

Meet - myg-vper-ijj

meet.google.com/myg-vper-ijj?authuser=0

R

Ritu Sibbal is presenting

S

SHYAMAL JAIN
and 39 more

47

1:05 PM

You

Arrays and Strings

RAHUL SHARMA has raised a hand

bxzgsjmls


Raise hand

Turn on captions


Ritu Sibbal
is presenting

R

... Ritu Sibbal




DHRUV KUMAR



SHOBHIT PRAKASH

R

RAHUL SHARMA



MEGHNA

Meet - myg-vper-juj

meet.google.com/myg-vper-juj?authuser=0

Ritu Sibbal is presenting

AMAN SONI and 52 more

1:06 PM

- An *array* is a collection of variables of the same type that are referred to through a common name.
- A specific element in an array is accessed by an index. In C, all arrays consist of contiguous memory locations. The lowest address corresponds to the first element and the highest address to the last element.
- Arrays can have from one to several dimensions.
- The most common array is the *string*, which is simply an array of characters terminated by a null.

bxzsgslm1s

Raise hand

Turn on captions

Ritu Sibbal is presenting

Ritu Sibbal

DHRUV KUMAR

MEGHNA

SHOBHIT PRAKASH

RAHUL SHARMA

Meet - myg-vper-juj

meet.google.com/myg-vper-juj?authuser=0

Ritu Sibbal is presenting

ESHAN MALHO... and 57 more

1:08 PM

R

Ritu Sibbal

DHRUV KUMAR

NISHCHAY HASIJA

R

RAHUL SHARMA

MEGHNA

bxczgslm1s

Raise hand

Turn on captions

Ritu Sibbal is presenting

Single dimension Arrays

- The general form for declaring a single-dimension array is
type var_name[size];
- Like other variables, arrays must be explicitly declared so that the compiler can allocate space for them in memory. Here, *type* declares the base type of the array, which is the type of each element in the array, and *size* defines how many elements the array will hold.
- For example a 100-element array called **balance** of type **double**, is declared using the statement :

```
double balance[100];
```

Meet - myg-vper-juj

meet.google.com/myg-vper-juj?authuser=0

PARANJAY SIN... and 58 more

1:09 PM

Ritu Sibbal is presenting

- An element is accessed by indexing the array name. For example,
- `balance[3] = 12.23;`
- assigns element number 3 in **balance** the value 12.23.
- In C, all arrays have 0 as the index of their first element. Therefore, when you write
- `char p[10];`
- you are declaring a character array that has 10 elements, **p[0]** through **p[9]**.

Ritu Sibbal

DHRUV KUMAR

NISHCHAY HASIJA

RAHUL SHARMA

ANSHUL JANGRA

bxzgsjmls

Raise hand

Turn on captions

Ritu Sibbal is presenting

Meet - myg-vper-juj

meet.google.com/myg-vper-juj?authuser=0

YASH CHAUHAN and 59 more

1:10 PM

Ritu Sibbal is presenting

R

Ritu Sibbal

DHRUV KUMAR

NISHCHAY HASIJA

R

RAHUL SHARMA

ANSHUL JANGRA

bxzgsmlms

Raise hand

Turn on captions

Ritu Sibbal is presenting

Example

- The following program loads an integer array with the numbers 0 through 99

```
#include <stdio.h>
int main(void)
{
    int x[100]; /* this declares a 100-integer array */
    int t;
    /* load x with values 0 through 99 */
    for(t=0; t<100; ++t) x[t] = t;
    /* display contents of x */
    for(t=0; t<100; ++t) printf('%d ', x[t]);
    return 0;
}
```

Meet - myg-viper-juj

meet.google.com/myg-viper-juj?authuser=0

R

Ritu Sibbal is presenting

ABHISHEK KUM...
and 59 more

09:47

1:11 PM

You

- The amount of storage required to hold an array is directly related to its type and size.
- For a single dimension m array, the total size in bytes is computed as shown here:

$$\text{Total bytes} = \text{sizeof}(\text{base type}) \times \text{length of array}$$

R

Ritu Sibbal

DHRUV KUMAR

NISHCHAY HASIJA

R

RAHUL SHARMA

ANSHUL JANGRA

bxzgzslm1s

Raise hand

Turn on captions

Ritu Sibbal
is presenting

Meet - myg-vper-juj

meet.google.com/myg-vper-juj?authuser=0

Ritu Sibbal is presenting

ESHAN MALHO... and 59 more

1:18 PM

You

- C has no bounds checking on arrays.
- You could overwrite either end of an array and write into some other variable's data or even into the program's code.
- For example, this code will compile without error, but it is incorrect because the **for** loop will cause the array **count** to be overrun.

```
int count[10], i;  
/* this causes count to be overrun */  
for(i=0; i<100; i++) count[i] = i;
```

R

Ritu Sibbal

V

VISHAL VERMA

NISHCHAY HASIJA

A

AMAN SONI

AKHIL DUBEY

bcxgzslm1s

Raise hand

Turn on captions

Ritu Sibbal is presenting

Meet - myg-vper-juj

meet.google.com/myg-vper-juj?authuser=0

R

Ritu Sibbal is presenting

R

ROHAN GOEL
and 59 more

1:19 PM

You

- Single-dimension arrays are essentially lists that are stored in contiguous memory locations in index order.
- For example,
 - Element a[0] a[1] a[2] a[3] a[4] a[5] a[6]
 - Address 1000 1001 1002 1003 1004 1005 1006
 - Figure shows how array **a** appears in memory if it starts at memory location 1000 and is declared as shown here:
 - `char a[7];`

bxzgsimls

Raise hand

Turn on captions

Ritu Sibbal
is presenting

R

Ritu Sibbal

V

VISHAL VERMA

NISHCHAY HASLIJA

A

AMAN SONI

AKHIL DUBEY

Meet - myg-vper-juj

meet.google.com/myg-vper-juj?authuser=0

Ritu Sibbal is presenting

SHREYA SINGH and 59 more

1:20 PM

You

Generating a Pointer to an Array

- You can generate a pointer to the first element of an array by simply specifying the array name, without any index. For example, given

```
int sample[10]
```
- you can generate a pointer to the first element by using the name **sample**. Thus, the following
- program fragment assigns **p** the address of the first element of **sample**:

```
int *p;
int sample[10];
p = sample;
```

Address of the first element of an array can be specified by using the **&** operator.
For example, **sample** and **&sample[0]** both produce the same results.

bxzgsjmls

Raise hand

Turn on captions

Ritu Sibbal is presenting

R

Ritu Sibbal

V

VISHAL VERMA

N

NISHCHAY HASLIJA

A

AMAN SONI

A

AKHIL DUBEY

Meet - myg-vper-juj

meet.google.com/myg-vper-juj?authuser=0

R

Ritu Sibbal is presenting

ANSHUL JANGRA
and 59 more

6/7

1:22 PM

You

Passing Single-Dimension Arrays to Functions

- In C, you cannot pass an entire array as an argument to a function.
- Pass a pointer to an array by specifying the array's name without an index. For example, the following program fragment passes the address of **i** to **func1()**:

```
int main(void)
{
    int i[10];
    func1(i);
    /* . . . */
}
```

boxzgsimls

Raise hand

Turn on captions

Ritu Sibbal
is presenting

Meet - myg-viper-juj

meet.google.com/myg-vper-juj?authuser=0

SHOBHIT PRAK... and 59 more

1:30 PM

Ritu Sibbal is presenting

- If a function receives a pointer to a single-dimension array, you can declare its formal parameter in
- one of three ways: as a pointer, as a sized array, or as an unsized array. For example, to receive `i`, a function called `func1()` can be declared as

```
void func1(int *x) /* pointer */  
{  
    . . .  
}

or
```

Ritu Sibbal

VISHAL VERMA

AKHIL DUBEY

SHIRISH TI JAIN

ANSHUL JANGRA

bxzgzslm1s

Raise hand

Turn on captions

Ritu Sibbal is presenting

Meet - myg-vper-ijj

meet.google.com/myg-vper-ijj?authuser=0

R

Ritu Sibbal is presenting

V

VISHAL MALIK
and 59 more

1:30 PM

You

- All three declaration methods produce similar results because each tells the compiler that an integer pointer is going to be received.
- The first declaration actually uses a pointer. The second employs the standard array declaration. In the final version, a modified version of an array declaration simply specifies that an array of type **int** of some length is to be received.

bxzgsjmls

R

...

Ritu Sibbal

V

VISHAL VERMA

AKHIL DUBEY

S

SHIRISH TI JAIN

ANSHUL JANGRA

Raise hand

Turn on captions

Ritu Sibbal
is presenting