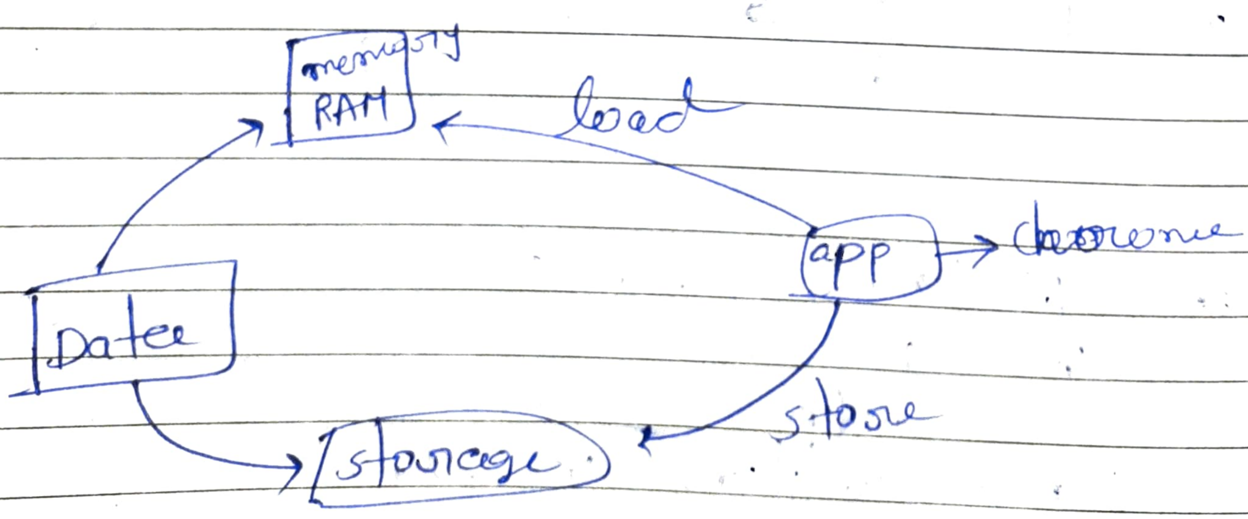
* storage and memory

- storage ni data parmenent rey
- while in memory pc sutdown it will deleted
  (RAM) Memory

→ the need of memory

- in storage there is ~~show~~ slow to read & write data
- But memory is faster for read and write data
- so that for read & write data use memory and then save it back to storage.

→ application - chrom



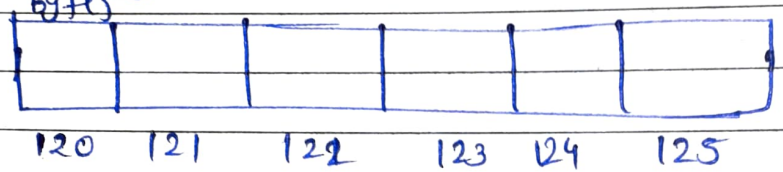
* data structure

* Array & memory

int a = 1 32 bits for int

1 = 00000000000000000000000000000001

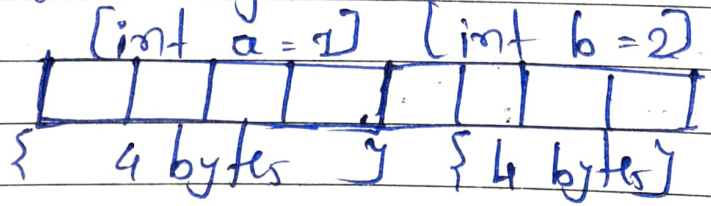
memory = store bytes



1 byte = 8 bits
= small unit of data

32 bits int માટે જોય
તો 8 bits = 1 byte
4 x 8 = 32
4 bytes

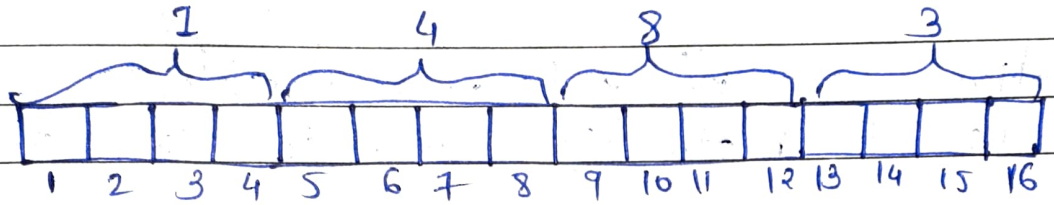
in memory



DATE: / / PAGE NO:

Array :-

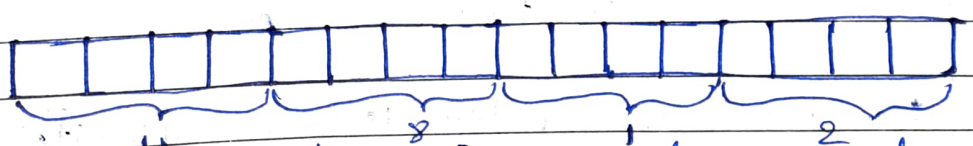
```
int a = 1;
int sarray[2] = {4, 8};
int c = 3;
```



If we add array[4] = {4, 8, 1, 2}

we can't add directly
because after the array it store the value of next variable.

- So the solution it create the big memory



→ copy 4 and 8 from above and add 1 & 2.
→ if add more value again then repeat again the ~~next~~ process.

= python list, java ArrayList....

* → Data structure :-
different ways of storing data on your computer

→ Algorithm :-

operations on different data structure
+ sets of instructions for executing them

- Common data structures
  - linked lists
  - Array
  - stack
  - Queue
  - Maps
  - Hash tables
  - Search trees .....

- Common categories of algorithm
  - search
  - sorting
  - tree traversing
  - Hashing & regex (string pattern matching)

- why it important ?
  - to understand organizing principles behind web development and programming work

DATE: / / PAGE NO:

class

- A class is user defined blue print or prototype from with objects are created.
- It represents the set of properties or methods that are common to all objects of one type.

→ Multiple objects

```
class Animal { }

class dog extends Animal { }
class cat extends Animal { }

public class Test
{
    // using dog object
  Animal obj = new Dog();
  Animal obj = new Cat();
}
```

⇒ By constructor

```
class Robot {
  string name;
  string colour;
  int weight;
}
```

```
Robot (string n, string c, int w) {
    this.name = n;
    this.colour = c;
    this.weight = w;
}
```

```
Robot r1 = new Robot ("Tom", "red", 30);
Robot r2 = new Robot ("Jerry", "blue", "40");
```

* OOPs (object-oriented programming system)

→ object means real-coord entity such as a pen, chair, table, computer, watch, etc.

→ object-oriented programming is a methodology or paradigm to design a program using classes and objects.

→ Concepts :- object
class
Inheritance
polymorphism
abstraction
Encapsulation

→ Object

- Any entity that has state and behavior is known as an object.
  Ex:- chair, pen, table, keyboard, bike, etc.
  It can be physical or logical.
- An object can be defined as an instance of a class.
- It contains an address and takes up some space in memory.
- object can ~~contains an~~ communicate without knowing the details of each other's data or code.

- The only necessary thing is the type of message accepted and the type of response returned by the objects.

- A dog is an object because it has state like color, name, breed, etc. as well as behaviors like wagging the tail, barking, eating, etc.

- object is an instance (result) of a class.
- object is a real-world entity.
- it is a runtime entity.
- it is an entity which has state & behavior.

- new keyword is used to allocate memory at runtime. all object get memory in heap memory area.

→ Class

→

- class has an group of object which have common properties.

- It is a template or blueprint from which objects are created.

- It is a logical entity. It can't be physical.

a class in java can contain:

- fields
- Methods
- Constructors
- Blocks
- Nested class and interface

- object is the concept which represents the class. with the help of a new operator we may easily create object of class and memory is created in the heap and object is called an instance of class.

- Real time Ex:- If animal is class then dog is the object, if human is the class then man is the object.