

# FUNCTION AND ARRAYS

## Digit Frequency

Easy

1. Each digit (0 to 9) denotes the student of the Optica Student community.
2. You are given a number  $n$  where  $i$ th digit denotes that  $i$ th task that is assigned to the corresponding digit student.
2. You are given a digit  $d$  denotes a student.
3. You are required to calculate the frequency of digit  $d$  in number  $n$  or how many tasks are assigned to student  $d$ .

### Constraints

$$0 \leq n \leq 10^9 \quad 0 \leq d \leq 9$$

### Format

#### Input

A number  $n$  A digit  $d$

#### Output

A number representing frequency of digit  $d$  in number  $n$

### Example

#### Sample Input

994543234

4

#### Sample Output

3

```
#include<iostream>
#include<cmath>
using namespace std;
int digFreq(int n, int d) {
    //write your code here
    int freq{0};
    int nod{0};
    int num{n};
    //finding no digits in n
```

```


while(num!=0){
    num/=10;
    nod++;
}
int p{};
int r{n};
int k{};
p=pow(10,(nod-1));
while (r!=0){
    k=r/p;
    r= r%p;
    p/=10;
    if(k==d){
        freq++;
    }
}
return freq;
}
int main() {
    int n, d;
    cin >> n >> d;
    int res = digFreq(n, d);
    cout << res << endl;
}

```


## Introduction To Number System

---

LEVEL-1



# INTRODUCTION TO NUMBER SYSTEM



**Sumeet Malik**

For better experience visit

🔗 [www.nados.pepcoding.com](http://www.nados.pepcoding.com)

# Decimal To Any Base

Easy

1. You are given a decimal number n.
2. You are given a base b.
3. You are required to convert the number n into its corresponding value in base b.

## Constraints

$0 \leq d \leq 512$

$2 \leq b \leq 10$

```
#include<iostream>
```

```
#include<cmath>
```

```
using namespace std;
int DecToAny(int n, int b){
//write your code here
    int q{n};
    int r{};
    int p{};
    int new_num{};
    while (q!=0){
        r =q%b;
        q = q/b;
        new_num += (r* pow(10,p));
        p++;
    }
    return new_num;
}
int main(){
    int n;
    int b;
    cin >> n;
    cin >> b;
    int res = DecToAny(n,b);
    cout << res << endl;
    return 0;
}
```

# Any Base To Decimal

Easy

1. You are given a number n.
2. You are given a base b. n is a number on base b.
3. You are required to convert the number n into its corresponding value in decimal number system.

## Constraints

$0 \leq d \leq 10000000000$

## Sample Input

```
57
2
```

## Sample Output

```
111001
```

## Example

### Sample Input

```
111001
2
```

### Sample Output

```
57
```

```

2 <= b <= 10
#include<iostream>
#include<cmath>
using namespace std;
int AnyToDec(int n,int b){
//write your code here.
    int q{n};
    int r{};
    int p{};
    int new_num{};
    while (q!=0){
        r =q%10;
        q = q/10;
        new_num += (r* pow(b,p));
        p++;
    }
    return new_num;
}
int main(){
    int n;
    int b;
    cin >> n;
    cin >> b;
    int res = AnyToDec(n,b);
    cout<<res<<endl;
}

```

## Any Base To Any Base

Easy

1. You are given a number n.
2. You are given a base b1. n is a number on base b.
3. You are given another base b2.
4. You are required to convert the number n of base b1 to a number in base b2.

### Constraints

0 <= n <= 512

2 <= b1 <= 10

2 <= b2 <= 10

### Format

#### Input

A number n

A base b1

A base b2

#### Output

A number of base b2 equal in value to n of base b1.

## Example

### Sample Input

111001

2

3

### Sample Output

2010

```
#include<iostream>
#include <cmath>
using namespace std;
int any_to_deci(int n,int b1){
    int q{n};
    int r{};
    int p{};
    int new_num_a{};
    while (q!=0){
        r =q%10;
        q = q/10;
        new_num_a += (r* pow(b1,p));
        p++;
    }
    return new_num_a;
}
int deci_to_any(int n,int b2){
    int q{n};
    int r{};
    int p{};
    int new_num_b{};
    while (q!=0){
        r =q%b2;
        q = q/b2;
        new_num_b += (r* pow(10,p));
        p++;
    }
    return new_num_b;
}
int AnyToAny(int n, int b1, int b2){
    //write your code here
    int s{any_to_deci(n,b1)};
    int new_num =deci_to_any(s,b2);
    return new_num;
}
int main(){
    int n;
    int b1;
    int b2;
    cin >> n;
    cin >> b1;
    cin >> b2;
    int res = AnyToAny(n,b1,b2);
    cout << res <<endl;
}
```

# Any Base Addition

Easy

1. You are given a base  $b$ . 2. You are given two numbers  $n1$  and  $n2$  of base  $b$ . 3. You are required to add the two numbers and print their value in base  $b$ .

## Constraints

$2 \leq b \leq 10$   $0 \leq n1 \leq 256$   $0 \leq n2 \leq 256$

## Format

### Input

A base  $b$  A number  $n1$  A number  $n2$

### Output

A number representing the sum of  $n1$  and  $n2$  in base  $b$ .

## Example

### Sample Input

8  
777  
1

### Sample Output

1000

```
#include<iostream>
#include<cmath>
using namespace std;
// int any_to_deci(int n,int b){
//     int q{n};
//     int r{};
//     int p{};
//     int new_num_a{};
//     while (q!=0){
//         r =q%10;
//         q = q/10;
//         new_num_a += (r* pow(b,p));
//         p++;
//     }
//     return new_num_a;
// }
// int deci_to_any(int n,int b){
//     int q{n};
```

```

//      int r{};
//      int p{};
//      int new_num_b{};
//      while (q!=0){
//          r =q%b;
//          q = q/b;
//          new_num_b += (r* pow(10,p));
//          p++;
//      }
//      return new_num_b;
// }
// int getSum(int b, int n1, int n2) {
//     // write your code here
//     int dn1{any_to_deci(n1,b)};
//     int dn2{any_to_deci(n2,b)};
//     int temp{dn1+dn2};
//     int result{deci_to_any(temp,b)};
//     return result;

// }
/////another solution
int getSum(int b, int n1, int n2) {
    //      // write your code here
    int c{};
    int result{};
    int p=1;
    while(n1>0 || n2>0 || c>0){
        int d1 =n1%10;
        int d2 =n2%10;
        n1 /=10;
        n2 /=10;
        int d =d1+d2+c;
        c=d/b;
        int k =d%b;
        result += (k*p);
        p*=10;
    }
    return result;
}

int main() {
    int b, n1, n2;
    cin >> b >> n1 >> n2;
    cout << getSum(b, n1, n2) << endl;
}

```

## Any Base Subtraction

Easy

1. You are given a base b.

2. You are given two numbers n1 and n2 of base b.
3. You are required to subtract n1 from n2 and print the value.

### Constraints

$2 \leq b \leq 10$

$0 \leq n1 \leq 256$

$n1 \leq n2 \leq 256$

### Format

#### Input

A base b

A number n1

A number n2

#### Output

A number of base b equal in value to  $n2 - n1$ .

### Example

#### Sample Input

8  
1  
100

#### Sample Output

77

```
#include<iostream>
using namespace std;
```

```
int sub_them(int n1, int n2,int b){
    int c{};
    int result{};
    int p=1;
    while(n2>0){
        int d1 =n1%10;
        int d2 =n2%10;
        n1 /=10;
        n2 /=10;
        int d{};
        d2=d2+c;
        if(d2>=d1){
            c=0;
            d=d2-d1;
        }else{
            c=-1;
            d=d2+b-d1;
        }
        result += (d*p);
        p*=10;
    }
    return result;
}
```

```
int main(){
    //cout<<"For Addition"<<endl;
```



```

        //cout<<"Enter their base: ";
        int b{};
        cin >>b;
        //cout<<"Enter the first number: ";
        int n1{};
        cin>>n1;
        //cout<<"Enter the second number: ";
        int n2{};
        cin>>n2;

        //subtracting n1 from n2
        cout<<sub_them(n1,n2,b);

        return 0;
}

```

## Any Base Multiplication

Easy

1. You are given a base b.
2. You are given two numbers n1 and n2 of base b.
3. You are required to multiply n1 and n2 and print the value.

### Constraints

$2 \leq b \leq 10$

$0 \leq n1 \leq 10000$

$0 \leq n2 \leq 10000$

### Format

#### Input

A base b

A number n1

A number n2

#### Output

A number of base b equal in value to  $n2 * n1$ .

### Example

#### Sample Input

```

5
1220
31

```

#### Sample Output

```

43320
#include<iostream>
using namespace std;
int add_them(int b,int n1, int n2){
    int c{};
    int result{};
    int p=1;
    while(n1>0 || n2>0 || c>0){

```

```

    int d1 =n1%10;
    int d2 =n2%10;
    n1 /=10;
    n2 /=10;
    int d =d1+d2+c;
    c=d/b;
    int k =d%b;
    result += (k*p);
    p*=10;
}
return result;

}
int product_with_single_digit(int b ,int n1, int d2){
    int r1{};
    int c{};
    int p{1};
    while(n1>0||c>0){
        int d1=n1%10;
        n1/=10;
        int d=((d1*d2)+c)%b;
        c=((d1*d2)+c)/b;
        r1+=d*p;
        p *=10;
    }
    return r1;
}
int multiply_them(int b,int n1,int n2){
    int result{};
    int p=1;
    while(n2>0){
        int d2=n2%10;
        n2/=10;
        int rr =product_with_single_digit( b , n1, d2);
        result=add_them(b,result,(rr*p));
        p*=10;
    }
    return result;
}

int main(){
    //cout<<"For multiplication"<<endl;
    //cout<<"Enter their base: ";
    int b{};
    cin >>b;
    //cout<<"Enter the first number: ";
    int n1{};
    cin>>n1;
    //cout<<"Enter the second number: ";

```

```
int n2{};  
cin>>n2;
```

```
cout<<multiply_them(b,n1,n2);
```

```
return 0;
```

```
}
```

## Introduction To Array In Java | Hindi



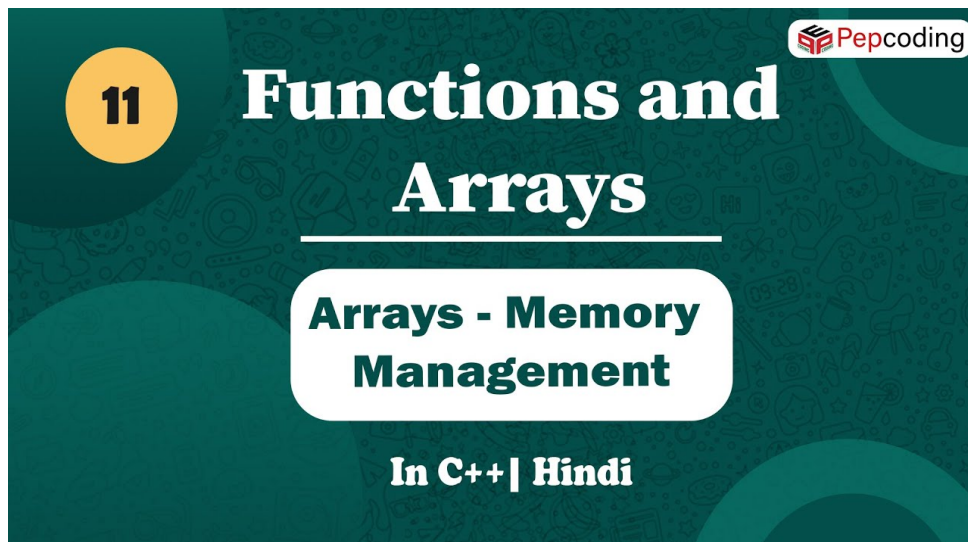
The thumbnail features a dark background with binary code. On the left, a progress bar shows five segments, with the first four filled. Below it, 'LEVEL-1' is written. The main title 'INTRODUCTION TO ARRAYS' is in large, bold, white and blue letters. On the right, there is a circular portrait of a man with arms crossed, wearing a black t-shirt with 'PEPCODER' on it. The 'Pepcoding' logo is in the top right corner. A light blue banner at the bottom contains the text 'For better experience visit' followed by a link icon and 'www.nados.pepcoding.com'.

LEVEL-1

# INTRODUCTION TO ARRAYS

For better experience visit [www.nados.pepcoding.com](http://www.nados.pepcoding.com)

## Arrays Memory Management In C++ | Hindi



The thumbnail has a dark green background with a pattern of small icons. In the top right corner is the 'Pepcoding' logo. On the left, a yellow circle contains the number '11'. The main title 'Functions and Arrays' is in large, bold, white letters. Below it, a white rounded rectangle contains the text 'Arrays - Memory Management' in bold. At the bottom, 'In C++ | Hindi' is written in white.

11

# Functions and Arrays

## Arrays - Memory Management

In C++ | Hindi

# Span Of Array

Easy

1. You are given a number n, representing the count of elements.
2. You are given n numbers.
3. You are required to find the span of input. Span is defined as difference of maximum value and minimum value.

## Constraints

$1 \leq n \leq 10^4$   
 $0 \leq n1, n2$   
.. n elements  $\leq 10^9$

## Format

### Input

A number n

n1

n2

.. n number of elements

### Output

A number representing max - min

## Example

### Sample Input

6  
15  
30  
40  
4  
11  
9

### Sample Output

36

```
#include<iostream>
using namespace std;
```

```
int main(){
    //write your code here
    int s{};
    //cout<<"Enter the size of the array: ";
    cin>>s;

    int* array= new int [s];
    for (int i{};i<s;i++){
        //cout<<"Enter "<<i+1<<" Number: ";
        cin>>array[i];
    }
    //finding minimum
    int min{array[0]};
    for(int i{};i<s;i++){
        if(array[i]<min){
            min=array[i];
        }
    }
}
```

```

    }
    //finding maximum
    int max{array[0]};
    for(int i{};i<s;i++){
        if(array[i]>max){
            max=array[i];
        }
    }
    cout<<max-min<<endl;
    //cout<<"The span of the array is "<<max-min<<endl;
    return 0;}

```

## Find Element In An Array

Easy

- 1.You are given a number n, representing the size of array a.
- 2.You are given n distinct numbers, representing elements of array a.
3. You are given another number d.
4. You are required to check if d number exists in the array a and at what index (0 based). If found print the index, otherwise print -1.

### Constraints

$1 \leq n \leq 10^7$   
 $-10^9 \leq n1, n2$   
 $\dots n \text{ elements} \leq 10^9$   
 $-10^9 \leq d \leq 10^9$

### Format

#### Input

A number n  
 n1  
 n2  
 .. n number of elements  
 A number d

#### Output

A number representing index at which d is found in array a and -1 if not found

### Example

#### Sample Input

6  
 15  
 30  
 40  
 4  
 11  
 9  
 40

#### Sample Output

2  

```

#include<iostream>
using namespace std;

```

```

int main(){
    //write your code here

    int s{};
    //cout<<"Enter the size of the array: ";
    cin>>s;

    int* array= new int [s];
    for (int i{};i<s;i++){
        //cout<<"Enter "<<i+1<<" Number: ";
        cin>>array[i];
    }
    //cout<<"Enter the number you want to find in array: ";
    int d{};
    cin>>d;

    int c{};
    for(int i{};i<s;i++){
        if(array[i]==d){
            cout<<i<<endl;
            c++;
            break;
        }
    }
    if(c==0){
        cout<<-1<<endl;
    }
    return 0;
}

```

## Bar Chart

Easy

1. You are given a number n, representing the size of array a.
2. You are given n numbers, representing elements of array a.
3. You are required to print a bar chart representing value of arr a.

### Constraints

1 <= n <= 30

0 <= n1, n2, .. n elements <= 10

### Format

#### Input

A number n

n1

n2

.. n number of elements

#### Output

A bar chart of asteriks representing value of array a

### Example

#### Sample Input

5  
3  
1  
0  
7  
5

### Sample Output

		*	
		*	
		*	*
		*	*
*		*	*
*		*	*
*	*	*	*

```
#include<iostream>
```

```
using namespace std;
```

```
int main(){
    //write your code here
    int s{};
    //cout<<"Enter the size of the array: ";
    cin>>s;

    int* array= new int [s];
    for (int i{};i<s;i++){
        // cout<<"Enter "<<i+1<<" Number: ";
        cin>>array[i];
    }
    int max{array[0]};
    for(int i{};i<s;i++){
        if(array[i]>max){
            max=array[i];
        }
    }

    for(int i{max};i>0;i--){
        for(int j{} ;j<s;j++){
            if(array[j]==i){
                cout<<"*\t";
                array[j]--;
            }else{
                cout<<"\t";
            }
        }
        cout<<endl;
    }
    return 0;
}
```

# Sum Of Two Arrays

Easy

1. You are given a number  $n1$ , representing the size of array  $a1$ .
2. You are given  $n1$  numbers, representing elements of array  $a1$ .
3. You are given a number  $n2$ , representing the size of array  $a2$ .
4. You are given  $n2$  numbers, representing elements of array  $a2$ .
5. The two arrays represent digits of two numbers.
6. You are required to add the numbers represented by two arrays and print the arrays.

## Constraints

$1 \leq n1, n2 \leq 100$

$0 \leq a1[i], a2[i] < 10$

## Format

### Input

A number  $n1$

$n1$  number of elements line separated

A number  $n2$

$n2$  number of elements line separated

### Output

A number representing sum of two numbers, represented by two arrays.

## Example

### Sample Input

```
5
3
1
0
7
5
6
1
1
1
1
1
1
1
1
```

### Sample Output

```
1
4
2
1
8
6
```

```
#include<iostream>
using namespace std;
```

```
int main(){
    int s1{};
    //cout<<"Enter the size of the array1: ";
```



```

cin>>s1;
int *arr1= new int[s1];
for (int i{};i<s1;i++){
    //cout<<"Enter "<<i+1<<" Number: ";
    cin>>arr1[i];
}

int s2{};
//cout<<"Enter the size of the array2: ";
cin>>s2;
int *arr2 = new int [s2];
for (int i{};i<s2;i++){
    //cout<<"Enter "<<i+1<<" Number: ";
    cin>>arr2[i];
}
// int diff =(s1>s2)?s1-s2:s2-s1;
// if(s1>s2){
//     int t{};
//     int c{};
//     for(int i{s2-1};i>=0;i--){
//         t=arr1[i+diff]+arr2[i]+c;
//         c=t/10;
//         arr1[i+diff] = t%10;
//     }
//     for(int i{diff-1};i>=0;i--){
//         t = arr1[i]+c;
//         c=t/10;
//         arr1[i]= t%10;
//     }
//     if(c!=0){
//         cout<<c<<endl;
//     }
//     for(int i{};i<s1;i++){
//         cout<<arr1[i]<<endl;
//     }
// }else{
//     int t{};
//     int c{};
//     for(int i{s1-1};i>=0;i--){
//         t=arr2[i+diff]+arr1[i]+c;
//         c=t/10;
//         arr2[i+diff] = t%10;
//     }
//     for(int i{diff-1};i>=0;i--){
//         t = arr2[i]+c;
//         c=t/10;
//         arr2[i]= t%10;
//     }
//     if(c!=0){
//         cout<<c<<endl;
//     }
// }

```

```

//      for(int i{};i<s2;i++){
//          cout<<arr2[i]<<endl;
//      }
// }
////second solution pepcoding
int s3{(s1>s2)? s1:s2};
int* arr3 = new int [s3];
int i= s1-1;
int j= s2-1;
int k= s3-1;
int c1{};

while(k>=0){
    int d{c1};
    if(i>=0){
        d+=arr1[i];
    }
    if(j>=0){
        d+=arr2[j];
    }
    c1=d/10;
    d=d%10;
    arr3[k]=d;
    i--;
    j--;
    k--;
}
if (c1!=0){
    cout<<c1<<endl;
}
for(int i{};i<s3;i++){
    cout<<arr3[i]<<endl;
}
return 0;
}

```

## Difference Of Two Arrays

Easy

1. You are given a number  $n_1$ , representing the size of array  $a_1$ .
2. You are given  $n_1$  numbers, representing elements of array  $a_1$ .
3. You are given a number  $n_2$ , representing the size of array  $a_2$ .
4. You are given  $n_2$  numbers, representing elements of array  $a_2$ .
5. The two arrays represent digits of two numbers.
6. You are required to find the difference of two numbers represented by two arrays and print the arrays.  $a_2 - a_1$

Assumption - number represented by  $a_2$  is greater.

### Constraints

$1 \leq n_1, n_2 \leq 100$   
 $0 \leq a_1[i], a_2[i] < 10$

number represented by a1 is smaller than number represented by a2

## Format

### Input

A number n1

n1 number of elements line separated

A number n2

n2 number of elements line separated

### Output

A number representing difference of two numbers ( $a2 - a1$ ), represented by two arrays.

## Example

### Sample Input

```
3
2
6
7
4
1
0
0
0
```

### Sample Output

```
7
3
3
```

```
#include<iostream>
#include<vector>
using namespace std;

int main(){
    int s1{};
    // cout<<"Enter the size of the array1: ";
    cin>>s1;
    int *arr1= new int[s1];
    for (int i{};i<s1;i++){
        //cout<<"Enter "<<i+1<<" Number: ";
        cin>>arr1[i];
    }

    int s2{};
    //cout<<"Enter the size of the array2: ";
    cin>>s2;
    int *arr2 = new int [s2];
    for (int i{};i<s2;i++){
        //cout<<"Enter "<<i+1<<" Number: ";
        cin>>arr2[i];
    }
    //a2 is greater than a1(number represented by them)

    int *arr3 = new int[s2];
```

```

int i = s1-1;
int j = s2-1;
int k = s2-1;
int c{};
while(k>=0){
    if(i>=0){
        if((arr2[j]+c)>=arr1[i]){
            arr3[k]=arr2[j]+c-arr1[i];
            c=0;
        }else{
            arr3[k] = arr2[j]+c+10-arr1[i];
            c=-1;
        }
    }else{
        if(arr2[j]+c>=0){
            arr3[k]=arr2[j]+c;
            c=0;
        }else{
            arr3[k]=arr2[j]+c+10;
            c=-1;
        }
    }
    i--;
    j--;
    k--;
}

int z{0};
for(int a{};a<s2;a++){
    if((arr3[a]!=0)|| (z!=0)){
        cout<<arr3[a]<<endl;
        z=1;
    }
}

return 0;
}

```

## Reverse An Array

Easy

1. You are given a number n, representing the size of array a.
2. You are given n numbers, representing elements of array a.
3. You are required to reverse the contents of array a.

### Constraints

$0 \leq n < 10^4$

$-10^9 \leq a[i] \leq 10^9$

### Format

## Input

Input is managed for you

## Output

Output is managed for you

## Example

### Sample Input

```
5
1
2
3
4
5
```

### Sample Output

```
5 4 3 2 1
```

```
#include<iostream>
```

```
using namespace std;
```

```
void reverse(int* arr, int n){
```

```
    // write your code here
```

```
////my solution
```

```
    // int*array= new int[n];
```

```
    // int n1{n};
```

```
    // for(int i{};i<n;i++){
```

```
        // array[i]=arr[n1-1];
```

```
        // n1--;
```

```
    // }
```

```
    // for(int i{};i<n;i++){
```

```
        // arr[i]=array[i];
```

```
    // }
```

```
////pepcoding solution
```

```
    int i=0;
```

```
    int j{n-1};
```

```
    while(i<j){
```

```
        int temp = arr[i];
```

```
        arr[i]=arr[j];
```

```
        arr[j]=temp;
```

```
        j--;
```

```
        i++;
```

```
    }
```

```
}
```

```
void display(int* arr, int n){
```

```
    for(int i = 0 ; i < n; i++){
```

```
        cout<<arr[i]<<" ";
```

```
    }
```

```
    cout<<endl;
```

```
}
```

```
int main(){
```

```
    int n;
```

```
    cin>>n;
```

```

    int* arr = new int[n];
    for(int i = 0 ; i < n; i++){
        cin>>arr[i];
    }
    reverse(arr,n);
    display(arr,n);
}

```

## Rotate An Array

Easy

1. You are given a number n, representing the size of array a.
2. You are given n numbers, representing elements of array a.
3. You are given a number k.
4. Rotate the array a, k times to the right (for positive values of k), and to the left for negative values of k.

### Constraints

$0 \leq n < 10^4$   
 $-10^9 \leq a[i], k \leq 10^9$

### Format

#### Input

Input is managed for you

#### Output

Output is managed for you

### Example

#### Sample Input

```

5
1
2
3
4
5
3

```

#### Sample Output

```

3 4 5 1 2

```

```

#include <iostream>
using namespace std;
void rotate (int*arr ,int n,int t){
    int k{};

    int *tarr= new int [n];
    for(int i{};i<n;i++){
        tarr[i]=arr[i];
    }

    for(int i{};i<n;i++){
        if(t>=0){

```

```

        arr[(i+t)%n]=tarr[i];
    }else{
        t -= (2*t); //making negative to positive (changing
its sign)
        if(i+t<n){
            arr[i]=tarr[i+t];
        }else{
            int d{(i+t-n)};
            arr[i]=tarr[d];
        }
        t -= (2*t); //making positive to negative (changing
its sign)
    }
}

void display(int*arr, int n){
    for(int i{}; i<n; i++){
        cout<<arr[i]<<" ";
    }
    // cout<<endl;
}
int main(){
    int n{};
    //cout<<"Enter the size of the array: ";
    cin>>n;

    int* arr= new int [n];
    for (int i{}; i<n; i++){
        //cout<<"Enter "<<i+1<<" Number: ";
        cin>>arr[i];
    }
    //cout<<"Enter how many times you want to rotate the
array: ";
    int t{};
    cin>>t;
    t=t%n;
    // cout<<t<<endl;
    rotate(arr,n,t);
    display(arr,n);
    delete arr;
    return 0;
}

```

## Inverse Of An Array

Easy

1. You are given a number n, representing the size of array a.

2. You are given n numbers, representing elements of array a.
3. You are required to calculate the inverse of array a.

For definition and constraints check this link

<https://www.pepcoding.com/resources/online-java-foundation/getting-started/inverse-of-a-number/ojquestion>

The only difference is the range of values is from 0 to n - 1, instead of 1 to n.

## Constraints

$0 \leq n < 10^7$

For more constraints check this

<https://www.pepcoding.com/resources/online-java-foundation/getting-started/inverse-of-a-number/ojquestion>

The only difference is the range of values is from 0 to n

- 1, instead

of 1 to n

of 1 to n

## Format

### Input

Input is managed for you

### Output

Output is managed for you

## Example

### Sample Input

```
5
4
0
2
3
1
```

### Sample Output

```
1
4
2
3
0
```

```
#include<iostream>
using namespace std;
```

```
int* inverse(int* arr, int n){
    // write your code here
    int *tarr = new int [n];
    for(int i{};i<n;i++){
        tarr[i]=arr[i];
    }
    for(int i{};i<n;i++){
        arr[tarr[i]]=i;
    }
    return arr;
}

void display(int* arr, int n){
    for(int i = 0 ; i < n; i++){
```



```

        cout<<arr[i]<<endl;
    }
    cout<<endl;
}

int main(){
    int n;
    cin>>n;
    int* arr = new int[n];
    for(int i = 0 ; i < n; i++){
        cin>>arr[i];
    }

    int* inv = inverse(arr,n);
    display(inv,n);

}

```

## Subarray Problem

Easy

1. You are given an array of size 'n' and n elements of the same array.
2. You are required to find and print all the subarrays of the given array.
3. Each subarray should be space separated and on a separate lines. Refer to sample input and output.

### Constraints

1 <= n <= 10

0 <= n1, n2

.. n elements <= 10 ^9

### Format

#### Input

A number n

n1

n2

.. n number of elements

#### Output

[Tab separated elements of subarray]

..

All subarrays

### Example

#### Sample Input

3

10

20

30

#### Sample Output

10

10 20

10 20 30

20

```
20    30
30
```

```
#include<iostream>
using namespace std;

void display (int*arr,int n){
    for(int i{};i<n;i++){
        for(int j{i};j<n;j++){
            for(int k{i};k<=j;k++){
                cout<<arr[k]<<"\t";
            }
            cout<<endl;
        }
    }
}

int main(){
    int n;
    cin>>n;
    int* arr = new int[n];
    for(int i = 0 ; i < n; i++){
        cin>>arr[i];
    }

    // write your code here
    display(arr,n);
    return 0;
}
```

## Subsets Of Array

Easy

1. You are given a number n, representing the count of elements.
2. You are given n numbers.
3. You are required to print all subsets of arr. Each subset should be on separate line. For more clarity check out sample input and output.

### Constraints

$1 \leq n \leq 10$

$0 \leq n_1, n_2, \dots, n \text{ elements} \leq 10^3$

### Format

#### Input

A number n

n1

n2

.. n number of elements

#### Output

[Tab separated elements of subset]

..

All subsets

## Example

### Sample Input

```
3
10
20
30
```

### Sample Output

-	-	-
-	-	30
-	20	-
-	20	30
10	-	-
10	-	30
10	20	-
10	20	30

```
#include <iostream>
#include<cmath>
#include<string>
using namespace std;

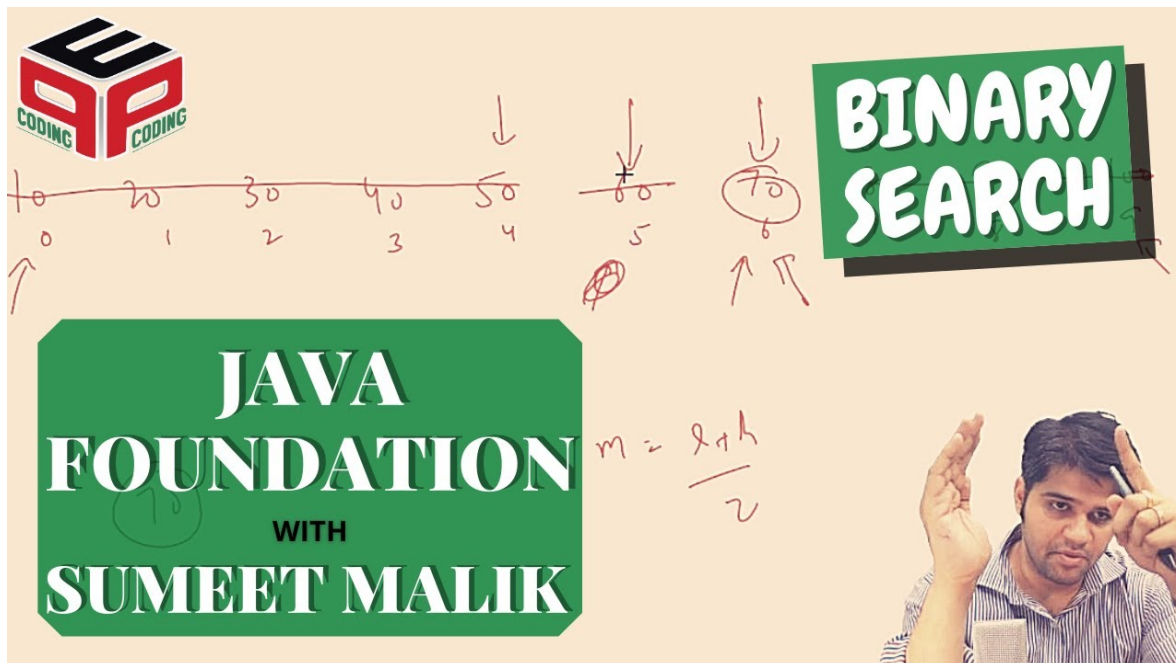
void display (int*arr,int n){
    for (int i{};i<pow(2,n);i++){
        string s1{""};
        int temp{i};
        for(int j{n-1};j>=0;j--){
            int r{temp%2};
            temp=temp/2;
            if(r==0){
                s1="-\t"+s1;
            }else{
                s1=to_string(arr[j])+"\t"+s1;
            }
        }
        cout<<s1<<endl;
    }
}

int main()
{
    int n;
    cin>>n;

    int *arr= new int[n];
    for(int i =0;i<n;i++){
        cin>>arr[i];
    }

    display(arr,n);
    return 0;
}
```

# Binary Search



## Broken Economy

Easy

In a country of novice government, the economic system is changed where only coins are used that too of various denominations. Whenever a foreigner visits this country, they visit a money exchanger to get the currency of the same country. As the foreigner is unaware of the denomination of the country, the money exchange prefers to tell them the denomination which is the nearest maximum and nearest minimum to the denomination mentioned by the foreigner. In case they get the correct guess of the denomination, they are told the same denomination. The denominations are always quoted in ascending order.

Example 1: In a country, 8 given denominations are as follows  
[5, 10, 15, 22, 33, 40, 42, 55]

The foreigner asks for denomination 25.

The money exchange tells them that denominations of 33 and 22 are available.

Example 2:

In a country, 5 given denominations are as follows  
[7, 14, 18, 25, 30]

The foreigner asks for the denomination of 18.

The money exchange tells them a denomination of 18 is available.

You are required to print the values told by the money exchange to the foreigner.

1. You are given a number  $n$ , representing the size of array  $a$ .
2. You are given  $n$  numbers, representing elements of the array  $a$ .
3. You are given another number  $d$ .

4. You are required to find the ceil and floor of d in array a.

### Constraints

$1 \leq n \leq 1000$

$-10^9 \leq n_1, n_2, \dots, n_{\text{elements}} \leq 10^9$

$-10^9 \leq d \leq 10^9$

### Format

#### Input

A number n

n1

n2

.. n number of elements

A number d

#### Output

A number representing ceil

A number representing floor

### Example

#### Sample Input

10

1

5

10

15

22

33

40

42

55

66

34

#### Sample Output

40

33

```
#include<iostream>
using namespace std;
```

```
int main(){
    int n;
    cin>>n;
    int* arr = new int[n];
    for(int i = 0 ; i < n; i++){
        cin>>arr[i];
    }
    int data;
    cin>>data;

    // write your code here
    if(data ==arr[0]){
        cout<<arr[0]<<endl<<arr[0]<<endl;
    }else if(data==arr[n-1]){
        cout<<arr[n-1]<<endl<<arr[n-1]<<endl;
    }else{
        int l{0};
```

```

int h{n-1};
int m{};
while ((h-l)>1){
    m=(l+h)/2;
    if(data<arr[m]){
        h=m;
    }else if(data>arr[m]){
        l=m;
    }else{
        cout<<arr[m]<<endl<<arr[m]<<endl;
        return 0;
    }
}

cout<<arr[h]<<endl<<arr[l]<<endl;

}
return 0;
}

```

## First Index And Last Index

Easy

1. You are given a number n, representing the size of array a.
2. You are given n numbers, representing elements of array a.

Assumption - Array is sorted. Array may have duplicate values.

### Constraints

1 <= n <= 1000

1 <= n1, n2, .. n elements <= 100

1 <= d <= 100

### Format

#### Input

A number n

n1

n2

.. n number of elements

A number d

#### Output

A number representing first index

A number representing last index

### Example

#### Sample Input

15

1

5

10

15

22

33

33  
33  
33  
33  
40  
42  
55  
66  
77  
33

### Sample Output

5  
9

```
#include<iostream>
using namespace std;

int main(){
    int n;
    cin>>n;
    int* arr = new int[n];
    for(int i = 0 ; i < n; i++){
        cin>>arr[i];
    }
    int data;
    cin>>data;

    // write your code here

    int first{};
    int last{};

    int l{0};
    int h{n-1};
    int m{1};

    if (data==arr[l]){
        first=l;
        last =l;
        while((arr[last]==arr[last+1])&&((last+1)<=(n-1))){
            last++;
        }
        cout<<first<<endl<<last<<endl;
        return 0;
    }else if(data==arr[h]){
        first=n-1;
        last =n-1;
        while((arr[first]==arr[first-1])&&(first-1>=0)){
            first--;
        }
        cout<<first<<endl<<last<<endl;
        return 0;
    }else{
```

```

while ((h-l>1)){
    m=(l+h)/2;
    if(data<arr[m]){
        h=m;
    }else if(data>arr[m]){
        l=m;
    }else{
        first =m;
        last=m;
        while((arr[first]==arr[first-1])&&(first-1>=0)){
            first--;
        }
        while((arr[last]==arr[last+1])&&((last+1)<=(n-1)))
        {
            last++;
        }
        cout<<first<<endl<<last<<endl;
        return 0;
    }
}

cout<<-1<<endl<<-1<<endl;
// 10 20 20 20 20 40 50 70 80 90

return 0;
}

```