

# Computer Programming

Dr. Deepak B Phatak

Dr. Supratik Chakraborty

Department of Computer Science and Engineering  
IIT Bombay

**Session: Sequential Execution in C++ Programs**

# Quick Recap of Some Relevant Topics

---



- Structure of a simple C++ program
- Variables and type declarations
- Assignment statements
- Arithmetic and logical expressions

# Overview of This Lecture

---



- Simple C++ programs with sequential execution
- Programming to solve problems

# Recalling Some Useful Facts



- Program: sequence of **compiler directives**, **declarations**, **instructions**
- Compiler directives used during compilation
- Declarations tell what variables (mostly) to be used
  - How much memory to allocate for variables
  - How to interpret stored sequence of bits
- Instructions tell what computer (or Mr. Dumbo) should do
  - By sequencing instructions carefully, we can get computer to do highly non-trivial tasks efficiently
  - Art of Programming !!!

# Simplest Programs

- Simplest programs: linear sequence of instructions
  - Computer executes instructions in linear order

; used to separate one executable statement from next

# Simplest Programs: Another Example



- Simplest programs: linear sequence of instructions
  - Computer executes instructions in linear order

# Sequential Execution of Programs



- All variables must be declared before being used
- Executable statements executed in sequence from “top” to “bottom”
- If a statement changes value of a variable, and a “later”/“below” statement refers to the variable, it sees the changed value
- Executing a “return” statement returns control back to the caller (function/OS)
  - Subsequent statements not executed

# Programming To Solve Problems



- Given a problem:  
**“Given two integers A and B, find if B divides A?”**
- Think of sequence of steps you would take if solving by pencil-and-paper
  - Find the remainder of A on division by B
  - If remainder is 0, then B divides A, otherwise not
- Think in terms of program statements seen so far
  - Can we sequence them in a linear order to do the same thing as we did on pencil-and-paper?
  - **We didn't learn to conditionally execute “If ... then ...”, have we?**

# Programming to Solve Problems



- We haven't learnt to conditionally execute instructions
  - But could there be a way to solve our problem by a linear sequence of instructions?
- How about
  - Calculate remainder R of A divided by B  
 $R = A \% B$
  - Set a boolean flag to true if R equals 0 and to false otherwise  
`flag = (R == 0);`
  - Output flag  
flag is true if and only if B divides A

# A Simple C++ Program



```
#include <iostream>
using namespace std;
// Program to find if B divides A
int main() {
    int A, B, R;          // Variable declarations
    bool dividesFlag; // Variable declarations
    cout << "Give A and B: " << endl;
    cin >> A >> B;
    R = A % B; // Remainder of A divided by B
    dividesFlag = (R == 0); // Is the remainder 0?
    cout << "Does B divide A? " << dividesFlag << endl;
    return 0;
}
```

Note the importance of the sequence:  
What happens if we swap the order of  
 $R = A \% B$  and  $\text{dividesFlag} = (R == 0)$  ?

# Summary

---



- Sequential execution of statements in C++ programs
- Problem solving by programming
  - Without conditional execution of statements, we are handicapped
  - Still, some interesting problems can be solved by sequential execution of programs