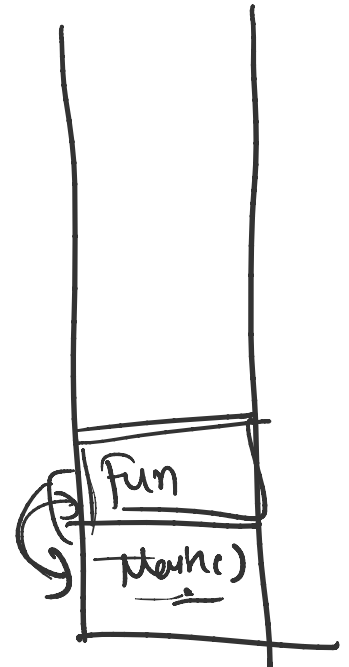


Functions:-

```

int main() {
    ✓
    ✓
    ✓
    → func();
    → ...
}
    
```



return-type      function-name (arg1, arg2, arg3...)

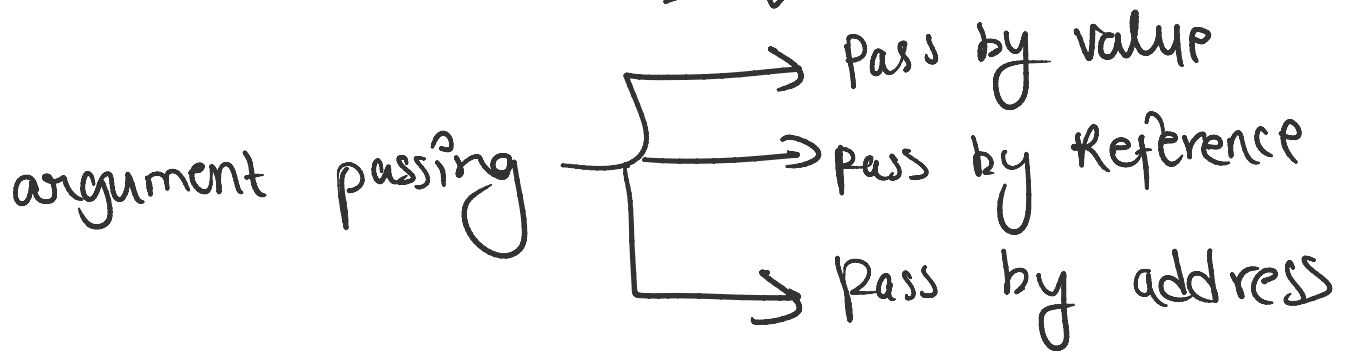
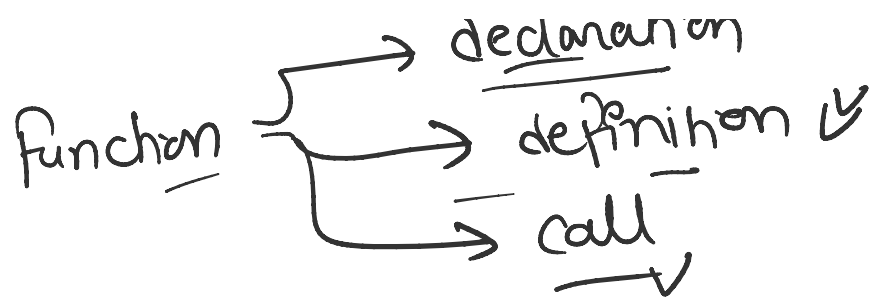
```

_____
/
/
/
_____
    
```

```

}
int main() {
    _____
    function-name(a);
}
    
```

→ declaration



	Call	Define
value	fun(a,b);	— fun(int a, int b) { ... }
reference	f(a,b);	— fun(int &a, int &b) { ... }
address	f(&a, &b);	— fun(int* a, int* b) { ... }

fun(      )

int main()

a, b

fun(a,b);

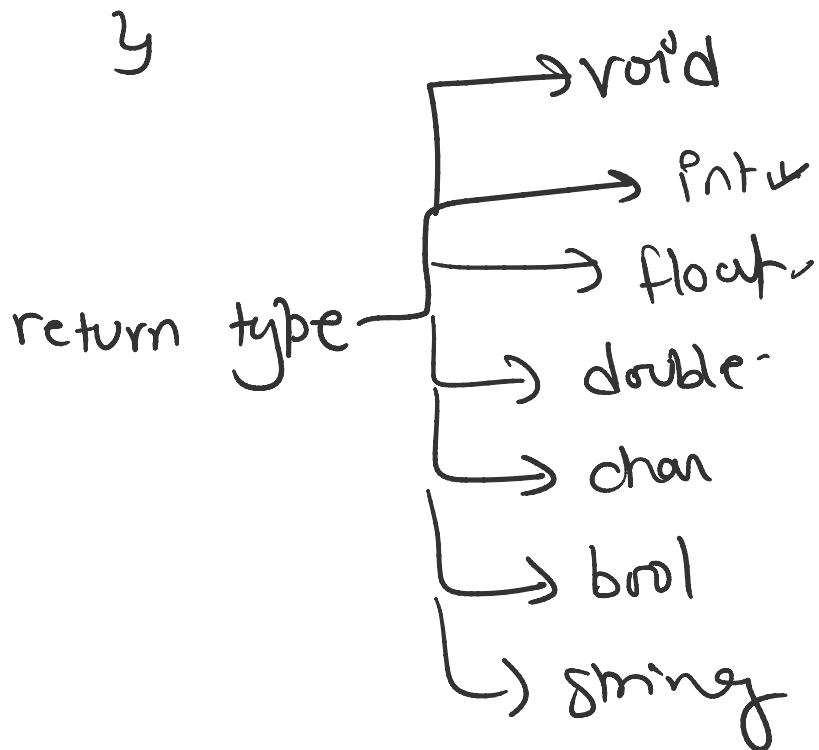
a-1, b+1

... int &b) {

```

    a--j
    b++j
}
void helper(int &a, int &b) {
    a--j
    b++j
}
int main() {
    int a, b;
    cin >> a >> b;
    helper(a, b);
    cout << a << " " << b << endl;
    return 0;
}

```



Array :-

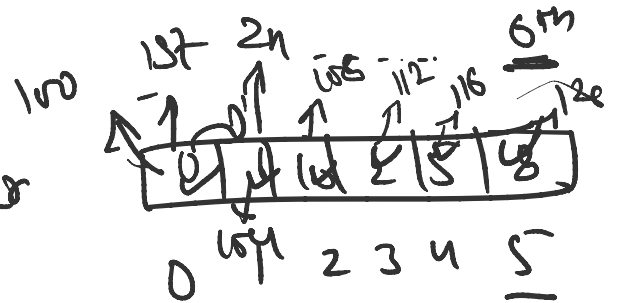
```

int a, b, c, d, e, f;
a = 0;

```

$a = 0;$   
 $b = 6;$   
 $c = 10;$   
 $d = 2;$   
 $e = 5;$

array  $\rightarrow$  container



declaration of array:-

$\text{int arr}[10];$

$\text{int arr}[] = \{1, 6, 8, 2, 3, 7, 8\};$

$\text{int arr}[5] = \{1, 2, 3, 4, 5\};$

$\text{int n};$   
 $c \text{rn} > n$

$\text{arr}[3]$



$\text{int main}() \{$

$\text{int n};$

```
int n;
```

```
cin >> n;
```

```
int arr[n];
```

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

```
    cin >> arr[i];
```

```
}
```

```
print(arr, n);
```

```
return 0;
```