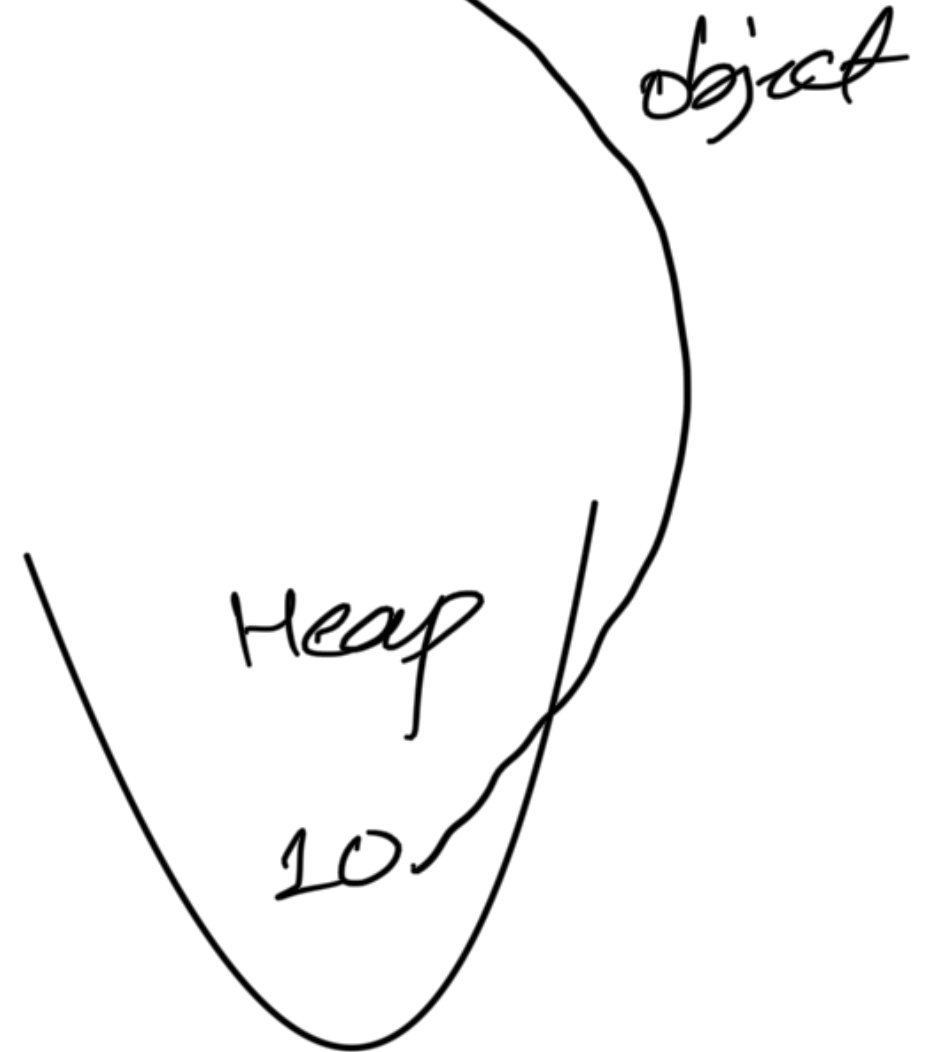
2. Heap

declare a variable

int ^{reference} a = 10 ^{object}



Reference variable
of
the object

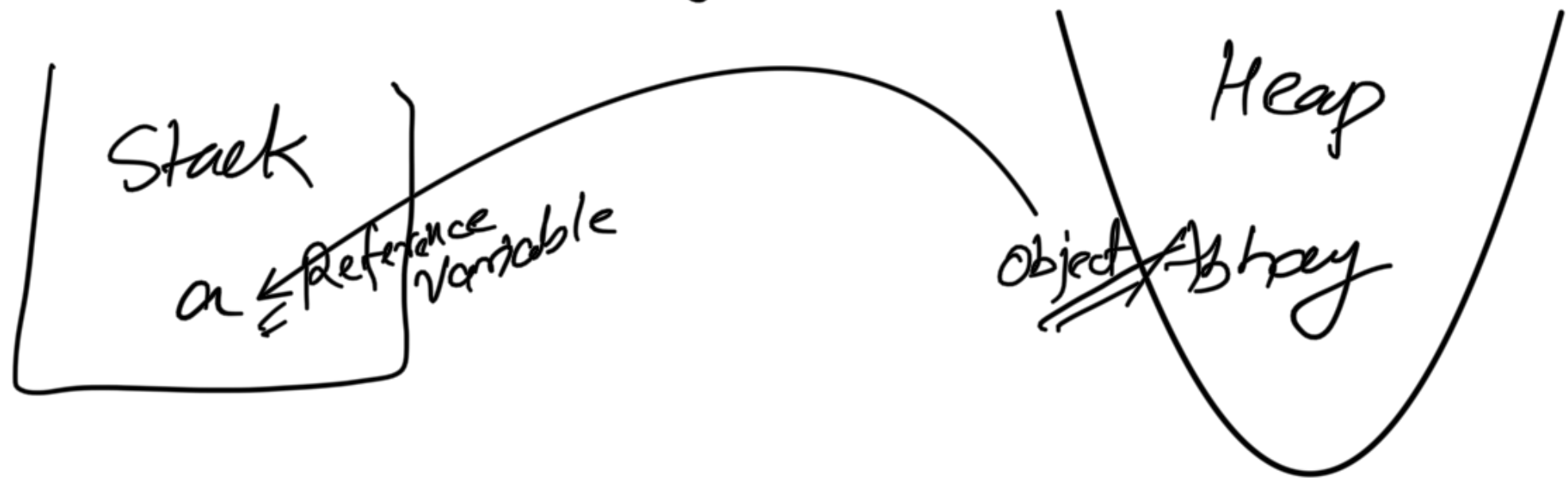


Heap memory
stores the

is stored in
a stack

object of
reference variable

int a = "Abhay"



Introduction to DSA

Data Structure

Algorithm
→ $\underset{\substack{\text{ref} \\ \text{var}}}{\text{inta}} = 10$ $\underset{\text{value}}{\text{object}}$

Data Structure is a way of collecting and organising data in such a way that we can perform operations in an effective way.

$$c = a + b$$

Eg. Abhay $\xrightarrow[\text{ref var}]{\text{name string}}$ studies programming

and he is 19 years old
Object value
int ref variable

→ java util - age
Print this : "Abhay's age is 19."

String name = "Abhay";

int age = 19;

System.out.println(" ");

int add (a, b) {

} echo -

→ print ✓
→ cout ✓

System.out.println(name + "s age is" + age);

Annotations:
- "name" is annotated with "ref variable".
- "s age is" is annotated with "S".
- "age" is annotated with "D".

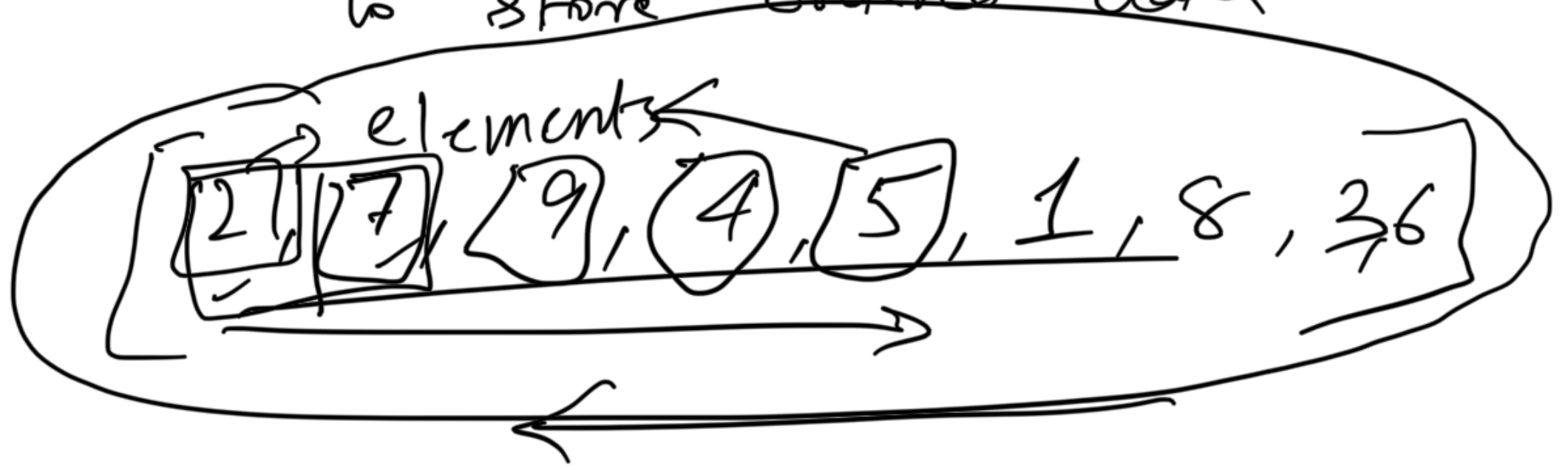
Output :- Abhay's age is 19

Annotation: "obj value" points to the output.

Data Structure → OOPs (code + data)
Class
Structure

Simple Language

DS are structured program
to store ordered data



Order operation

ASC - 1, 2, 3, 4, 5, 6, 7, 8

DES - 8, 7, 6, 5, 4, 3, 2, 1

D.S.

Find an element 5 in an array

1 To 100,000

unordered \rightarrow Difficult

ordered \rightarrow less difficult

1, 2, 3, | 4, 5, 6, 7, 8

$3 < 5$

$4 < 5$

1 to 100,000

Worst Case Scenario

7000

7124, 7126, 10000, ..., 7000

1,00,000

7000 → 7000
1 to 1,00,000

Search 7000

7000 > 100,000

7000 sec

1 million sec

1 sec = 1 operation

Amazon

Product: TV

TV. 7000

Unordered

Total products:

1 million

1 million sec

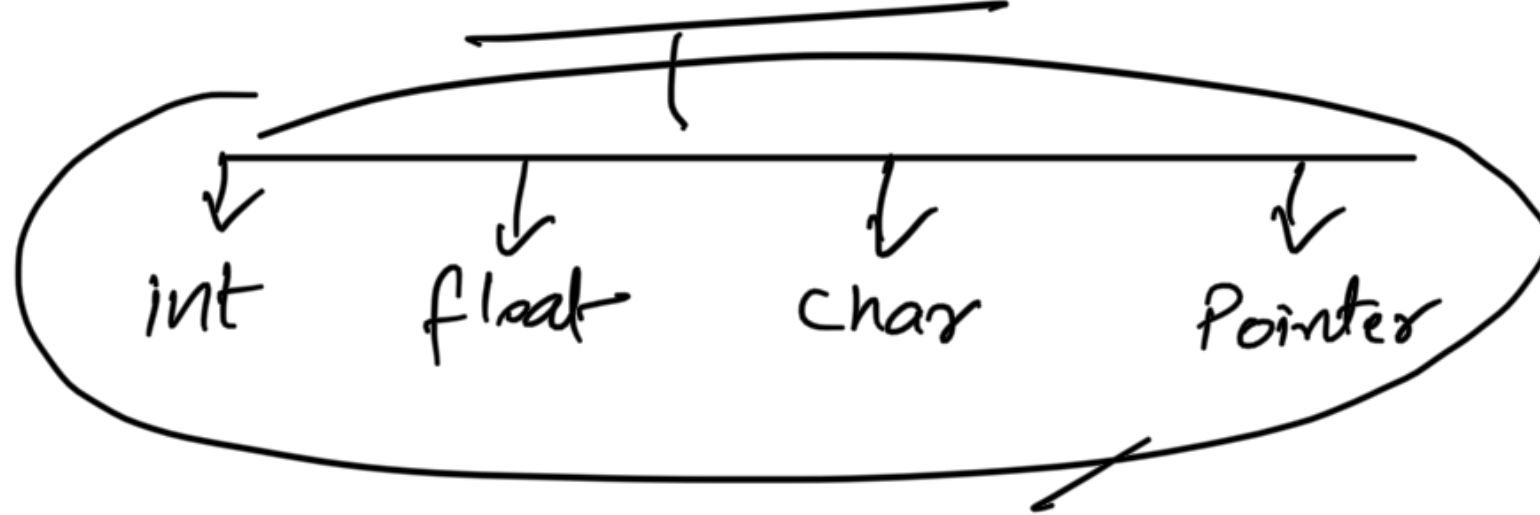
Ordered

7000 sec

7 ms ✓

Data Structure

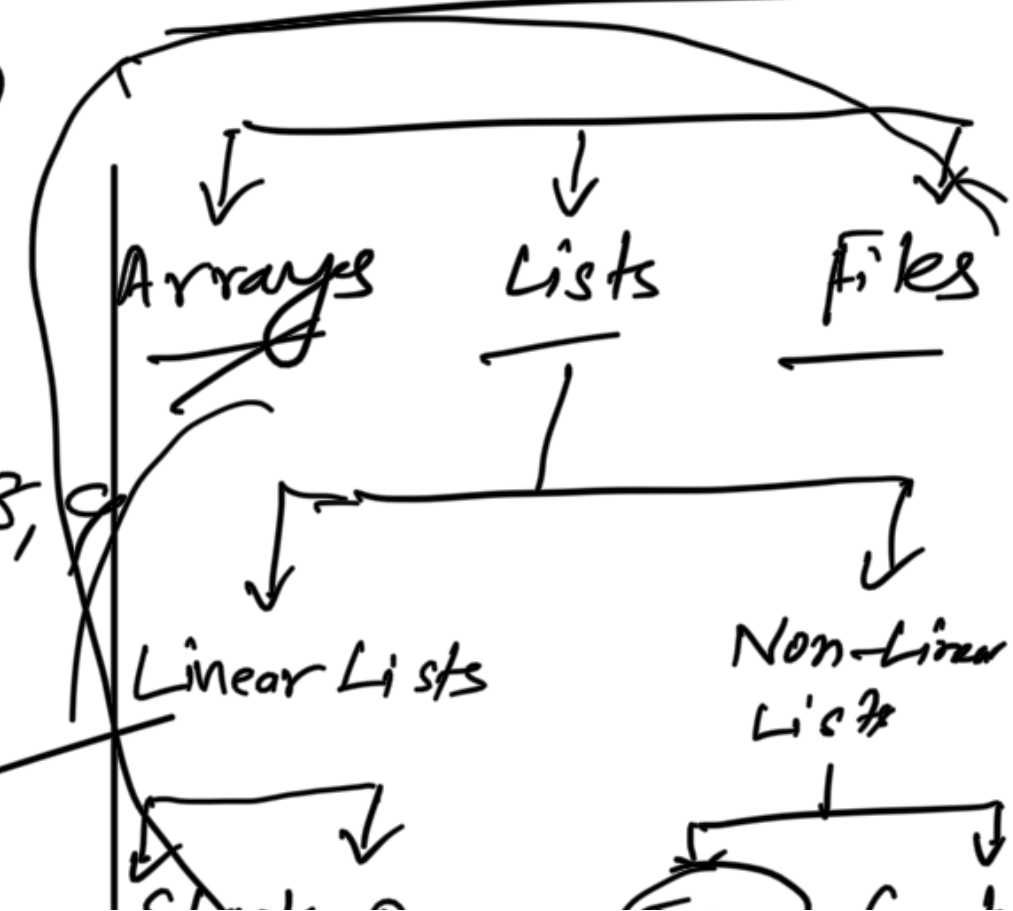
Built-in



int : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

int a = 10

User Defined Data Structures



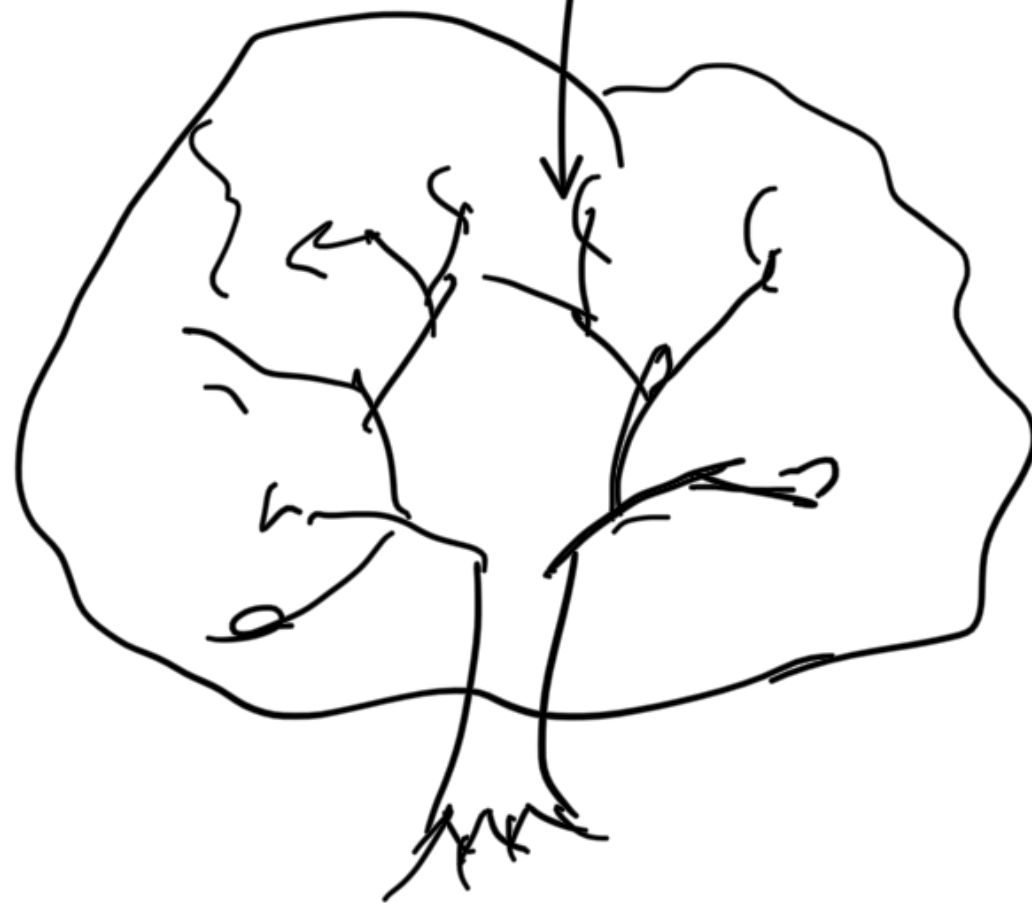
Stacks Queues (Tree) Graph

Line
Straight
horizontal or vertical

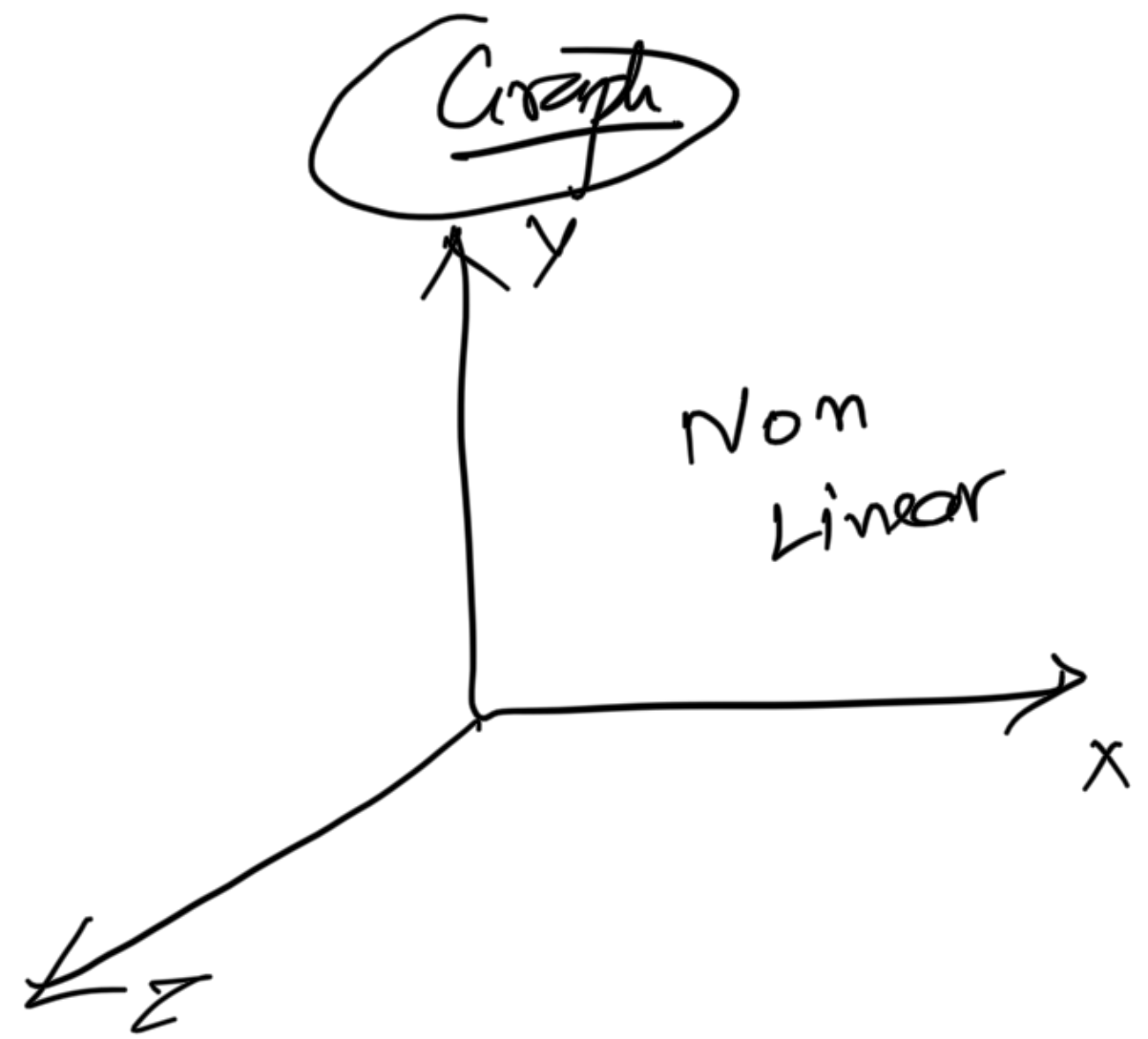
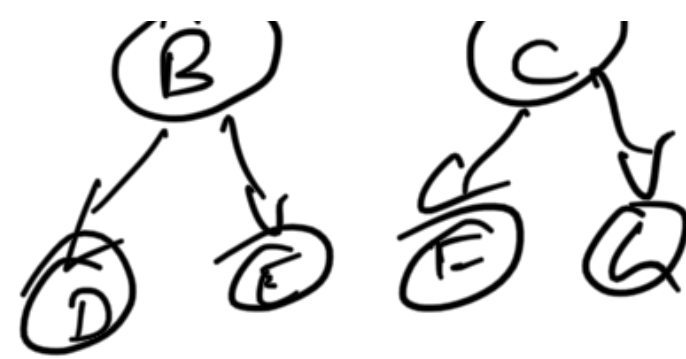
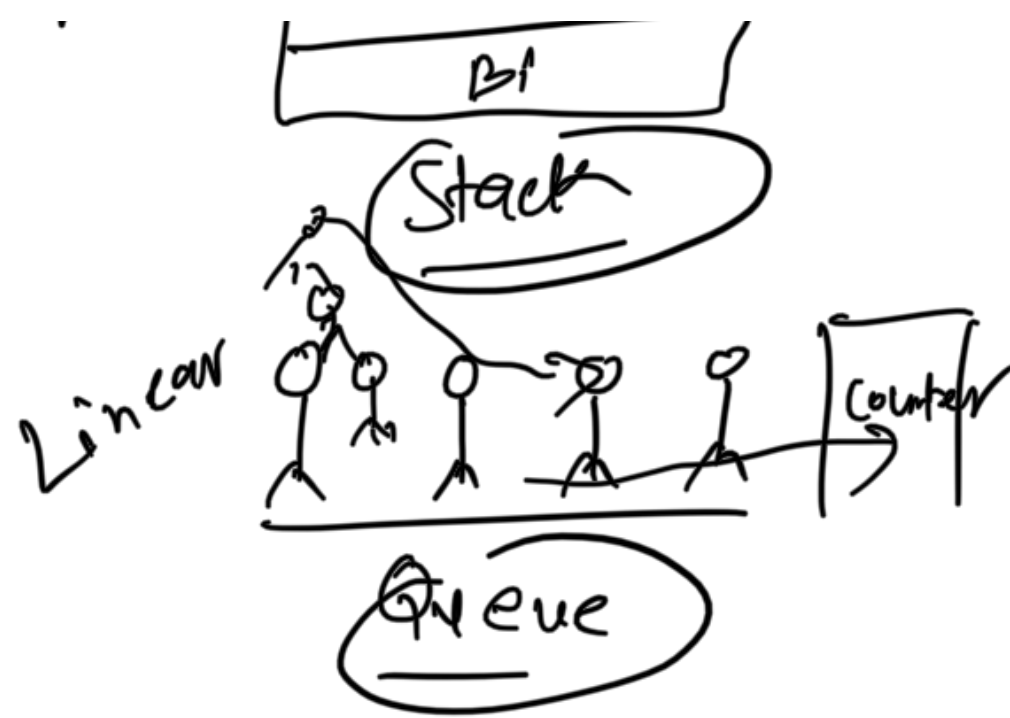
row = 0 1 2 3
[1, 2, 3, 4]

Linear

B4
B3
B2



Non Linear



Algorithm

✓ Logic :- To print the sum of 2 integers

Algo

- We will take two integers
- We will add these two
- And store it in a variable
- print that variable

Pseudo
Code

1. `int a = 10 ;`
`int b = 20 ;`

2 & 3 `c = a + b`

4. `print c` or `print (a+b);`

Lecture 2

- Vs Code
 - IntelliJ
- } Install

Next we will learn

- First Java Program
- flow of the program
- Java Intro

- methods (function),