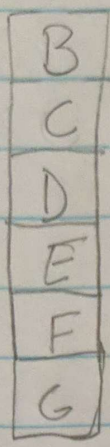
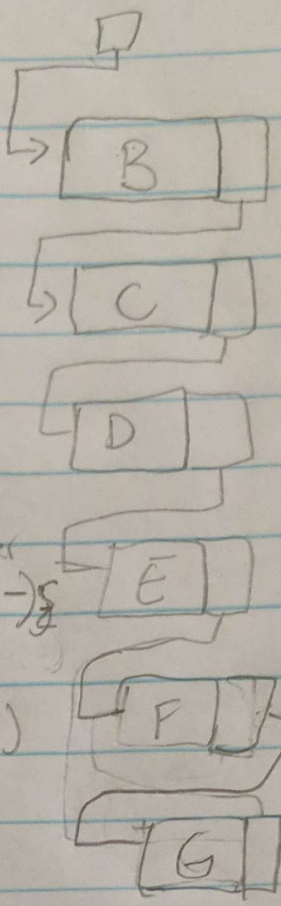
2)





5) Array based = O(n)

```
for(int i=0; i<arr.length; i++){
    System.out.print(arr[i]+" ");
}
```

linked list = O(n)

```
for(int i=0; i < list.size(); i++){
    System.out.Print(list.get(i));
}
```

6) enter number

```
for(int i=arr.length-1; i>#; i--){
    arr[i] = arr[i-1];
}
arr[#] = "word"; ⇒ O(n)
```

→ `list.add(i, "word");`
↳ O(1)

7) `Arr[i] = stuff;` O(1)
`list.add(stuff);` (O1)

8) `arr[6] = "H";` (Last) O(1)
`list.Push("H");` O(1);

9) enter index

```
for(int i = index; i < arr.length - 1; i++){ } ⇒ O(n)
    arr[i] = arr[i+1];
}
```

⇒ `list.remove("word");` ⇒ O(1)

10) `arr.remove(Lastindex);` O(1)

`list.POP();` O(1)