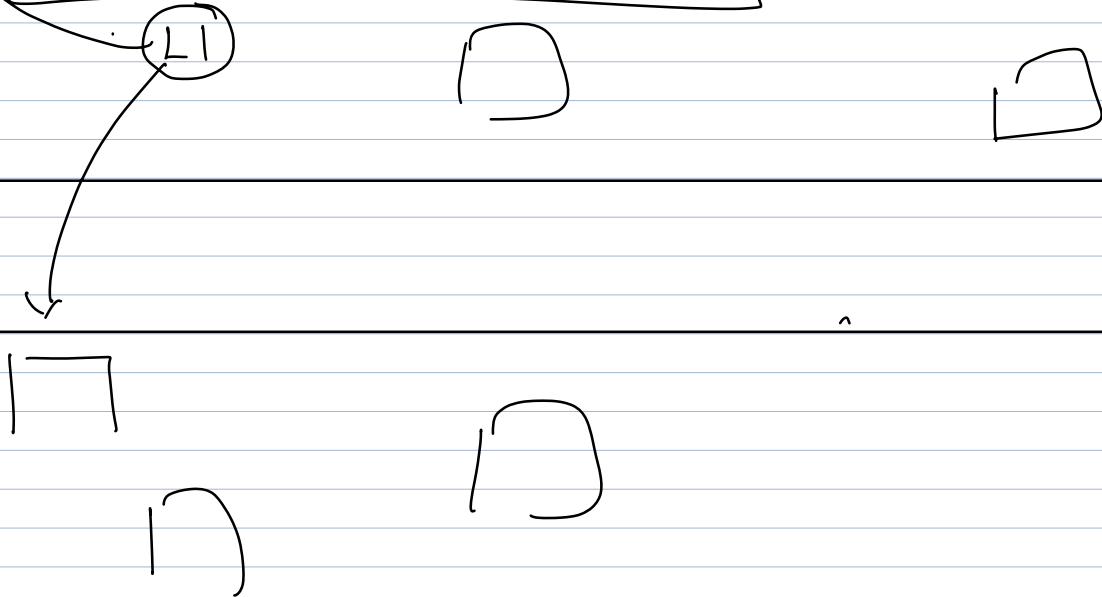


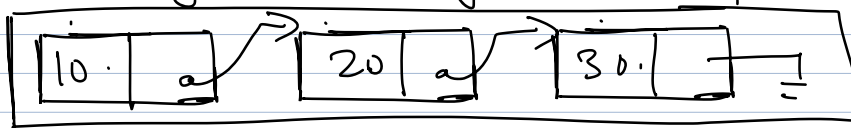
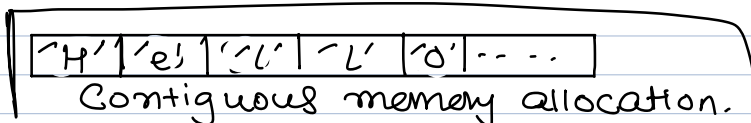
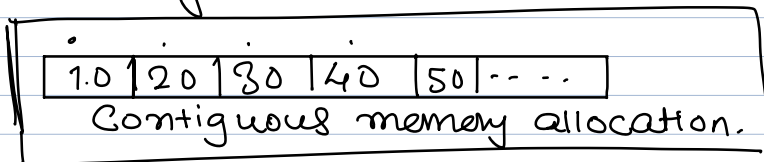
ROAD

Line 21abullat Agency



Bungalow \equiv 105 integer value / 105 char value.

Bungalow \equiv vector, string, list
Colony



Line
21abullat
Agency

line 214: Agency = (iterator class)

iterator object

Vector::iterator iter;

String::iterator iter;

List::iterator iter;

```
int sum_array(int a[], int N)
{
    int sum = 0;
    int i;
    for (i = 0; i < N; ++i)
        sum = sum + a[i];
    return (sum);
}
```

Array is a data-type which can hold fixed number of instances of any data type in a contiguous manner.

(Array) itself is a data-type which can hold instances of other data types using some storage strategy.

Data Structure

i) Representing data (X)

ii) Arranging data (eiser Bas)

debit card \rightarrow Real entity

\downarrow

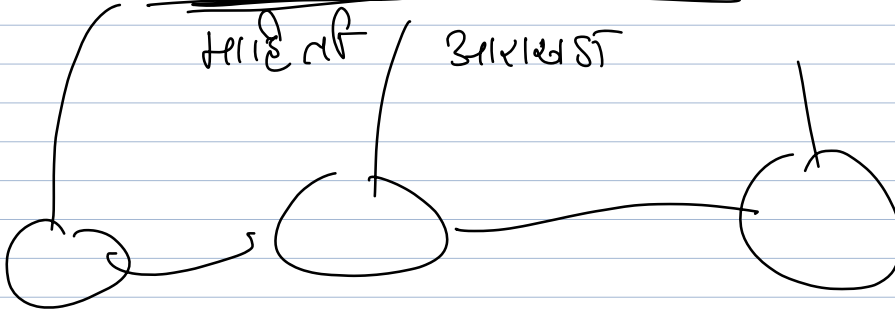
computer memory

Struct

class

Information Structure:

hierarch / linked list



list \times L;

\rightarrow contains data type

\rightarrow defines a storage strategy to store multiple instances of 'any' data type!

list <int> L;

Concrete data type

```
class  
{  
    private  
    int  
}
```

Contained type commit
class template

| —

Vector<int>::iterator iter;

string::iterator iter