July 11, 2015

Launchpad

Lecture - 3

Programming Fundamentals - 2

Aman Bahl

Any doubts?

Brain Teasers?

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?

BT – 6: Greedy Pirates

A pirate ship captures a treasure of 1000 golden coins. The treasure has to be split among the 5 pirates: 1, 2, 3, 4, and 5 in order of rank. The pirates have the following important characteristics: infinitely smart, bloodthirsty, greedy. Starting with pirate 5 they can make a proposal how to split up the treasure. This proposal can either be accepted or the pirate is thrown overboard. A proposal is accepted if and only if a majority of the pirates agrees on it. **What proposal should pirate 5 make?**

BT – 7: Infinite Quarter Sequence

You are wearing a blindfold and thick gloves. An infinite number of quarters are laid out before you on a table of infinite area. Someone tells you that 20 of these quarters are tails and the rest are heads. He says that if you can split the quarters into 2 piles where the number of tails quarters is the same in both piles, then you win all of the quarters. You are allowed to move the quarters and to flip them over, but you can never tell what state a quarter is currently in (the blindfold prevents you from seeing, and the gloves prevent you from feeling which side is heads or tails). **How do you partition the quarters so that you can win them all?**

BT – 8: Daughters' Ages

Local Berkeley professors Dr. X and Dr. Y bump into each after a long time.

X hey! how have you been?

Y great! i got married and i have three daughters now

X really? how old are they?

Y well, the product of their ages is 72, and the sum of their ages is the same as the number on that building over there

X right, ok ... oh wait ... hmm, i still don't know

Y oh sorry, the oldest one just started to play the piano

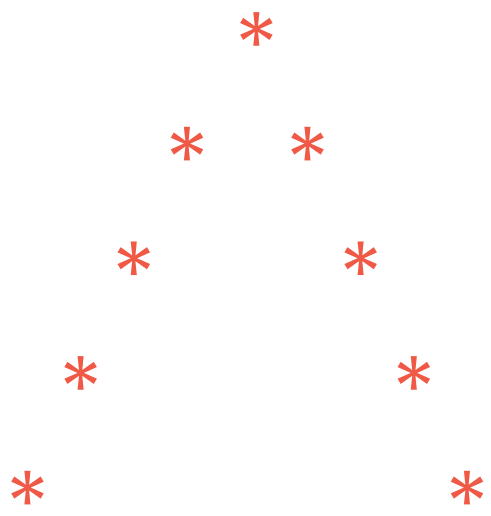
X wonderful! my oldest is the same age!

How old are the daughters?

CODING BLOCKS

Warm up!

Print following pattern:



CODING BLOCKS

Square root of a given number?

Constants

C++ has two types of constants

I. Literals
   i. Integer [212, 215u,0x4b,077, 30ul, 75L)
   ii. Floating [3.14, 31459E-5]
   iii. Boolean [true, false]
   iv. Character ['a', '\n', '\t', '\\', '\'']
   v. String ["Coding Blocks"]
II. Symbolic Constants
   i. #define preprocessor
   ii. const keyword

Type conversion!

I. Implicit – Based on the operator and operands
II. Explicit – (new type) expression

Operators we have seen

I. Unary [+, -]
II. Arithmetic [+, -, /, *, %]
III. Brackets [()]
IV. Assignment [=]
V. Relational [==, !=, >, <, >=, <=]
VI. 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!

I. Arithmetic – [++ , --]
II. Bitwise Operators – [&, | , ~, ^, <<, >>]
III. 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

Another type of a loop!

for(*initialization;condition;step*) {}

Lets convert some problems to use for

I. Print all prime numbers between 2 to N
II. Reverse a number
III. Print the following pattern

1

0 1

1 0 1

0 1 0 1

1 0 1 0 1

To alter normal flow of a loop

I. break;
II. continue;

Easy?

In the below code, change/add only one character and print '*' exactly 20 times.

```
int main() {
        int i, n = 20;
        for (i = 0; i < n; i--)
                cout << "*";
        return 0;
}
```

Time to try?

I. Given an integer n, count number of bits set (number of bits which are 1) in it.

II. Given N,Print following pattern (FOR N = 5)

ABCDEEDCBA

ABCDDCBA

ABCCBA

ABBA

AA

What is next class about?

I. Some more basic constructs
II. Arrays