ArrayLevel-2

Q.1 Find Unique elements

i/p →

_ , _ , _ , _ , _ , _ _

2, 10, 11, 13, 10, 2, 15, 13, 15

↑

Repeats one time.

How to solve?

⇒ XOR

for same value → 0
for diff value → 1

Code:

```
int getUnique (int arr[], int n) {
    int ans = 0;
   for( int i = 0; i<n; i++) {
     ans = ans ^ arr[i];
   }
   return ans;
}
```

H.W ⇒ Pair sum/
Two sum

Q2) Find pairs of given array.

i/p → array → { }

{10, 20, 30}

o/p ⇒

(10,10)   (20,10)   (30,10)
(10,20)   (20,20)   (30,20)
(10, 30)   (20,30)   (30,30)

Using 2loops i & j

code:

```
for (int i=0; i<n; i++){
     for (int j=0; j<n; j++){
   cout<<arr[i] <<", "<<arr[j]<<endl;
   }
}
```

Q.3

i/p → array → [1, 2, 3, 4]

Print all triplets.

3 loops we will need.

Code:

```
for(int i=0; i<n; i++) {
   for (int j=0; j<n; j++) {
      for (int k=0; k<n; k++) {
     cout<< arr[i] <<"  "<<arr[j]
                  <<  "  " <<arr[k]
                              <<endl;
     }
} }
```

Q. 4
Sort 0's & 1's

i/p → 0 1 0 1 1 0 0 0 0

o/p → 0 0 0 0 0 0 1 1 1

Methods:-

1. Counting and Fill
2. 2 pointer Approach → H.W
3. sort() → incremently

1st Approach

```
void sortZeroOne( int arr[], intn){
   int zeroCount=0;
   int one count =0;
  //step A : count 0 and 1
  for( int i=0 ; i<n ; i++){
      if (arr[i] ==0){
         zeroCount++;
  }
      if (arr[i] ==1) {
         one count++ ;
    }
 //step B: place all zeros first
  int i;
for( int i=0; i< zerocount ; i++ ) {
      arr[i] =0;
  }
// Place all ones first
for(int i=i; i<n; i++) {
    arr[i] = 1;
  }
}
```

```
//EASY WAY
int index = 0;

while(zerocount--) {
    arr[index] = 0;
    index++;
}
while(oneCount--) {
    arr[index]=1;
    index++;
}
```

Shift Arrays elements by 1.

i/p → 10 20 30 40 50 60

o/p → 60 10 20 30 40 50

Method to solve.

① temp = 60 (temp= arr[n-1])

② arr[i+1] = arr[i];

③ arr[0] = temp

H.W ⇒ Left shift

code :

```
void shiftArray( int arr[], int n) {

    //step 1
    int temp = arr[n-1];

    //step 2
    // shift → arr[i] = arr[i-1]
  for( int i = n-1 ; i>=1 ;i--) {
        arr [i] = arr[i-1];
  }

    // step3 copy temp into 0th index
}   arr[0]= temp;
```

10 | 20| 30|40 |50| 60

K = 2

50| 60| 10|20|30|40

① temp = { 50, 60}

② arr[i+2] = arr[i]

③ include temp elements

H.W ↗