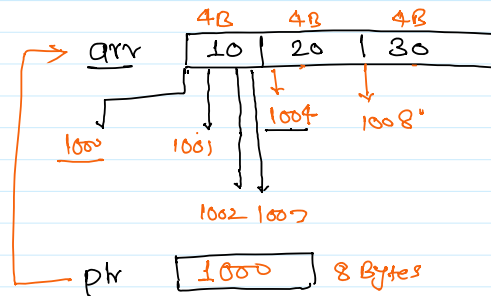


Pointer Arithmetic

```
int arr[3] = {10, 20, 30}; // 12 byte
int *ptr = arr; // array names refers to its address
// int *ptr = &arr[0]
```



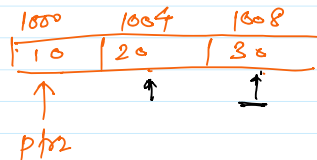
// Traverse arr using pointer.

```
① printf("%d", *ptr); // 10 ptr [1000]
    printf("%d", *(ptr+1)); // 20 ptr [1004] -
    printf("%d", *(ptr+2)); // 30 ptr [1008] -
```

} ptr is still pointing to first element

Q: How to print nth element of Array

```
int *ptr2 = arr;
printf("%d", *(ptr2+2)); // 30
```



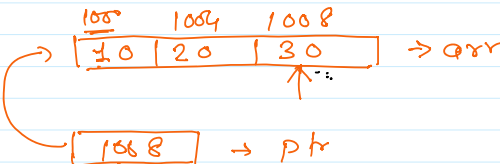
② // Traverse arr using pointer

```
int arr[3] = {10, 20, 30};
int *ptr = arr;
```

```
printf("%d", *ptr); // 10
```

```
printf("%d", *(++ptr)); // 20
```

```
printf("%d", *++ptr); // 30
```



Pre-increment
[1004]
[1008]

③ int arr[3] = {10, 20, 30};

```
int *ptr = arr;
```

```
printf("%d", *ptr); // 10
```

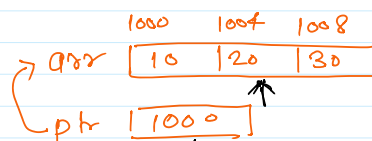
```
printf("%d", *(ptr++)); // 10
```

```
// ptr is post increment
```

```
→ printf("%d", *ptr); // 20
```

```
→ printf("%d", *(ptr++)); // 20
```

```
→ printf("%d", *ptr); // 30
```



P [1004]

ptr [1004]

ptr [1008]

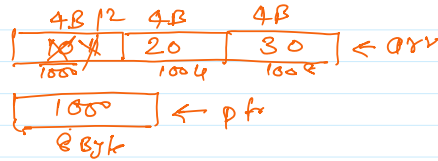
④ `int arr[3] = {10, 20, 30};`

`int *ptr = arr;`

`printf("%d", (*ptr)++);` 1000

`printf("%d", (*ptr)++);`

`printf("%d", (*ptr));`



`ptr` 1000

`ptr` 1000

⑤ `int arr[3] = {10, 20, 30};`

`int *ptr = arr;`

`*ptr = 100;`

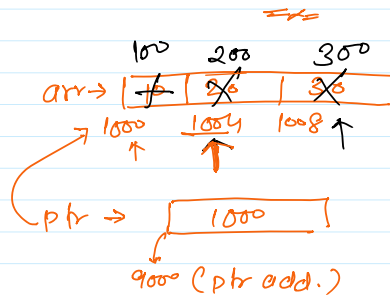
`printf("%d", *ptr);` → 100

`*++ptr = 200;`

`printf("%d", *ptr);` → 200

`*++ptr = 300;`

`printf("%d", *ptr);` → 300



`ptr++` → Will not jump to next element

`ptr++` → Will jump to next element

`++ptr;` → `ptr = ptr + 1` ✓

`ptr++` = address of next

Q

`int a = 10;`

~~`a = 5;`~~

`a++` → `a = a + 1 = 20`

`a` 10

Q

`printf("%d", a++)` → 10