# Solve Array

$n = 5$

num =

| 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 10 | 20 | 30 | 40 | 50 |

index =

| 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 3 | 0 | 4 | 1 | 2 |

output

target =

| 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| 20 | 40 | 50 | 10 | 30 |

Exp:- target[i] = num[index[i]];

val = num[i]
idx = index[i]

$i = 0$, val = 10
idx = 3

$i = 1$, val = 20
idx = 0

$i = 2$, val = 30
idx = 4

$i = 3$, val = 40
idx = 1

$i = 4$, val = 50
idx = 2

Code

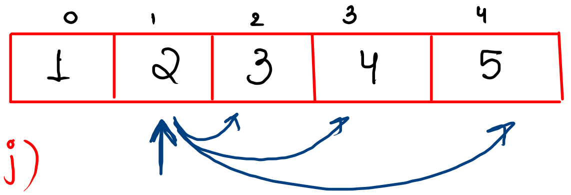
```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[] nums = new int[n];
    for (int i = 0; i < n; i++) {
        nums[i] = scn.nextInt();
    }
    int[] index = new int[n];
    for (int i = 0; i < n; i++) {
        index[i] = scn.nextInt();
    }
    int[] ans = solveArray(n, nums, index);
    for (int i = 0; i < n; i++) {
        System.out.print(ans[i] + " ");
    }
}
public static int[] solveArray(int n, int[] nums, int[] index) {
    int[] target = new int[n];
    for (int i = 0; i < n; i++) {
        int val = nums[i];
        int idx = index[i];

        target[idx] = val;
    }
    return target;
}
```

## → Variation of nested loop

## Print Pair

$n = 5$

arr =



(i, j)

pairs

1, 2
1, 3
1, 4
1, 5

2, 3
2, 4
2, 5

3, 4
3, 5

4, 5

code

```
for (int i=0 ; i < n ; i++){
    for (int j= i+1 ; j<n ; j++){
        print arr[i], arr[j]
    }
}
```

## Theory

- Combination :- when we move only in forward direction
  - comb. with repetation [c w r]
  - comb. without repetation [c wt r]
- Permutation :- when we move only in both direction
  - perm. with repetation [p w r]
  - permu. without repetation [p wt r]

Ex:- 1, 2, 3, 4

C wt r  (1,2), (1,3)(1,4)(2,3), (2,4)(3,4)

C w r  (1,1) (1,2), (1,3) (1 4), (2,2) (2,3)(2,4) (3,3) (3,4)
(4,4)

P wt r  (1, 2) (1, 3)(1, 4) (2,1) (2, 3)(2,4)(3,1)(3, 2)
(3,4)(4,1)(4,2)(4,3)

P w r  (1, 2) (1, 3)(1, 4) (2,1) (2, 3)(2,4)(3,1)(3, 2)
(3,4)(4,1)(4,2)(4,3) | (1,1) (2,2)(3,3)(4,4)

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[] arr = new int[n];
    for (int i = 0; i < n ; i++) {
        arr[i] = scn.nextInt();
    }

    printPair(arr, n);
}
public static void printPair(int[] arr, int n) {

    // comb without repe
    for (int i = 0; i < n; i++) {
        for (int j = i + 1; j < n; j++) {
            System.out.println(arr[i] + " " + arr[j]);
        }
    }

}
```

## Coding part :-

1) Combination without repetation :- $i = 0$
$j = i+1$

2) Combination with repetation :- $i = 0$
$j = i$

3) Permutation with repetation :- $i = 0$
$j = 0$

4) Permutation without repetation :- $i = 0$
$j = 0$
if ( i != j )

# Find all Combination

(Comb. with repe)

n = 5

arr =

| 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 |

target = 8

Ex:- arr[i] + arr[j] == target

loops

```
for(int i=0; i<n; i++){
    for(int j=i; j<n; j++){
        create sum with value arr[i] + arr[j]
        check if sum == target
            then print that pair
    }
}
```

# Code

$arr = [\overset{0}{1}\ \overset{1}{2}\ \overset{2}{3}\ \overset{3}{4}\ \overset{4}{5}]$ target = 8

$\overset{\uparrow\uparrow}{ij}$

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[] arr = new int[n];
    for (int i = 0; i < n ; i++) {
        arr[i] = scn.nextInt();
    }
    int target = scn.nextInt();
    findAllComb(arr, n, target);
}
public static void findAllComb(int[] arr, int n, int target) {

    // comb with repe
    for (int i = 0; i < n; i++) {
        for (int j = i; j < n; j++) {
            int sum = arr[i] + arr[j];
            if ( sum == target ) {
                System.out.println(arr[i] + " " + arr[j]);
            }
        }
    }

}
```

o/p

```
3 5
4 4
```

i=0, j=0 sum = 2
j=1 sum = 3
j=2 sum = 4
j=3 sum = 5
j=4 sum = 6

---

i=1, j=1, sum=4
j=2 sum=5
j=3 sum=6
j=4 sum=7

---

i=2, j=2 sum=6
j=3 sum=7
j=4 sum=8

---

i=3, j=3, sum=8
j=4, sum=9

---

i=4, j=4, sum=10

# Greater Than Me

$$\text{arr} \Rightarrow \begin{bmatrix} 5 & 2 & 3 & 1 & 5 & 4 \end{bmatrix}$$

$$\begin{tikzcd}
0 \arrow[d] & 1 \arrow[d] & 2 \arrow[d] & 3 \arrow[d] & 4 \arrow[d] & 5 \arrow[d] \\
0 & 4 & 3 & 5 & 0 & 2
\end{tikzcd}$$

Permutation