

The screenshot shows a Google Meet interface. At the top, the browser tabs include 'New Tab', 'Meet - tqm-snye-1ct', 'Something went wrong', and 'Meet - jzo-qgbb-gib'. The address bar shows the URL 'meet.google.com/jzo-qgbb-gib?authuser=0'. The top bar of the Meet window indicates 'Ritu Sibbal is presenting' and 'ROHAN GOEL and 42 more' are in the call. The time is 1:11 PM. The main content area displays the title 'Programming exercise 7' and a bullet point: 'Write a program in 'C' using pointers to implement insertion and deletion in a queue. A queue is a data structure that follows first in first out i.e. the element that goes in first is the first to come out'. On the right, a vertical list of participants shows avatars for 'You', 'APURVA MISHRA', 'RITIK YADAV', 'Ritu Sibbal' (highlighted), and 'ROHIT SHARMA'. At the bottom, there are icons for 'Raise hand', 'Turn on captions', and 'Ritu Sibbal is presenting'.

Programming exercise 7

- Write a program in 'C' using pointers to implement insertion and deletion in a queue. A queue is a data structure that follows first in first out i.e. the element that goes in first is the first to come out

New Tab

Meet - tajm-nye-fct

Something went wrong

Something went wrong

Meet - jzo-qgbb-gjb

meet.google.com/jzo-qgbb-gjb?authuser=0

R

Ritu Sibbal is presenting

R

ROHAN GOEL and 42 more

1:11 PM

You

## Pointers and Arrays

- There is a close relationship between pointers and arrays. Consider this program fragment:
- `char str[80], *p1;`
- `p1 = str;`
- Here, **p1** has been set to the address of the first array element in **str**. To access the fifth element in **str**, you could write
- `str[4]`
- or
- `*(p1+4)`
- Both statements will return the fifth element. Remember, arrays start at 0. To access the fifth element, you must use 4 to index **str**. You also add 4 to the pointer **p1** to access the fifth element because **p1** currently points to the first element of **str**.

bcxgzslm1s

Raise hand

Turn on captions

Ritu Sibbal is presenting

You

A

APURVA MISHRA

RITIK YADAV

R

Ritu Sibbal

ROHIT SHARMA

New TabMeet - tam-nyie-fctSomething went wrongSomething went wrongMeet - jzo-qgbb-gjb

meet.google.com/jzo-qgbb-gjb?authuser=0

1:13 PM

Ritu Sibbal is presenting

DHRUV KUMAR and 47 more

54

1:13 PM

You

You

A

APURVA MISHRA

M

MANAN SURI

R

Ritu Sibbal

ROHIT SHARMA

bxzgsmlms

Raise hand

Turn on captions

Ritu Sibbal is presenting

- An array name without an index returns the starting address of the array, which is the address of the first element.)
- C provides two methods of accessing array elements: pointer arithmetic and array indexing. Although the standard array-indexing notation is sometimes easier to understand, pointer arithmetic can be faster.
- Since speed is often a consideration in programming, C programmers often use pointers to access array elements

New TabMeet - tam-snye-fctSomething went wrongSomething went wrongMeet - jzo-qgbb-gjb

meet.google.com/jzo-qgbb-gjb?authuser=0

1:13 PM

LEKHA SONI and 47 more

R

Ritu Sibbal is presenting

# Array of Pointers

- Pointers can be arrayed like any other data type. The declaration for an **int** pointer array of size 10 is
- `int *x[10];`
- To assign the address of an integer variable called **var** to the third element of the pointer array, write
- `x[2] = &var;`
- To find the value of **var**, write
- `*x[2]`

You

A

APURVA MISHRA

M

MANAN SURI

R

Ritu Sibbal

ROHIT SHARMA

bcxgzslm1s

Raise hand

Turn on captions

Ritu Sibbal is presenting

Browser tabs: New Tab, Meet - tqm-nye-fc1, Something went wrong, Something went wrong, Meet - jzo-qgbb-gjb

Address bar: meet.google.com/jzo-qgbb-gjb?authuser=0

Top bar: Ritu Sibbal is presenting, AMANJOT SINGH and 53 more, 1:16 PM, You

## Multiple Indirection

- You can have a pointer point to another pointer that points to the target value.
- This situation is called *multiple indirection*, or *pointers to pointers*.

Participants: You, KAUSTAV SARKAR, MANAN SURI, Ritu Sibbal, APURVA MISHRA

Bottom bar: bcxgzslm1s, Raise hand, Turn on captions, Ritu Sibbal is presenting

Single and Multiple indirection

```
graph LR; P1[Pointer address] --> V1[Variable value]; P2[Pointer address] --> P3[Pointer address]; P3 --> V2[Variable value];
```

Single Indirection

Multiple Indirection

bxzgzslms

ISHITA NAYAK and 55 more

1:18 PM

You

KAUSTAV SARKAR

MANAN SURI

Ritu Sibbal

APURVA MISHRA

Ritu Sibbal is presenting

meet.google.com/jzo-qgbb-gjb?authuser=0

Ritu Sibbal is presenting

MOKSHI SHAR... and 55 more

1:18 PM

- The Figure helps to clarify the concept of multiple indirection.
- The value of a normal pointer is the address of the object that contains the desired value.
- In the case of a pointer to a pointer, the first pointer contains the address of the second pointer, which points to the object that contains the desired value.
- Multiple indirection can be carried on to whatever extent desired.
- In fact, excessive indirection is difficult to follow and prone to conceptual errors.

bcxgzslm1s

Raise hand Turn on captions Ritu Sibbal is presenting

Participants: You, KAUSTAV SARKAR, AMAN SONI, Ritu Sibbal, ROHIT SHARMA

New Tab

Meet - tqm-snye-fct

Something went wrong

Something went wrong

Meet - jzo-qgbb-gjb

meet.google.com/jzo-qgbb-gjb?authuser=0

1:20 PM

SHUBHAM AGGAR... and 55 more

R

Ritu Sibbal is presenting

- A variable that is a pointer to a pointer must be declared as such. You do this by placing an additional asterisk in front of the variable name. For example, the following declaration tells the compiler that **newbalance** is a pointer to a pointer of type **float**:
- `float **newbalance;`
- **newbalance** is not a pointer to a floating-point number but rather a pointer to a **float** pointer.

You

M

MANAN SURI

A

AMAN SONI

R

Ritu Sibbal

ROHIT SHARMA

bcxzgslm1s

Raise hand

Turn on captions

Ritu Sibbal is presenting



New Tab

Meet - tqm-snye-fct

Something went wrong

Something went wrong

Meet - jzo-qgbb-gjb

meet.google.com/jzo-qgbb-gjb?authuser=0

Ritu Sibbal is presenting

AKASH SHEKHAR PAULI... and 55 more

1:21 PM

You

## Example

```
#include <stdio.h>
int main(void)
{
    int x, *p, **q;
    x = 10;
    p = &x;
    q = &p;
    printf("%d", **q); /* print the value of x */
    return 0;
}
```

- Here, **p** is declared as a pointer to an integer and **q** as a pointer to a pointer to an integer. The call to **printf()** prints the number **10** on the screen.

bxzgzslm1s

Raise hand

Turn on captions

Ritu Sibbal is presenting

You

MANAN SURI

AMAN SONI

R

Ritu Sibbal

ROHIT SHARMA

New TabMeet - tqm-snye-fctSomething went wrongSomething went wrongMeet - jzo-qgbb-gjb

meet.google.com/jzo-qgbb-gjb?authuser=0

12:22 PM

Ritu Sibbal is presentingDHANANJAY G... and 55 more

# Initialising Pointers

- After a nonstatic, local pointer is declared but before it has been assigned a value, it contains an unknown value. (Global and static local pointers are automatically initialized to null.)
- If you try to use the pointer before giving it a valid value, it might crash your program or possibly your computer's operating system as well
- There is an important convention that most C programmers follow when working with pointers: A pointer that does not currently point to a valid memory location is given the value null (which is zero). Null is used because C guarantees that no object will exist at the null address. Thus, any pointer that is null implies that it points to nothing and should not be used.

bcxgzslm1s

Raise hand

Turn on captions

Ritu Sibbal is presenting

You

M

MANAN SURI

A

AMAN SONI

R

Ritu Sibbal

ROHIT SHARMA

New TabMeet - tajm-snye-fctSomething went wrongSomething went wrongMeet - jzo-qgbb-gjb

meet.google.com/jzo-qgbb-gjb?authuser=0

Ritu Sibbal is presentingSHUBHAM AGGAR... and 55 more1:24 PM

- One way to give a pointer a null value is to assign zero to it. For example, the following initializes **p** to null.  
`char *p = 0;`
- Additionally, many of C's headers, such as `<stdio.h>`, define the macro `NULL`, which is a null pointer constant. Therefore a pointer assigned null value using a statement such as this:  
`p = NULL;`
- However, just because a pointer has a null value, it is not necessarily "safe." The use of null to
- indicate unused pointers is simply a convention that programmers follow. It is not a rule enforced by
- the C language. For example, the following sequence, although incorrect, will still be compiled
- without error:  
`int *p = 0;`  
`*p = 10; /* wrong! */`
- In this case, the assignment through **p** causes an assignment at 0, which will usually cause a program
- crash.

bcxgzslm1s

Raise handTurn on captions

Ritu Sibbal is presenting

You

M

MANAN SURI

A

AMAN SONI

R

Ritu Sibbal

ROHIT SHARMA