

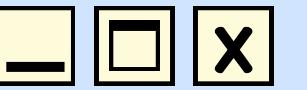
Embedded Systems

Task 3

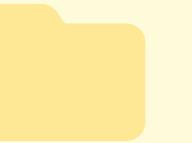
by Mohammed Elahmady



Task 2 Objects



- ✓ Task 2 Objects
- > 1d Arrays
- > 2d Arrays



1d Arrays

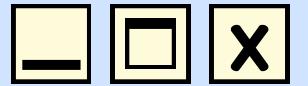


2d Arrays





Arrays



- an array is essentially a collection of elements.
- the data types for all elements must be the same.
- and store the data in a contiguous memory location.
- Each element of the array is individually referenced by using an index “Start from 0”.
- it means if you want to get the first element of the array then the index must = 0.
- the name of array point to the address of the first element[0].

Array Syntax :

```
//Declaration  
Data_Type Array_name[index];  
  
//Definition  
Data_Type Array_name[] = {1,2,3,4,5,6,7,8,9,10};
```

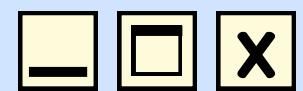
We can initialize with this method in C99 Standard :

```
unsigned int arr[20] = {1,2,3,[15]=22,4,5,[13]=2};
```





Arrays



- The name of the array is the address of the first element.

```
printf("add = %#x\n",arr);
printf("add = %#x\n",&arr[0]);|
```

- The Address of an array element =

```
&arr[n] => &arr[0] + (size of data type * n)
```

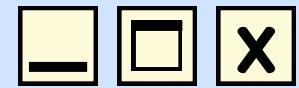
- index of array is a static configuration I can't change it at runtime.
- Size of array = size of data type * number of elements
- the number of elements in array =

```
C ↴
sizeof(arr)/sizeof(arr[0])
(*(&arr+1)-*(&arr))
```



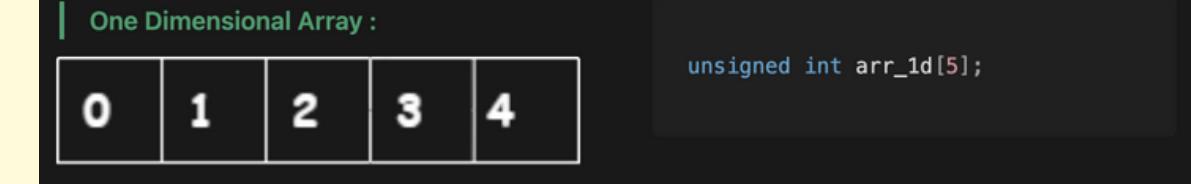


Arrays

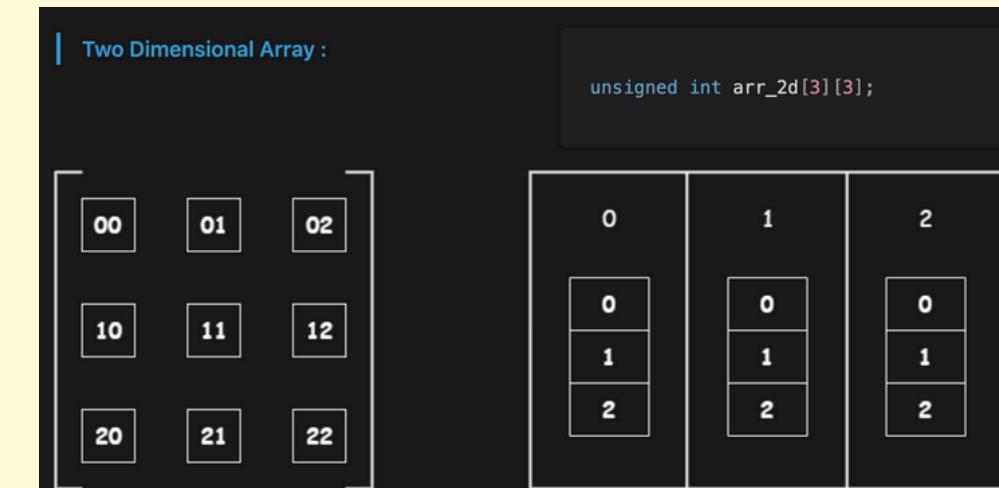


Types of Arrays :

One Dimensional Array :

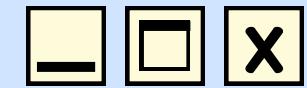


Two Dimensional Array :





String



- Initialize String in C Using Arrays :

To initialize array of String :

```
char arr[n+1] = {"String"};
Equal to
char arr[7] = {'S','t','r','i','n','g'};
```

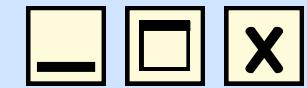
- To input a string into an array, I use a function called `gets`.
- You can use `scanf`, but you must use `fflush(stdin);` afterward to clear the buffer, because `scanf` does not clear the buffer itself.
- You use `fflush(stdin);` to clear any leftover "enter" after using `scanf` for a string, in to clear the buffer.
- array doesn't have any type of boundary check

```
int arr[5] = {2,4,6,8,10};
arr[6] = 17;
printf("arr[6] = %i\n",arr[6]); // => 17
```





String



- Initialize String in C Using Arrays :

To initialize array of String :

```
char arr[n+1] = {"String"};
Equal to
char arr[7] = {'S','t','r','i','n','g'};
```

- To input a string into an array, I use a function called `gets`.
- You can use `scanf`, but you must use `fflush(stdin);` afterward to clear the buffer, because `scanf` does not clear the buffer itself.
- You use `fflush(stdin);` to clear any leftover "enter" after using `scanf` for a string, in to clear the buffer.
- array doesn't have any type of boundary check

```
int arr[5] = {2,4,6,8,10};
arr[6] = 17;
printf("arr[6] = %i\n",arr[6]); // => 17
```



THANK YOU

Head : Tasnem Sabry

Vice : Ahmed Yasser