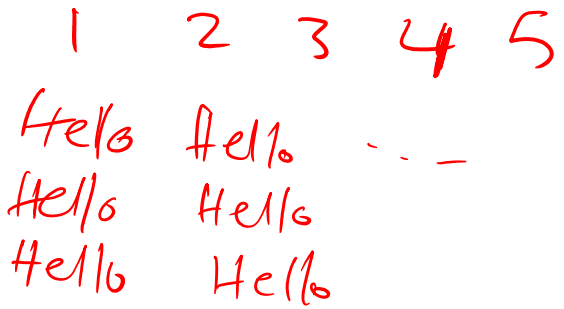
Session 6

Mostafa Akram

More about loops !

- Infinite Loop
- Nested loop ⟶
- If condition + Loop
- Loop + array (later)
- Loop + strings (later) ... etc



Tasks

- Take from user 2 numbers and find the result of AND(&), OR(|) and OXR (^) from the minimum number to the max.

Ex:

Input

2, 10

output

AND OR XOR

2, 3

2, 4

2, 5

2, 6

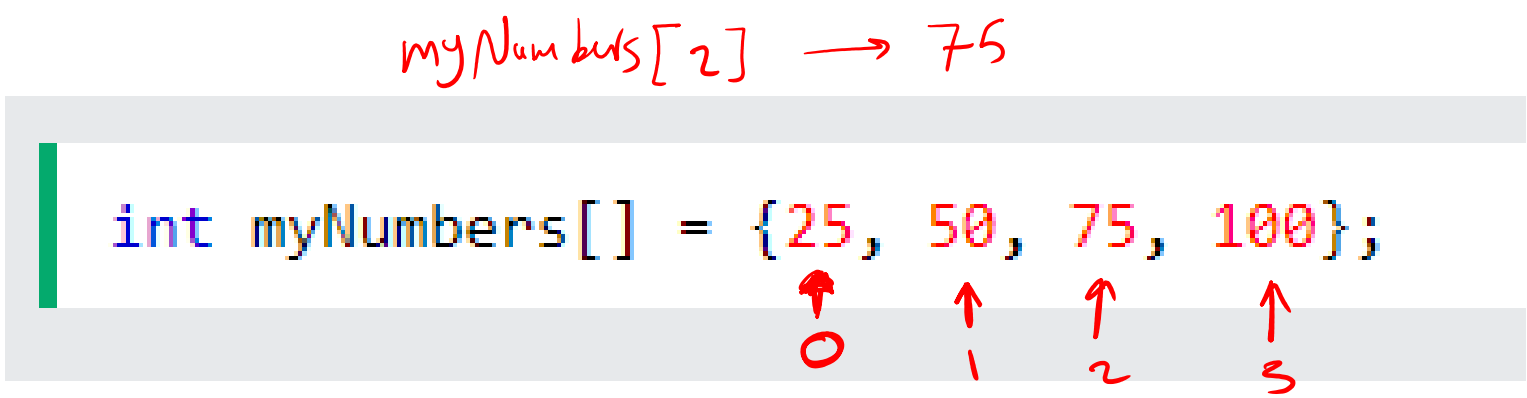
.....etc.

2, 10

Arrays

- Arrays are used to store multiple values in a single variable, instead of declaring separate variables for each value.

- To create an array, define the data type (like int) and specify the name of the array followed by square brackets [].



Access the Elements of an Array

```c
int myNumbers[] = {25, 50, 75, 100};
printf("%d", myNumbers[0]);

// Outputs 25
```

Change an Array Element

```
myNumbers[0] = 33;
```

Change an Array Element

```
int myNumbers[] = {25, 50, 75, 100};
myNumbers[0] = 33;

printf("%d", myNumbers[0]);

// Now outputs 33 instead of 25
```

Loop Through an Array

```c
int myNumbers[] = {25, 50, 75, 100};
int i;

for (i = 0; i < 4; i++) {
  printf("%d\n", myNumbers[i]);
}
```

Set Array Size

```
// Declare an array of four integers:
int myNumbers[4];

// Add elements
myNumbers[0] = 25;
myNumbers[1] = 50;
myNumbers[2] = 75;
myNumbers[3] = 100;
```

Get Array Size or Length

```c
int myNumbers[] = {10, 25, 50, 75, 100};
printf("%lu", sizeof(myNumbers)); // Prints 20
```

Why did the result show 20 instead of 5, when the array contains 5 elements?

You learned from the Data Types chapter that an int type is usually 4 bytes, so from the example above, 4 x 5 (4 bytes x 5 elements) = 20 bytes so, to know size of array:

```
int myNumbers[] = {10, 25, 50, 75, 100};
int length = sizeof(myNumbers) / sizeof(myNumbers[0]);

printf("%d", length);  // Prints 5
```

Making Better Loops

```c
int myNumbers[] = {25, 50, 75, 100};
int i;

for (i = 0; i < 4; i++) {
  printf("%d\n", myNumbers[i]);
}
```

```c
int myNumbers[] = {25, 50, 75, 100};
int length = sizeof(myNumbers) / sizeof(myNumbers[0]);
int i;

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

Task

- calculates the average of different ages.
-
```
// An array storing different ages
int ages[] = {20, 22, 18, 35, 48, 26, 87, 70};
```

- Note print 2 decimal numbers after floating point (%.2f)

Links