

Arrays

PRATEEK JAIN

Youtuber at - PRATEEK JAIN Academy
@iNeuron

1) How to create an Array?

Data Type variable name [size];

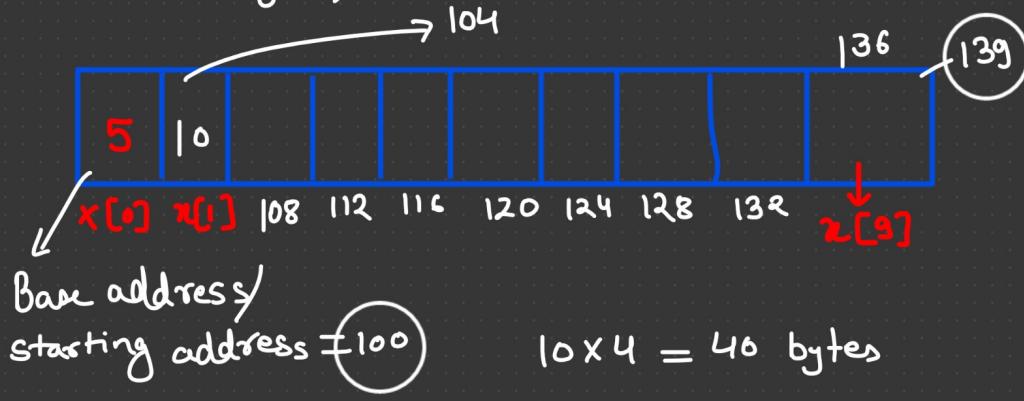
Ex:- int x[10];

x[0] = 5;

x[1] = 10;

⋮

int = 4 byte



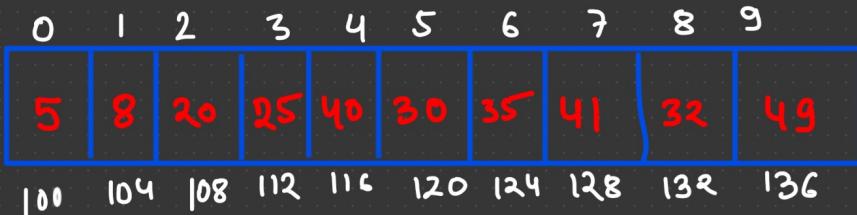
Array is collection of similar type of Elements, stored in contiguous memory allocation.

int x[10];

↓
Declaration

X = Base address = 100

x[0] = Value at 100. = 5



$$x+1 = 104$$

$$x[1] = \text{value at } 104 = 8$$

$$x+6 = 124$$

$$x[6] = \text{value at } 124 = 35$$

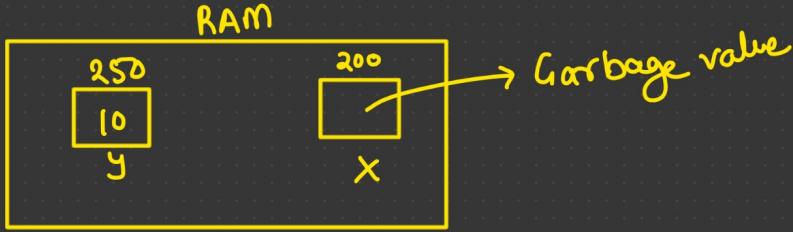
$$x[0] = x + 0 \times 4 = 100 + 0 = 100$$

$$x[2] = 100 + 2 \times 4 = 108$$

$$x[6] = 100 + 6 \times 4 = 124$$

Address = Base Address + (index \times size)

Initialization of an Array:-



int x;
 x=50; → Declaration
 → Assignment
 → int y = 10; → Initialization

int a[5];
 ↓
Declare

1	2	3	4	5
0	1	2	3	4

int a[5] = {1, 2, 3, 4, 5}; // Initialization

int a[5] = {1, 2, 3}; // 1, 2, 3, 0, 0

int a[] = {1, 2, 3, 4, 5}; // size = 5.

2) Accessing Array Elements:-

```

for(i=0 ; i<n ; i++)
{
  printf("%d", a[i]);
}
  
```

i = 0 , a[0] = 5
 i = 1 , a[1] = 10
 i = 2 , a[2] = 15
 i = 3 , a[3] = 25
 i = 4 , a[4] = 35

5	10	15	25	35
0	1	2	3	4

Q:- WAP to find maximum element from the array.



Write a program

20	18	10	15	30	25	35	28	32	40
0	1	2	3	4	5	6	7	8	9

max
40

```
max = a[0];  
for (i= 1 ; i< 10 ; i++)  
{  
    if [ max < a[i] ]  
        max = a[i];  
}  
printf(" Maximum is %d ", max);
```

Q:- Find the sum of all the elements in the given array.

1	4	5	3	4	7	6
0	1	2	3	4	5	6

Sum
x

$$1+4=5$$

$$5+5=10$$

$$10+3=13$$

$$13+4=17$$

Sum=0;

```
for(i=0 ; i< 7 ; i++)  
{
```

```
    sum = sum + a[i];
```

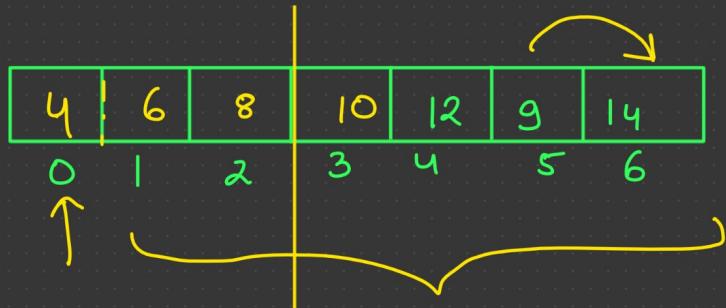
```
}
```

Q:- WAP to sort the given array in increasing order

$$\{ 2, 6, 4, 9, 8 \} \longrightarrow \{ 2, 4, 6, 8, 9 \}$$

$a[0]$

$\text{swap}(j_0, 6);$ | $\text{swap}(10, 6);$
 $\text{swap}(6, 4);$ | $\text{swap}(10, 8);$



$i=0$

$[1, 2, 3, 4, 5, 6]$

$i=1$

$[2, 3, 4, 5, 6]$

$i=2$

$[3, 4, 5, 6]$

$\text{for } (i=0; i < 6; i++)$

{

$\text{for } (j=i+1; j < 7; j++)$

{

$\text{if } (a[i] > a[j])$

$\text{swap}(a[i], a[j]);$

}

}

2D Array :-

	0	1	2	3	Columns
Row → 0	00	01	02	03	
→ 1	10	11	12	13	
→ 2	20	21	22	23	
→ 3	30	31	32	33	

Y-axis

(4,5)

(1,2)

X-axis

Matrix = $4 \times 4 = 16$ cells
 ↓ ↓
 Row Column

int a[4][4];

↓ ↓
 Row Column
 (0-3) (0-3)

int $a[5][4]$;

	0	1	2	3
0	0	1	2	3
1	1	2	3	4
2	2	3	4	5
3	3	4	5	6
4	4	5	6	7

$\uparrow i$

$\uparrow j$

```

for (i = 0 ; i <= 4 ; i++)
{
    for (j = 0 ; j <= 3 ; j++)
        a[i][j] = i + j;
}

```

Memory Representation of 2D Array -

int $a[4][4]$;

	0	1	2	3
0	1	2	3	4
1	5	6	7	8
2	9	10	11	12
3	13	14	15	16

$\rightarrow 16 \text{ byte}$

$\rightarrow 16 \text{ byte}$

$\rightarrow 16 \text{ byte}$

$\rightarrow 16 \text{ byte}$

Row 0 Row 1 Row 2 Row 3

100 104 108 112 116 120 124 128 132 136 140 144

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑ ↑

0 1 2 3 0 1 2 3 0 1 2 3 0 1 2 3

4x4

$$a[1][2] = 100 + [(4 \times 4) \times 1] + (4 \times 2) = 124$$

$$a[2][3] = 100 + [(4 \times 4) \times 2] + (4 \times 3) = 144$$

(no. of column) sizeof(int)

`int a[R][C];` $a[i][j]$

$$\text{Address} = \text{Base Address} + [C \times (\text{Size of Data Type}) \times i] + (\text{Size of DT}) \times j$$

Q:- Addition of two 3×3 matrix.

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 2 & 1 & -1 \\ 4 & 5 & 2 \end{bmatrix}_{3 \times 3}, \quad B = \begin{bmatrix} 3 & 1 & 4 \\ 2 & 0 & 6 \\ -1 & 2 & 4 \end{bmatrix}_{3 \times 3}$$

$$A+B = \begin{bmatrix} 4 & 3 & 7 \\ 4 & 1 & 5 \\ 3 & 7 & 6 \end{bmatrix}$$

$$i=0 \quad j=0, 1, 2$$

$$[0][0] \quad [0][1] \quad [0][2]$$

$$\begin{aligned} \text{int } a[3][3] &= \{1, 2, 3, 2, 1, -1, 4, 5, 2\}; \\ \text{int } b[3][3] &= \{3, 1, 4, 2, 0, 6, -1, 2, 4\}; \\ \text{int } c[3][3] &= \end{aligned}$$

`for (i=0 ; i<3 ; i++)`

{ `for (j=0 ; j<3 ; j++)`

$$\{ \quad c[i][j] = a[i][j] + b[i][j];$$

}

2

Q:- Matrix multiplication 2×2 .

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \xrightarrow{\text{R}_1} \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix} \xrightarrow{\text{R}_2} \begin{bmatrix} 1 & 2 \\ 0 & 4 \end{bmatrix}, \quad B = \begin{bmatrix} C_1 & C_2 \\ 1 & -1 \\ 2 & 3 \end{bmatrix} \downarrow \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \times \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$A \times B = \begin{bmatrix} 1 \times 1 + 2 \times 2 & 1 \times (-1) + 2 \times 3 \\ 3 \times 1 + 4 \times 2 & 3 \times (-1) + 4 \times 3 \end{bmatrix}$$

$$\text{Res} \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 1 & 1 \end{bmatrix}$$

$$R=0 \times C_1$$

$$R=0 \times C_2$$

$$R=1 \times C_1$$

$$R=1 \times C_2$$

$$\text{for } (i=0; i<2; i++) \rightarrow i=0$$

$$\text{for } (j=0; j<2; j++) \rightarrow j=0$$

$$\text{for } (k=0; k<2; k++)$$

$$i=0, j=0 \rightarrow \text{sum} = \begin{bmatrix} (0,0) & \times & (0,0) \\ (0,0) & \times & (0,1) \end{bmatrix} + \begin{bmatrix} (0,1) & \times & (1,0) \\ (0,1) & \times & (1,1) \end{bmatrix}$$

$$\{ \quad c[i][j] = \text{sum}; \quad \}$$

$$\left| \begin{array}{c} i=1 \\ j=1 \\ K=(0,1) \end{array} \right| \quad \begin{array}{c} i=1 \\ j=0, j=1 \\ K=(0,1) \end{array}$$

$$\text{a}[i][k] * b[k][j] \rightarrow \text{sum}$$

$$\text{sum} = \text{sum} + a[i][k] * b[k][j]$$

$$\begin{matrix}
 & A & & B \\
 \left[\begin{array}{cc} 00 & 01 \\ 10 & 11 \end{array} \right] & & \left[\begin{array}{cc} 00 & 01 \\ 10 & 11 \end{array} \right]
 \end{matrix}$$

$(00) \xrightarrow{i \uparrow j \downarrow} (00) \times (00)$ + $(01) \xrightarrow{i \downarrow j \uparrow} (01) \times (10)$
 $\xrightarrow{\quad K=0 \quad}$ $\xrightarrow{\quad K=1 \quad}$
 $(10) \xrightarrow{i \downarrow j \uparrow} (10) \times (00)$ + $(11) \xrightarrow{i \uparrow j \downarrow} (11) \times (10)$
 $\xrightarrow{\quad K=0 \quad}$ $\xrightarrow{\quad K=1 \quad}$

$$\begin{array}{c}
 \text{Diagram 1: } \\
 \begin{array}{l}
 \text{Left: } (00) \times (01) \quad \text{Right: } (01) \times (11) = (01) \\
 \text{Bottom Left: } (10) \times (01) \quad \text{Bottom Right: } (11) \times (11) = (11)
 \end{array}
 \end{array}$$

```
for(i=0 ; i<2 ; i++)
```

```
for (j=0 ; j<2 ; j++)  
{
```

```

sum = 0;
for(k=0; k<2; k++)
{
    sum = a[i][k] * b[k][j] + sum;
}

```

$$\sum_j c_i[j] = \text{sum}_j$$

2

Q:- Find Transpose of the given matrix?

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 4 & 7 \\ 2 & 5 & 8 \\ 3 & 6 & 9 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix}$$

$$\begin{bmatrix} \underline{00} & \underline{01} & \underline{02} \\ \underline{10} & \underline{11} & \underline{12} \\ \underline{20} & \underline{21} & \underline{22} \end{bmatrix} \rightarrow \begin{bmatrix} 00 & 10 & 20 \\ 01 & 11 & 21 \\ 02 & 12 & 22 \end{bmatrix}$$

for ($i=0$; $i<3$; $i++$)

{
for ($j=0$; $j<3$; $j++$)

{
 $b[i][j] = a[j][i];$

}

}

$$\begin{bmatrix} 00 & 01 & 02 \\ 10 & 11 & 12 \end{bmatrix}$$

2×3

$$\begin{bmatrix} 00 & 10 \\ 01 & 11 \\ 02 & 12 \end{bmatrix}$$

3×2

Q:- Sum of right diagonal / left diagonal?

$$\begin{bmatrix} 00 & 01 & 02 \\ 10 & 11 & 12 \\ 20 & 21 & 22 \end{bmatrix}$$

Left

$$\begin{bmatrix} 00 & 01 & 02 \\ 10 & 11 & 12 \\ 20 & 21 & 22 \end{bmatrix}$$

Right

for ($i=0$; $i<3$; $i++$)
sum = $a[i][i] + \text{sum};$

{

if ($i+j==2$)
sum = $a[i][j] + \text{sum};$

for ($i=0$, $j=2$; $i<3$; $i++$, $j--$)
sum = $a[i][j] + \text{sum};$

3D Array :-

DaryMilk = 0

DaryMilk = 1

No. of DM
Row of DM
Col of DM
 $DM[0][1][2]$

00	01	02	03
10	11	12	13
20	21	22	23

00	01	02	03
10	11	12	13
20	21	22	23

3×4

$0 \rightarrow (12)$

3×4

$1 \rightarrow (12)$

$DM[1][1][2]$

1	2	3	4
---	---	---	---

$1 DM \rightarrow 12$ pieces (each row 4 piece)

$$\frac{12}{4} = 3$$

Box $\rightarrow 4 DM$

Size of Box = ?

`int a[3][4][2];`

`Box[4][3][4];`

$4 \times 3 \times 4$

no. of DM

no. of Row

no. of Col

0	0
1	
2	
3	

0	0	1	1
1			
2			
3			

0	2
1	
2	
3	

No. of element

$$= 3 \times 4 \times 2$$

$$= 24$$

$$a[1][2][1] = 3;$$

int $a[2][2][3] = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\};$

int $a[2][2][3] = \{ \{ \{1, 2, 3\}, \{4, 5, 6\} \}, \{ \{7, 8, 9\}, \{10, 11, 12\} \} \};$

1	2	3
4	5	6

7	8	9
10	11	12

0

1

int $a[] [2] [3] = \{ \underbrace{1, 2, 3}_{0}, \underbrace{4, 5, 6}_{0}, \underbrace{7, 8, 9}_{1}, \underbrace{10, 11, 12}_{1} \};$

1 matrix 2 matrix

int $a[] [3] [2] = \{ \underbrace{1, 2, 3, 4}_{0}, \underbrace{5, 6}_{0}, \underbrace{7, 8, 9}_{1}, \underbrace{10, 11}_{1}, \underbrace{12, 13, 14}_{1}, \underbrace{15, 16, 17}_{2}, 0 \}$

③

①

②

③

for ($i=0; i<3; i++$)

{
 for ($j=0; j<3; j++$)

{
 for ($k=0; k<2; k++$)

{
 $a[i][j][k] = i + j + k;$

{

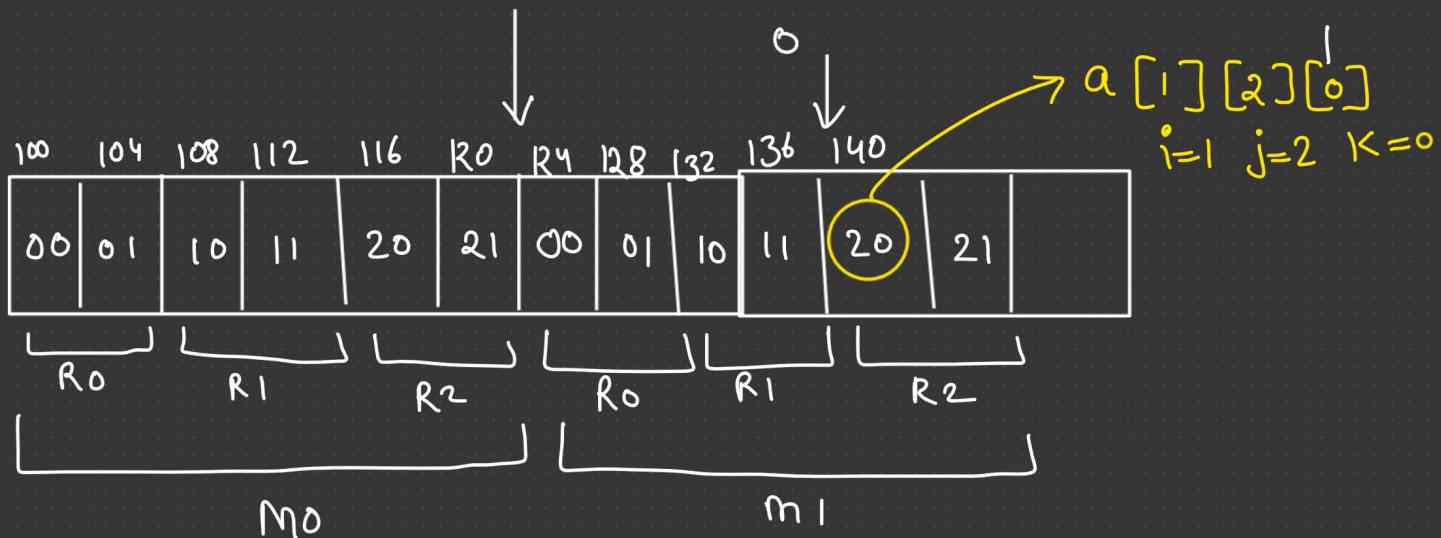
}

int a[2][3][2];

N R C

100	
00	01
10	11
20	21

00	01
10	11
20	21



$$a[1][2][0] = 100 + [1 \times (3) \times (2) \times 4] + [2 \times (2) \times 4] + [0 \times 4]$$

$$= 100 + 24 + 16 + 0 = 140$$

Address = Base Address + $[i \times (R \times C) \times \text{size}] + [j \times (C \times \text{size})] + [k \times \text{size}]$

Q:- WAP to check the given matrix / Det. is sparse or not?

$$\begin{bmatrix} - & 0 & - \\ - & - & 0 \\ - & - & 0 \end{bmatrix}$$

$3 \times 3 = 9 \text{ elements}$

$$\frac{9}{2} = 4 \leq \text{zero's} \Rightarrow \text{sparse}$$

$$\Rightarrow \text{dense}$$

```

for( i=0; i<n ; i++)
{
    for ( j=0; j<m; j++)
    {
        if( a[i][j] == 0)
            count++;
    }
}
if( count > (m*n)/2)
    printf(" sparse");
else printf(" dense");

```

```

int a[n][m];
int count=0;

```

Q. WAP to print lower & upper triangular matrix:-

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} 1 & 2 & 3 \\ 0 & 5 & 6 \\ 0 & 0 & 9 \end{bmatrix}$$

$$\Downarrow \begin{bmatrix} 1 & 0 & 0 \\ 4 & 5 & 0 \\ 7 & 8 & 9 \end{bmatrix}$$

$$\begin{bmatrix} 00 & 01 & 02 \\ 10 & 11 & 12 \\ 20 & 21 & 22 \end{bmatrix}$$

$\nearrow i < c$
 $\searrow i > c$
 $i = c$

```

for( i= 0; i< n ; i++)
{
    for ( j=0; j< m ; j++)
    {
        if( i <= j)
            printf("%d", a[i][j]);
        else
            print("0");
    }
}

```

WAP to find 2nd largest element in the given array?

1	4	10	8	7	9	5	3	2	11
$\rightarrow \rightarrow \rightarrow \rightarrow \rightarrow$									

$$\max_1 = 10$$

$$\max_2 = 8$$

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

```
{
```

```
    if (max1 < num)
```

```
{ max2 = max1;
```

```
    max1 = num;
```

```
}
```

```
else if (max2 < num)
```

```
    max2 = num;
```

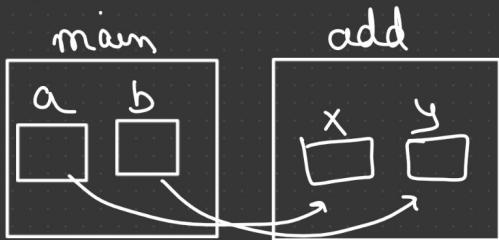
```
}
```

```
printf("Second Largest no. is %d", max2);
```

Array & functions

```

main()
{
    f1(); // TNRN
    f2(n); // TSRN
    int y = f3(); // TNRS
    int z = f4(y); // TSRS
}
  
```



```

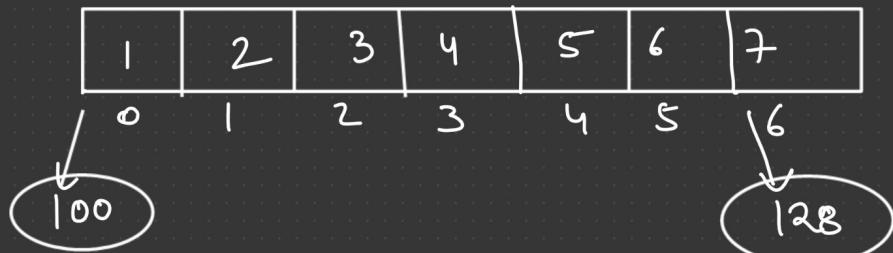
main()
{
    int a=5, b=10;
    add(a,b);
}

void add(int x,int y)
{ }
  
```

≡

```

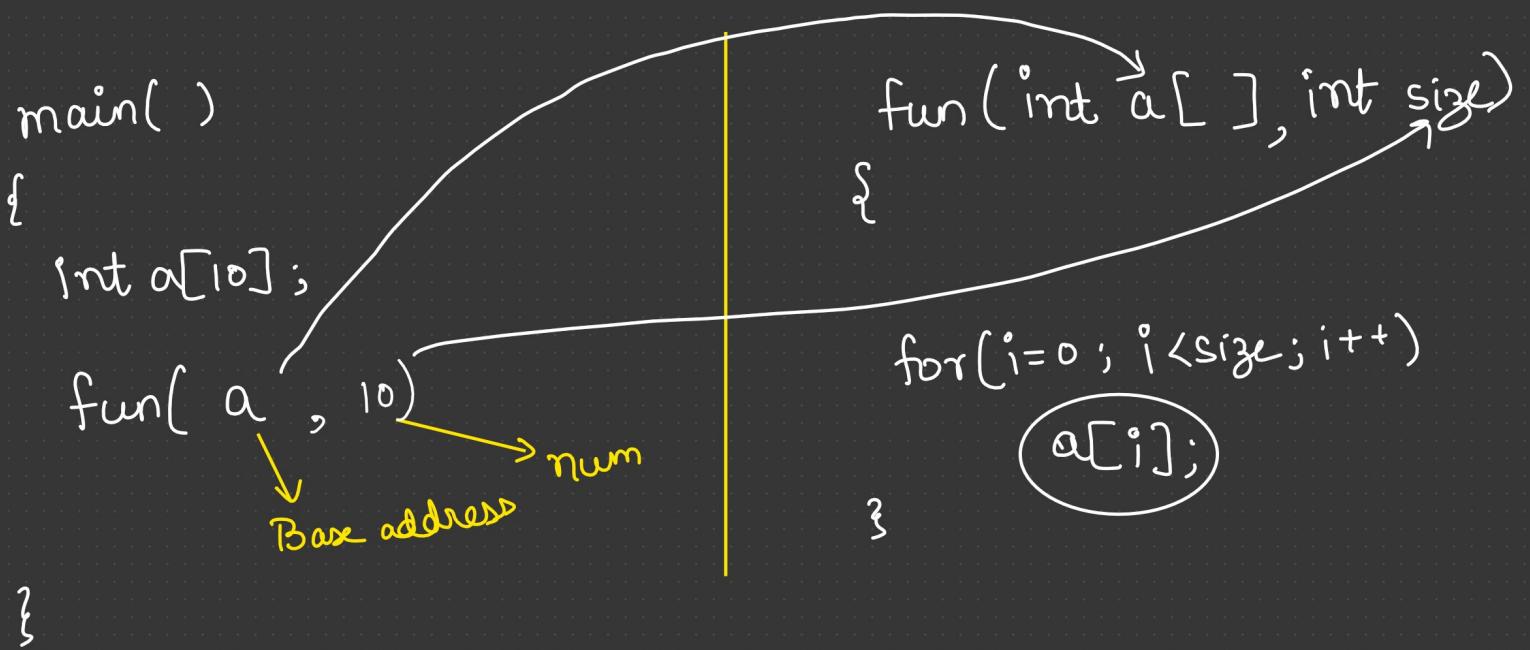
int a[7];
a[0] = 1;
a[2] = 3;
  
```



$a = \&a[0] = \boxed{\text{base address} + \text{Data type} + \text{no. of element}}$

(int) 4-byte = $\boxed{100, 101, 102, 103}$

$$100 + 7 \times 4 = 128$$



WAP to find largest number in a given array using fm.

```

main( )
{
    int a[5]={2,4,3,5,1};
    findMax(a);
}

```

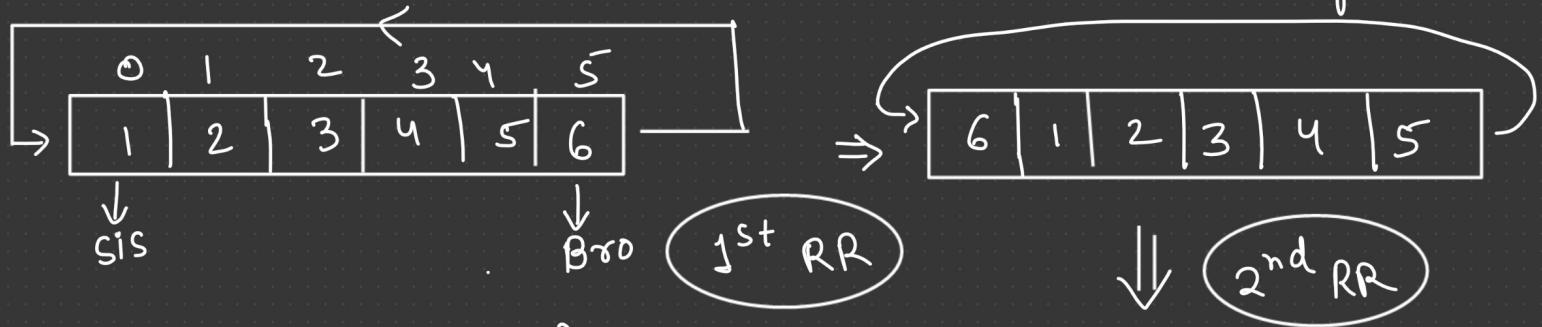
```

Void findMax( int x[ ] )
{
    int i, max = 0;
    for( i=0 ; i<5 ; i++ )
    {
        if( max < x[ i ] )
            max=x[ i ];
    }
    printf("Max %.d", max);
}

```

WAP to print all the elements of a given array using function.

WAP to rotate right by "d" distance in a given array of length "n".



for ($j = 1; j \leq d; j++$)

{ int $x = a[\text{size} - 1];$



for ($i = \text{size} - 1; i > 0; i--$)

$a[i] = a[i - 1];$

}

$a[0] = x;$

}

WAP to find frequency of all the no. in a given array.

1	4	5	3	4	1	2	5	3	4	5
---	---	---	---	---	---	---	---	---	---	---

No.	frequency	Count = X 2
1	→ 2 times	
2	→ 1	
3	→ 2	
4	→ 3	
5	→ 3	

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

```

for( i=0 ; i<n; i++)
{
    for( j=0 ; j<n; j++)
    {
        → n×n times
    }
}

```

$$n^2 > 2n$$

```

for( i=0 ; i<n ; i++)
{
    → n
    for( j=0 ; j<n ; j++)
    {
        → n
    }
}

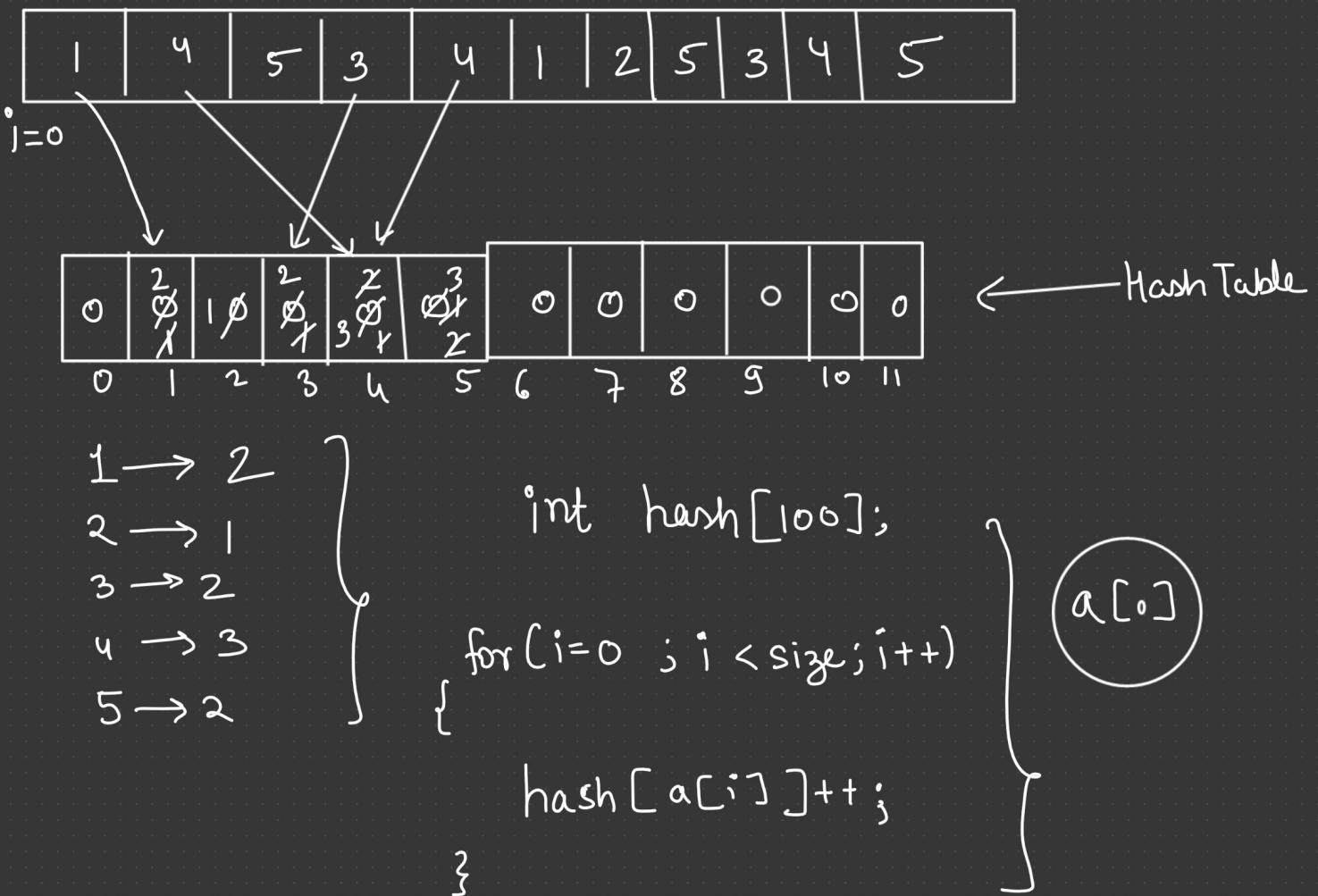
```

↑

This code is same as one loop code.

2n

Hashing Technique



Q:- WAP to find duplicate number/numbers?

```
main()
{
    int a[5][10] = { . . . . };
    fun( a , 5 , 10 );
}

fun( int a[ ][10] , int i , int j )
{
}
```

String in C

Sequence of characters is called string.

'P', 'R', 'A', 'T', 'E', 'E', 'K' \Rightarrow characters

"prateek" \Rightarrow string.

Declaration of String :-

char name[] = { 'P', 'R', 'A', 'T', 'E', 'E', 'K', '\0' };

predefined fn \rightarrow string \rightarrow printf

'P' \rightarrow character
"P" \rightarrow string

printf("%s", name);

%d, %f, %c

2) `char name[6] = {'A', 'B', 'C', 'D', 'E', '\0'};`

3) `Char name[] = "Prateek";` ↗

Input:- 1) `scanf("%s", a);`

2) `gets(a);`

3) `fgets(a, max, stdin);`

↳ $a[\text{strlen}(a)-1] = '\0';$ → v. Imp

WAP to count number of vowels in a given string?

Ex:-

"Prateek" → 3 (a, e, e)

WAP to convert uppercase to lowercase & vice versa.

PrATeeK → pRatEEK

$$'a' = 97$$

$$- 'A' = 65$$

$$\underline{32}$$

$$'a' - 32 = 'A'$$

$$'A' + 32 = 'a'$$

$$'z' = 122$$

$$'Z' = 90$$

WAP to check the given string is palindrome or not.

"aabbaa" ✓

i=0 j = s.length - 1

→ ← j--

i++

"Nayan" ✓

→ ←

WAP to find frequency of all character in the given string.

Ex:- char a[] = "abab~~cac~~";

a = 3 int hash [200] = {0};
b = 2
c = 2

```
for(i=0; a[i] != '\0'; i++)
{
    hash[a[i]]++;
}

for(i=0; i < 123; i++)
{
    if(hash[i] > 0)
        printf("%c = %d", i, hash[i]);
}
```

Multiple Strings

char a[] = "Prateek";

P	R	A	T	E	E	K	'\0'	
a[0]	a[1]	a[2]	a[3]	a[4]	a[5]	a[6]	a[7]	a[8]

a[0] = 'P'

a[5] = 'E'

a[3] = 'T'

char a[5][10] = { "Prateek", "Shubham", "Milan", "Arnab", "Rashant" };

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

String No. Char no. of
 that string.
(R) (C)

a[0] = "Prateek";

a[1] = "Shubham";

a[2][3] = 'a'

a[3][2] = 'n'

a[0] → 0

a[1] → 1

a[2] → 2

a[3] → 3

a[4] → 4

0	1	2	3	4	5	6	7	8	9
P	r	a	t	e	e	K			
S	h	u	b	h	a	m			
M	i		l	a	n				
A	r	n	a	b					
P	r	a	s	h	a	n	t		

for (i = 0 ; i < 5 ; i++)

printf("%s\n", a[i]);

for (i = 0 ; i < 5 ; i++)

{ for (j = 0 ; a[i][j] != '\0' ; j++)

{ if (a[i][j] == 'a')

count++;

}

WAP to sort given names in dictionary order.

- 1) strcmp("Prateek", "prashant"); > 0 (swap)
- 2) strcmp("Prateek", "Prateek"); = 0 (equal)
- 3) strcmp("Prashant", "Prateek"); < 0 (no need to swap)

Char name[5][20]; | n=5

Char temp[20];

for(i=0; i < n-1; i++)

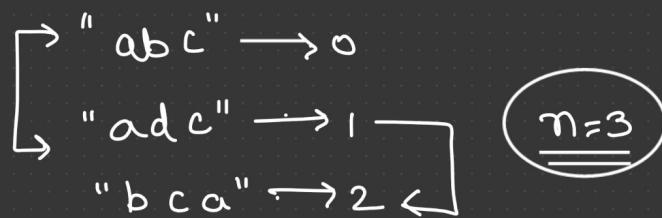
{ for(j=i+1; j < n; j++)

{ if(strcmp(name[i], name[j]) > 0)

{ strcpy(temp, name[i]);
strcpy(name[i], name[j]);
strcpy(name[j], temp);

{

}



	0	1	2	3	4	5	6	7	8	9
a[0] → 0	A	r	n	a	b					
a[1] → 1	S	h	u	b	h	a	m			
a[2] → 2	P	r	a	t	e	e	k			
a[3] → 3	M	i	l	a	n					
a[4] → 4	P	r	a	s	h	a	n	t		

WAP to Search a name in the given List of Strings.

```
char a[5][10] = { "Pratcek", "Shubham", "Milan", "Arnab", "Prashant" };
```

```
char name[] = "milan";
```

```
for (i=0 ; i< 5; i++)
{
    if (strcmp(name, a[i]) == 0)
        printf("String found");
}
```