

C++ LAUNCHPAD



Lecture-3

Fundamentals -II

- Programming Fundamentals contd..

Utkarsh Nath

BT – 5: Circular Jail Cell

There is a circular jail with 100 cells numbered 1-100. Each cell has an inmate and the door is locked. One night the jailor gets drunk and starts running around the jail in circles. In his first round he opens each door. In his second round he visits every 2nd door (2,4,6---) and shuts the door. In the 3rd round he visits every 3rd door (3,6,9---) and if the door is shut he opens it, if it is open he shuts it. This continues for 100 rounds (i.e. 4,8,12 ---; 5,10,15 ---; ---; 49,98 etc.) and exhausted the jailor falls down.

How many prisoners found their doors open after 100 rounds?

Square root of a given number?



sizeof() operator ?

Type conversion!

- Implicit – Based on the operator and operands
- Explicit – (new type) expression

Operators we have seen

- Unary [+, -]
- Arithmetic [+, -, /, *, %]
- Brackets [()]
- Assignment [=]
- Relational [==, !=, >, <, >=, <=]
- Logical Operators [&&, ||, !]

PS - 1: Relational Operators and Logical Operators always Evaluate to 0 or 1

PS - 2: For logical evaluation any non-zero value is true.

PS - 3: Evaluation of a logical expressions stops as soon as the final value is known.



Some more operators!

- Arithmetic – [++ , --]
- Bitwise Operators – [&, | , ~, ^, <<, >>]
- Compound assignment operators – [+=, *=, /=, %=, &=, |=, ^=, <<=, >>=]

Precedence & Associativity

TABLE 2-1. PRECEDENCE AND ASSOCIATIVITY OF OPERATORS

OPERATORS	ASSOCIATIVITY
() [] -> .	left to right
! ~ ++ -- + - * & (type) sizeof	right to left
* / %	left to right
+ -	left to right
<< >>	left to right
< <= > >=	left to right
== !=	left to right
&	left to right
^	left to right
	left to right
&&	left to right
	left to right
?:	right to left
= += -= *= /= %= &= ^= = <<= >>=	right to left
,	left to right

Unary +, -, and * have higher precedence than the binary forms.

Another type of a loop!

```
for(initialization; condition; step) {  
    // do something  
}
```

Lets convert some problems to use for

- Print all prime numbers between 2 to N
- Reverse a number
- Print the following pattern

A

BB

CCC

DDDD

Time to try?

- Given an integer n, count number of bits set (number of bits which are 1) in it.
- Given N, Print following pattern (FOR N = 5)

ABCDEEDCBA

ABCDDCBA

ABCCBA

ABBA

AA

What is next class about?

- Some more basic constructs
- Arrays

C++ LAUNCHPAD



Thank You!

Utkarsh Nath