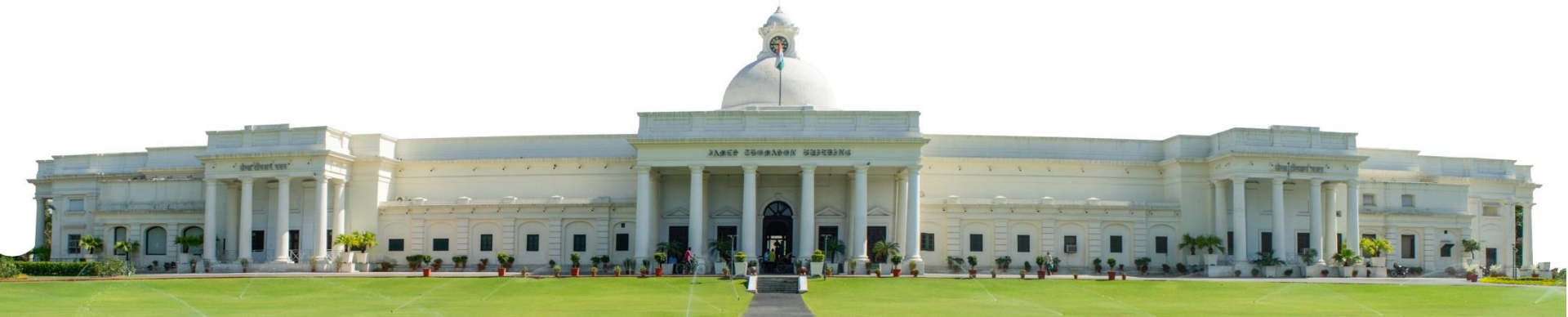
Programming with C and C++

CSC-101 (Lecture 15)

Dr. R. Balasubramanian
Professor
Department of Computer Science and Engineering
Mehta Family School of Data Science and Artificial Intelligence
Indian Institute of Technology Roorkee
Roorkee 247 667

bala@cs.iitr.ac.in
https://faculty.iitr.ac.in/cs/bala/



Reading and Writing Character Array

```c
#include <stdio.h>

int main() {
    char input[100]; // Declare a character array to store input
    int i = 0;
    char c;

    printf("Enter a string (up to 100 characters): ");

    // Use a loop to read characters until the array is full
    // or Enter (newline) is pressed
    while (i < sizeof(input) - 1 && (c = getchar()) != '\n') {
        input[i] = c; // Store the character in the array
        i++;
    }

```

https://ideone.com/VrREiG

```
17      // Null-terminate the character array to make it a valid C string
18      input[i] = '\0';
19
20      printf("You entered: %s\n", input); // Print the input
21
22      return 0;
23  }
24
```

stdin

```
Welcome to IIT Roorkee
```

stdout

```
Enter a string (up to 100 characters): You entered: Welcome to IIT Roorkee
```

Printing Character Array

```c
#include<stdio.h>
int main()
{
    char str[10]="Hello IITR";
    //printf("Enter a String: ");
    //scanf("%s", &str);

    for (int i=0; i<10; i++)
    printf("%c", str[i]);
}
```

input Output

Success #stdin #stdout 0s 5516KB

```
Hello IITR
```

https://ideone.com/NZkTUu

Arrays - Example 1

</> source code

https://ideone.com/U8V9mi

```
1  #include <stdio.h>
2
3  int main(void) {
4      int a[8]={20,30,40};
5      for (int i=0; i<8; i++)
6      printf("%d ",a[ i ]);
7      return 0;
8  }
9
```

input Output

Success #stdin #stdout 0s 5392KB

```
20 30 40 0 0 0 0 0
```

Example 2

</> source code

https://ideone.com/K9vvrU

```c
#include <stdio.h>

int main(void) {
    int a[4]={10,20,30,40};
    for (int i=0; i<8; i++)
    printf("%d ",a[ i ]);
    return 0;
}

```

stdout

```
10 20 30 40 1378877920 22025 -91800064 -1841019489
```

Example 3

</> source code

https://ideone.com/kcbBIT

```c
#include <stdio.h>

int main(void) {
    int a[4]={10,20,30};
    for (int i=0; i<8; i++)
    printf("%d ",a[ i ]);
    return 0;
}
```

stdout

```
10 20 30 0 -1573006880 22002 2033631232 2068984768
```

Example 4

</> source code

https://ideone.com/Rkd9pu

```
1   #include <stdio.h>
2
3   int main(void) {
4       int a[]={10,20,30};
5       for (int i=0; i<8; i++)
6       printf("%d ",a[ i ]);
7       return 0;
8   }
9
10
```

stdout

```
10 20 30 -551353856 -234756495 0 0 2134995424
```

Some Famous Problems

- Write a program to find the kth largest element in an unsorted array.
- Suppose you have an array in ascending order. Write a program to sort it in descending order using the fewest possible steps.
- Binary Array sorting problem
- Implement Tricolor Flag Sorting Problem.
- Write a program to find the majority element in an array (an element that appears more than n/2 times, where n is the size of the array).

- Write a program to check if two strings are anagrams of each other in C.
  - "listen" and "silent",
  - "cinema" and "iceman“
  - "debit card" and "bad credit" are anagrams

- Take an input n where n is an integer. Write a program to print the prime numbers less than n (Famous **Sieve of Eratosthenes algorithm** takes $n \log\log(n)$ steps and **Sieve of Atkin** takes $n/\log\log(n)$ steps).

Binary Search Algo'

```c
#include <stdio.h>

int main() {
    int arr[] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
    int size = sizeof(arr) / sizeof(arr[0]);
    int target = 6;

    int left = 0;
    int right = size - 1;
    int found = -1; // Initialize a variable to track if the element is found

    while (left <= right) {
        int mid = (left + right) / 2;

        if (arr[mid] == target) {
            found = mid; // Element found, store its index
            left = right + 1; // Set left > right to exit the loop
        } else if (arr[mid] < target) {
            left = mid + 1; // Search the right half
        } else {
            right = mid - 1; // Search the left half
        }
    }

```

```
25      if (found != -1) {
26          printf("Element found at index %d\n", found);
27      } else {
28          printf("Element not found in the array\n");
29      }
30
31      return 0;
32  }
33
```

https://ideone.com/XjhcDe

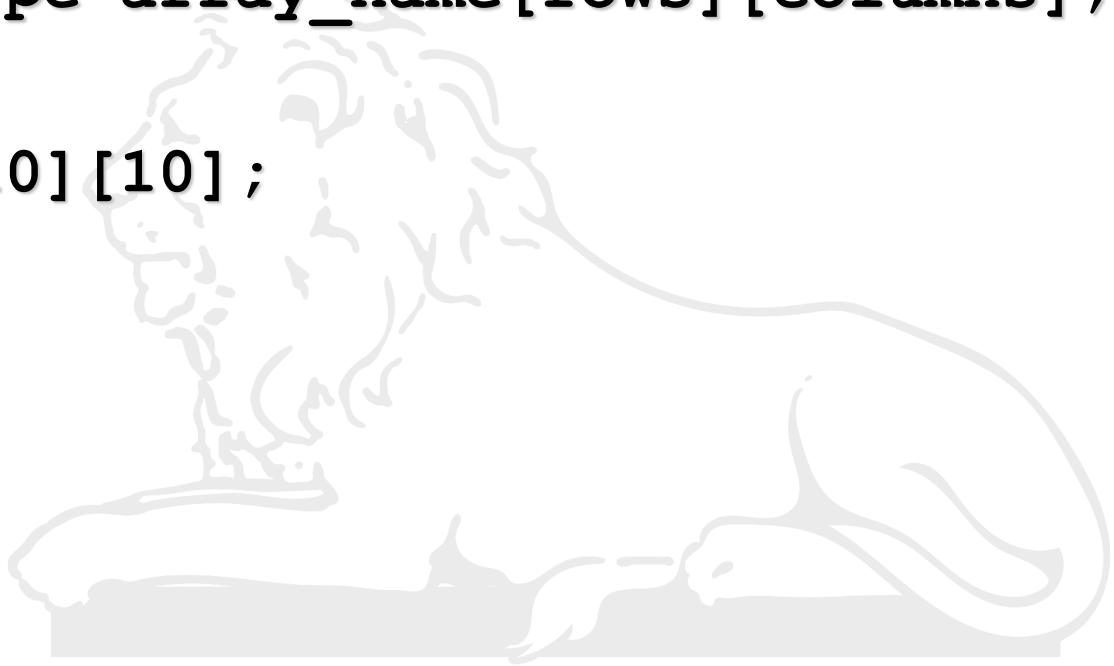
input Output

Success #stdin #stdout 0s 5472KB

```
Element found at index 5
```

Two Dimensional Array in C

- Declaration of two dimensional Array in C
- `data_type array_name[rows][columns];`
- `int a[10][10];`



</> source code

```c
#include <stdio.h>
void main ()
{
    int arr[3][3],i,j;
    for (i=0;i<3;i++)
    {
        for (j=0;j<3;j++)
        {
            printf("Enter a[%d][%d]: ",i,j);
            scanf("%d",&arr[i][j]);
        }
    }
```

```
13      printf("\n printing the elements ....\n");
14      for(i=0;i<3;i++)
15      {
16          printf("\n");
17          for (j=0;j<3;j++)
18          {
19              printf("%d\t",arr[i][j]);
20          }
21      }
22  }
```

https://ideone.com/dL9sjF

Success #stdin #stdout 0.01s 5536KB

comments (0)

stdin

copy

```
1 2 3 4 5 6 7 8 9
```

stdout

copy

```
Enter a[0][0]: Enter a[0][1]: Enter a[0][2]: Enter a[1][0]: Enter a[1][1]: Enter a[1]
[2]: Enter a[2][0]: Enter a[2][1]: Enter a[2][2]:
 printing the elements ....

1   2   3
4   5   6
7   8   9
```

Thank You!