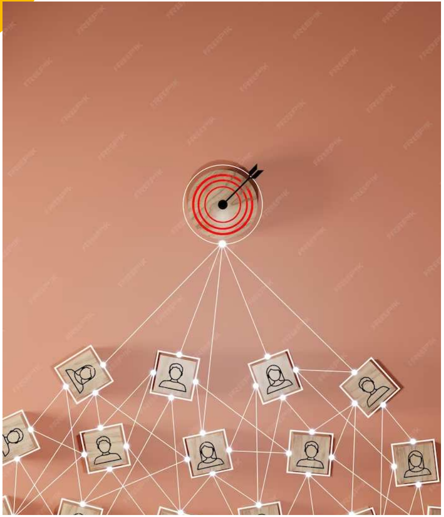
MASTERING C FUNDAMENTALS: POINTERS, ARRAYS, AND BEYOND

Biraj Shrestha
Kripash Shrestha
Ishwodhan Maharjan



CONTENTS

INTRODUCTION TO POINTERS

- Pointers are **variables** that store **memory addresses**.
- **Declaration:** int *ptr; declares a pointer to an integer.
- **Initialization:** int x = 10; int *ptr = &x;

PASS BY VALUE

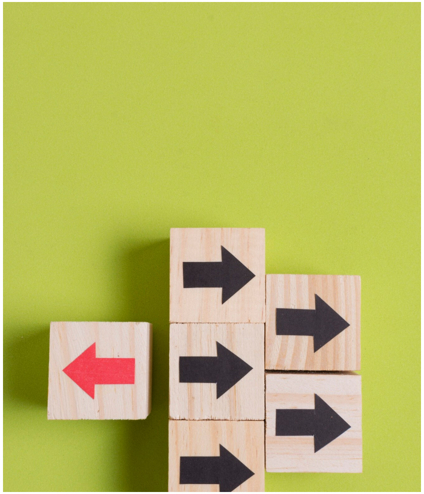
- Function parameters receive a copy of the argument’s value.
- Changes to the parameter do not affect the original variable.
- Efficient for simple data types but not suitable for large objects.

PASS BY REFERENCE

- Uses pointers to pass the memory address of a variable.
- Changes to the parameter affect the original variable.
- Allows efficient manipulation of large data structures.

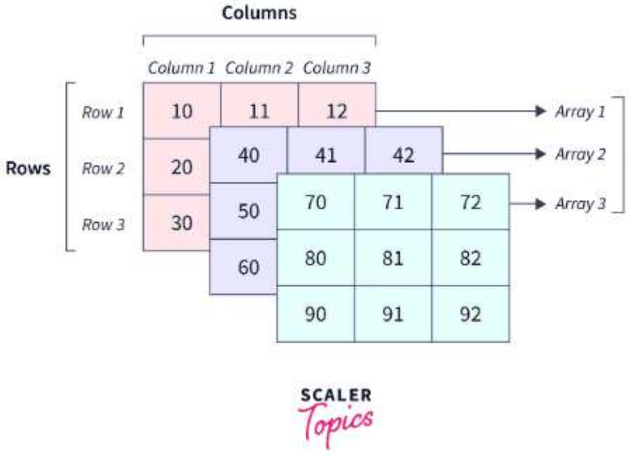
ARRAYS

- Collections of elements of the same data type.
- Declaration: int arr[5]; declares an integer array of size 5.
- Accessing elements: arr[0], arr[1], ...

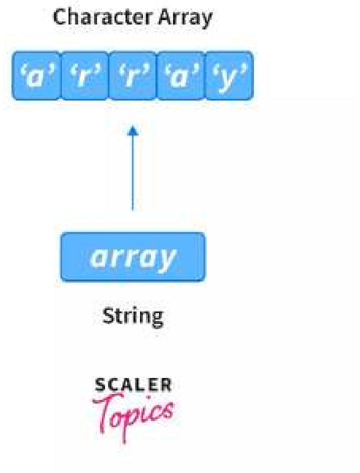


MULTI-DIMENSIONAL ARRAYS

- 2D arrays are arrays of arrays.
- Declaration: int matrix[3][3]; declares a 3x3 matrix.
- Accessing elements: matrix[0][0], matrix[1][2], ...



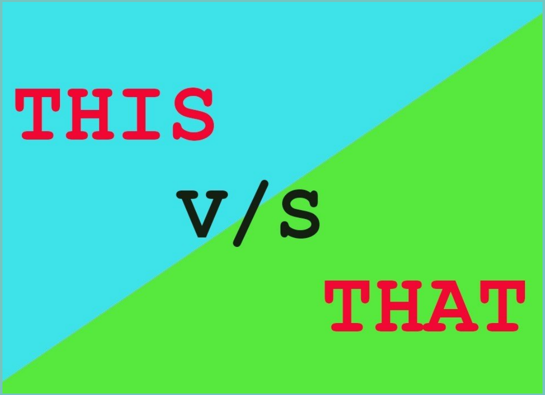
STRING AND CHARACTER ARRAYS



- Character arrays are arrays of characters, often used for strings.
- Strings are null-terminated character arrays.
- Example: char str[10] = “Hello”;

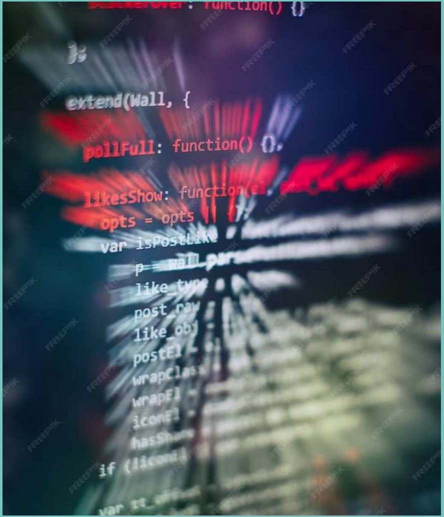
DIFFERENCE BETWEEN ARRAY AND POINTERS

- Arrays decay into pointers when passed to functions.
- Array size is fixed; pointer can be reassigned.
- sizeof(array) gives total size; sizeof(pointer) gives pointer size.



LIBRARY FUNCTIONS FOR STRING HANDLING

- **strlen(str**): Returns the length of the string.
- **strcpy(dest, src)**: Copies the source string to the destination.
- **strcat(dest, src)**: Concatenates the source string to the destination.



CONCLUSION

In summary, our exploration of pointers, arrays, and related concepts today lays a strong foundation for effective C programming. Pointers grant us precision and control over memory, enabling efficient manipulation of data. The interplay between pass by value and pass by reference, coupled with insights into arrays and multi-dimensional arrays, enhances our ability to structure and access information.

Thanks!

Do you have any questions?