July 5, 2015

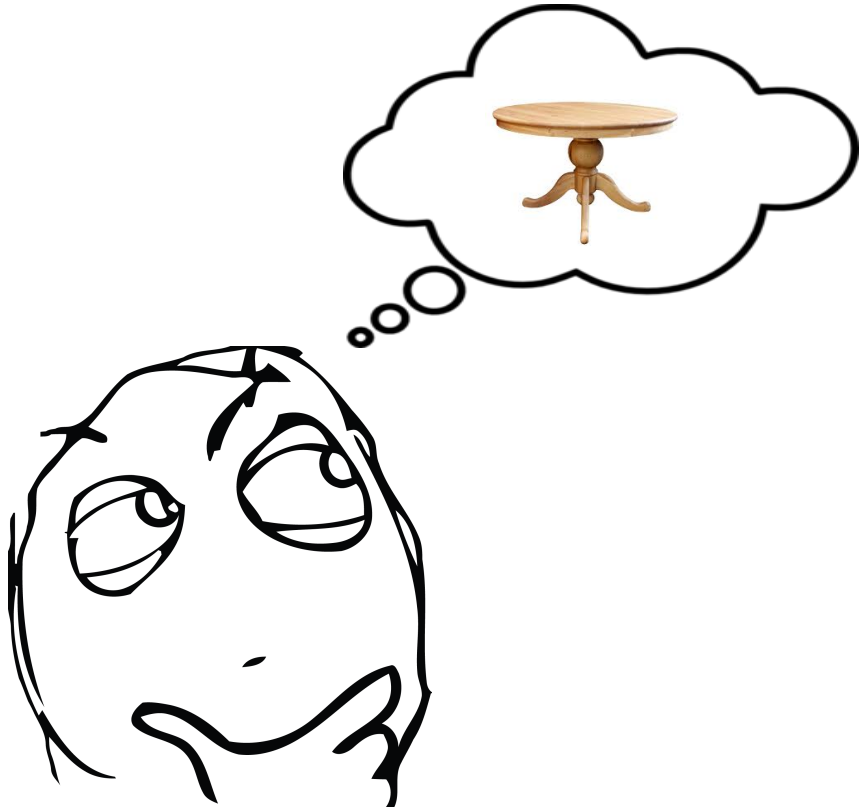
Launchpad

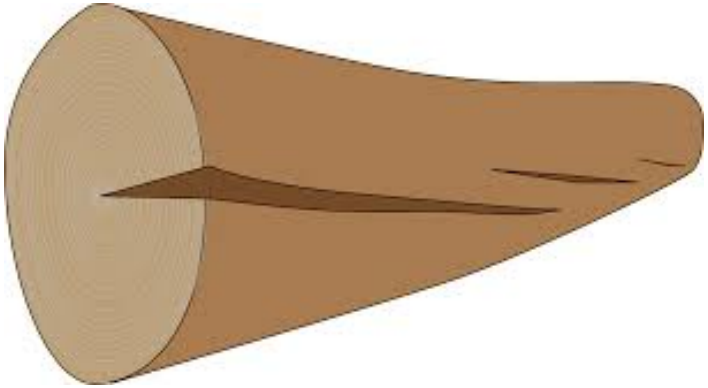
Lecture - 2

Basics of Problem Solving

Aman Bahl

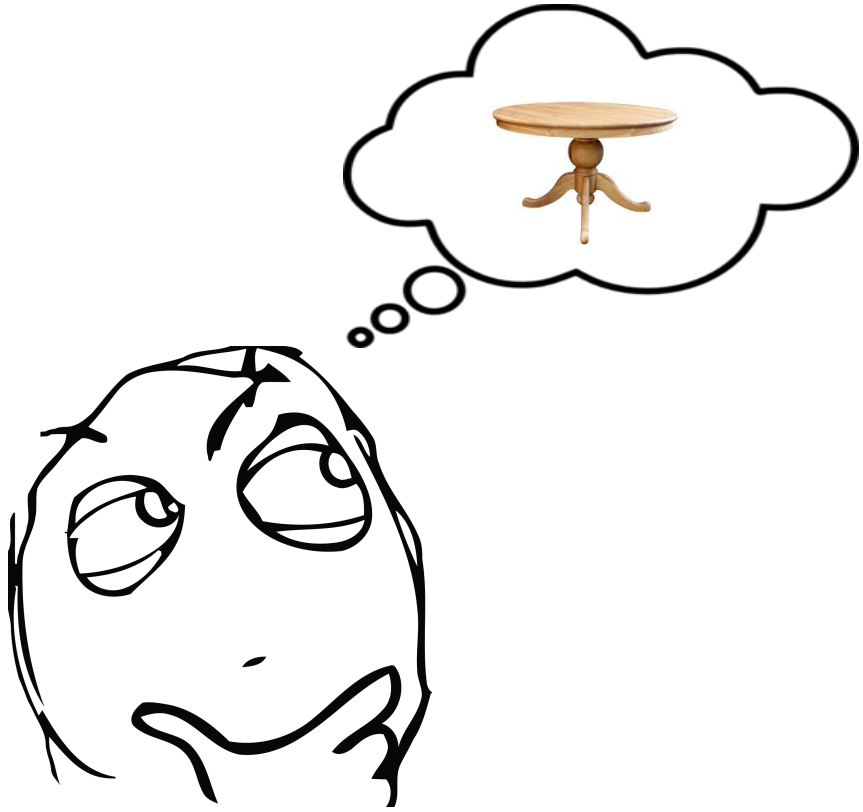
Programming is like carpentry

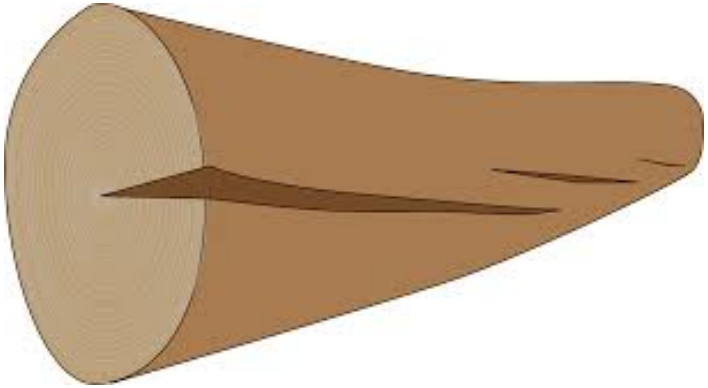




CODING BLOCKS

Programming is like carpentry





What does he need?

CODING BLOCKS

Before he starts cutting wood he needs **"A clear idea of what he wants to build and how is he going to build"**

Programming is like carpentry

Before he starts cutting wood he needs **"A clear idea of what he wants to build and how is he going to build".**

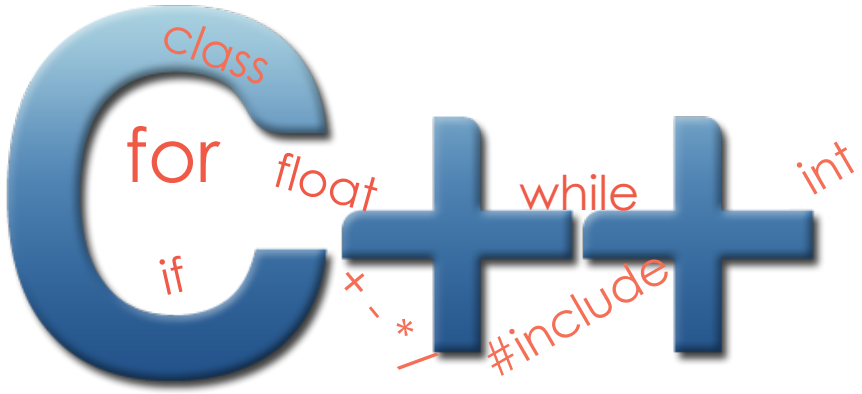
Tool Box!

What does he need?

CODING BLOCKS

We need a toolbox for programming

I. We call it programming language
II. We get an idea, we choose appropriate tools to build our idea(program)
III. For this course we prefer C++



CODING BLOCKS

Tools are we going to learn/use today

CODING BLOCKS

Tools are we going to learn/use today

CODING BLOCKS

Tools are we going to learn/use today

1. **Input / Output**
2. **Memory and variable declaration**
    1. Data types: char, int, float
3. **Operators**
    1. basic arithmetic and relation operators
4. **Control statements -** Conditional statements
5. **Control statements -** Loops

Tools are we going to learn/use today

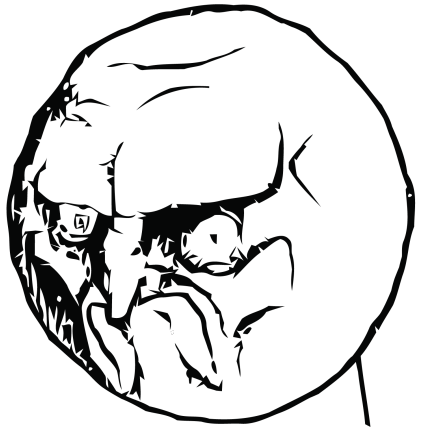
1. **Input / Output**
2. **Memory and variable declaration**
    1. Data types: char, int, float
3. **Operators**
    1. basic arithmetic and relation operators
4. **Control statements -** Conditional statements
5. **Control statements -** Loops

Input / Output

I. For this we will have to understand
   I. Keyboard layout
   II. Keyboard drivers
   III. Monitor drivers
   IV. their connection to our compiler, terminal shell ….

Input / Output

I. For this we will have to understand
   I. Keyboard layout
   II. Keyboard drivers
   III. Monitor drivers
   IV. their connection to our compiler, terminal shell ….



NO.

CODING BLOCKS

Input / Output

I. We will use standard libraries for input/ output which people have already written thus not reinventing the wheel

Input: cin

I. cin >> roll_number;
   I. -- read an integer from user

II. cin >> number1 >> number2;
   I. If user types 3 and 5
   II. number1 will have value 3
   III. number2 will have value 5

Output: cout

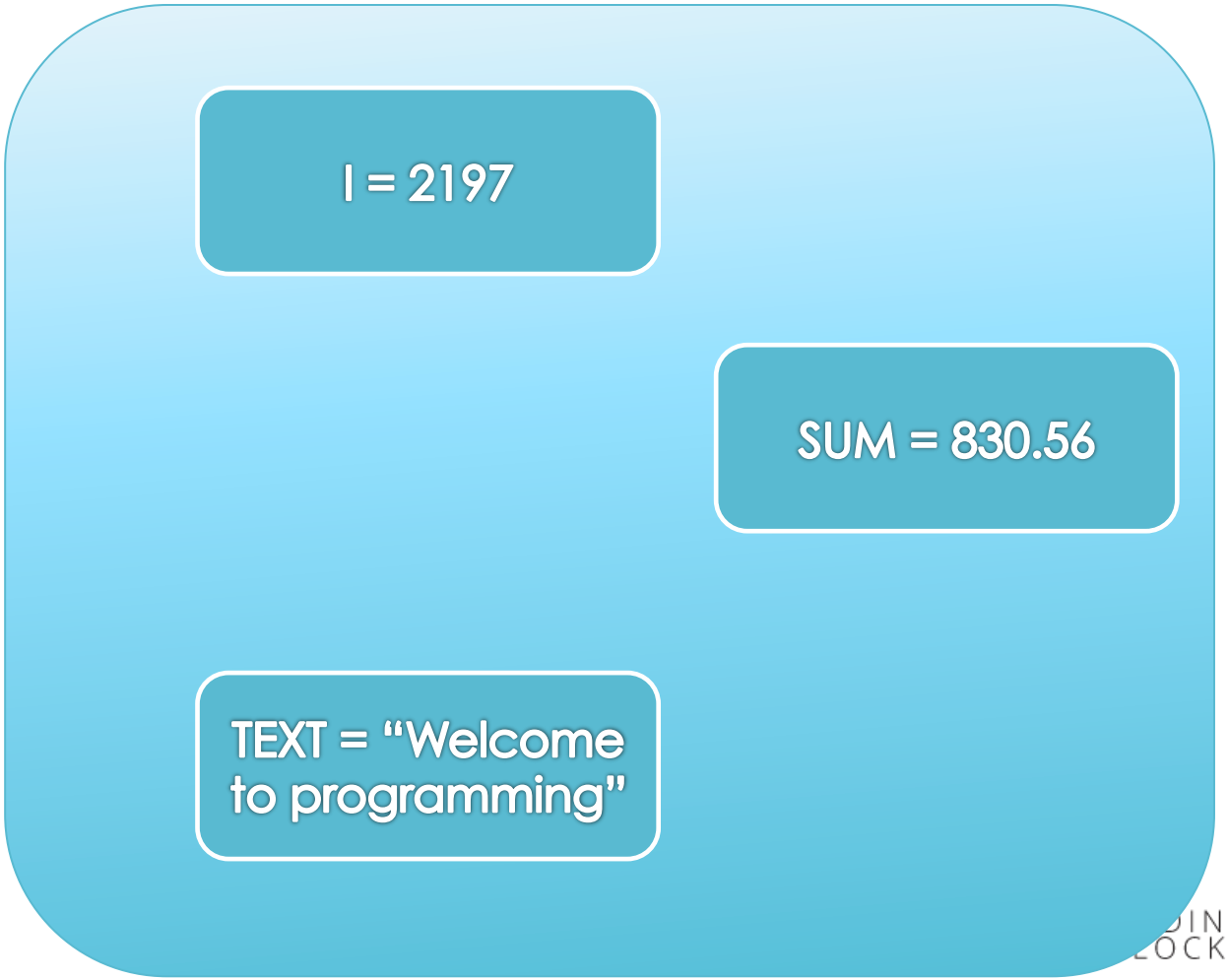
I. cout << "This is a print statement"
   I. This will print whatever text you give with in quotes

II. cout << number1 << " " << number2
   I. This will print number1 followed by space and then number2

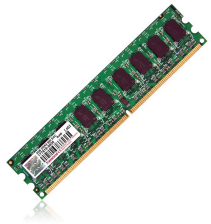
Tools are we going to learn/use today

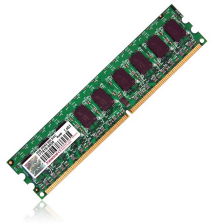
1. **Input / Output**
2. **Memory and variable declaration**
    1. Data types: char, int, float
3. **Operators**
    1. basic arithmetic and relation operators
4. **Control statements -** Conditional statements
5. **Control statements -** Loops

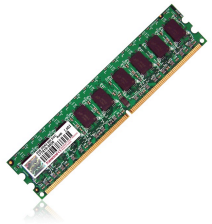
Need of memory?



Need of memory?







I = 2197

SUM = 830.56

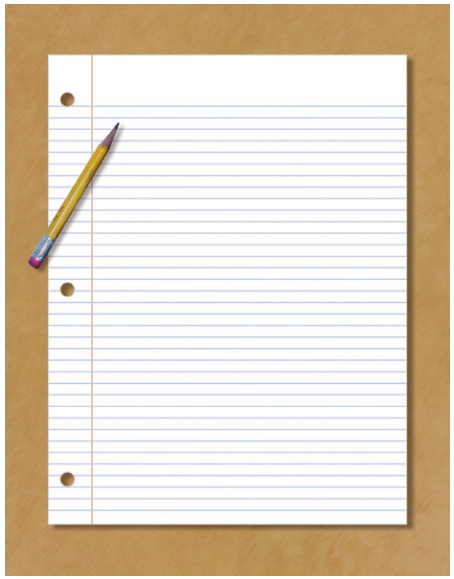
TEXT = “Welcome to programming”

Need of memory?

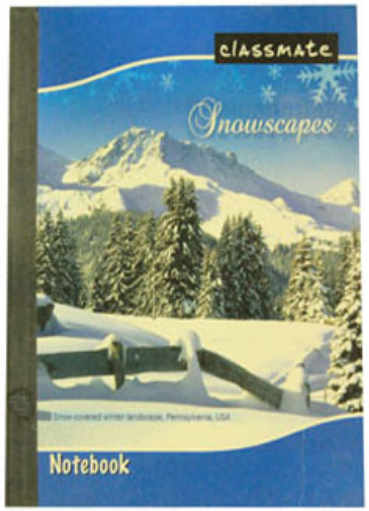
Lets understand how data types were modeled....

Size of data

Address of a student



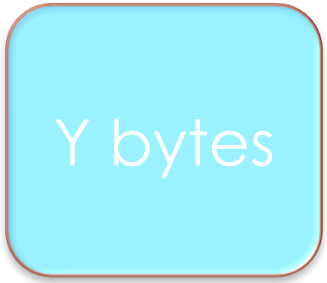
A particular question



Lecture Notes

CODING BLOCKS

Size of data



Address of a student

A particular question

Lecture Notes

X < Y < Z

CODING BLOCKS

Type of data

Remember Natural numbers, whole numbers, integers …

Whole numbers

Integers

Decimal point numbers

CODING BLOCKS

Defining variables

I. int roll_number;

roll_number

II. float marks;

marks

CODING BLOCKS

Defining variables

I. int roll_number;

roll_number

II. float marks;

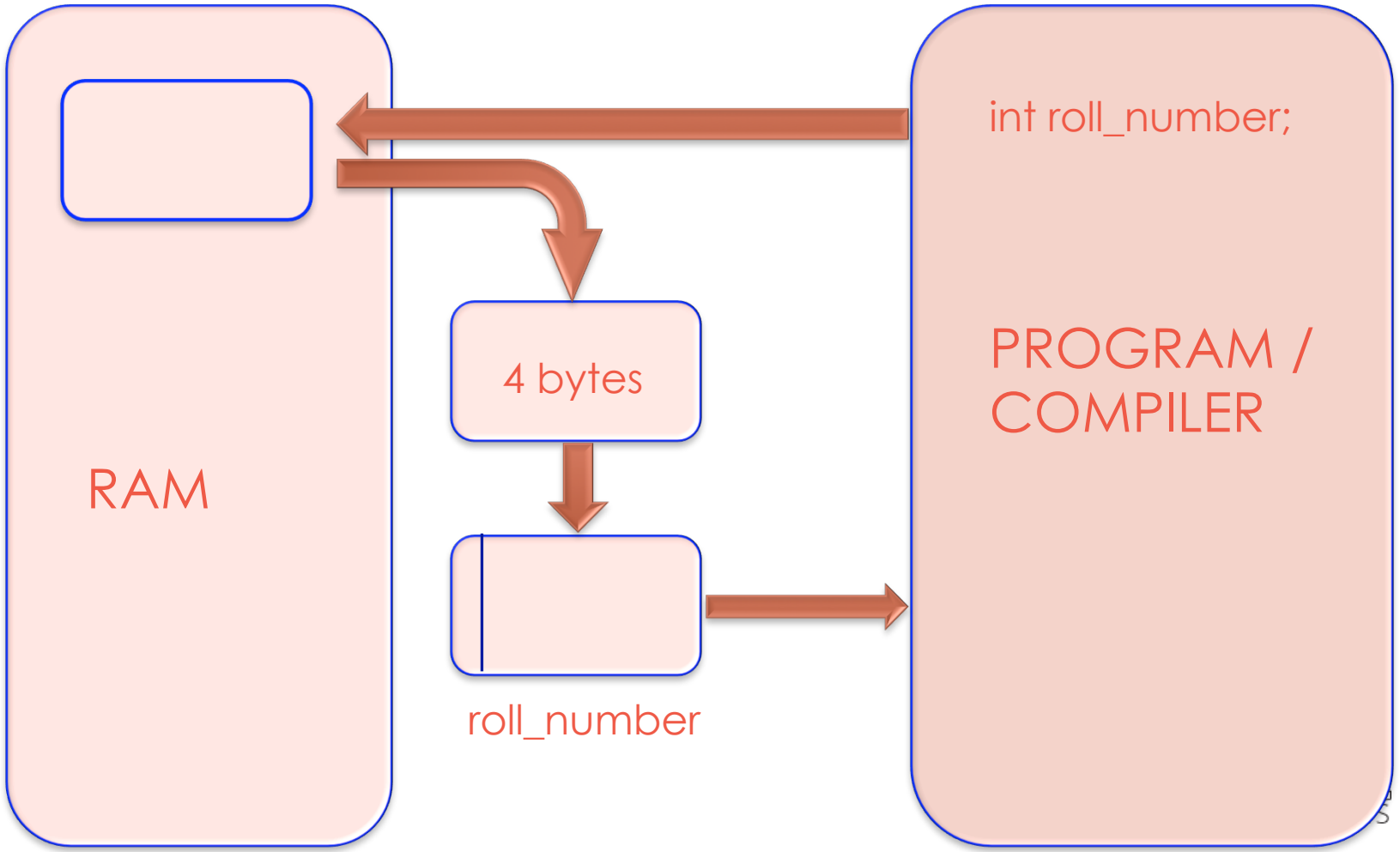
marks

Caution:

This memory chunk might not be empty. Do assign some value before using it.

CODING BLOCKS

Whole process of memory allocation:



R E C A P

Expressing Algorithms?

I. Algorithms can be expressed in many kind of notations, including natural languages, pseudocode, flowcharts, etc.

II. Natural Language expressions of algorithms tend to be verbose and ambiguous, and are rarely used for complex or technical algorithms.

III. Programming languages are primarily intended for expressing algorithms in a form that can be executed by a computer.

Two basic aspects of programming

I. Data
II. Instructions

To understand data we need to understand **Variables**!

What are Variables?

I. Variables in a computer program are analogous to Buckets or Envelopes where information can be maintained and referenced.
II. On the outside of the bucket is a name.
III. When referring to the bucket, we use the name of the bucket, not the data stored in the bucket.

Variable Actions!

I. Create one (with a nice name). A variable should be named to represent all possible values that it might contain. Some examples are: midterm_score, midterm_scores, data_points, course_name, etc.
II. Put some information into it (destroying whatever was there before). We "put" information into a variable using the assignment operator, e.g., midterm_score = 93;
III. Get a copy of the information out of it (leaving a copy inside). We "get" the information out by simply writing the name of the variable, the computer does the rest for us, e.g., average = (grade_1 + grade_2) / 2.

Now lets talk about Instructions!

What are instructions?

I. Its an order given to computer.
II. At lowest level each of these is a sequence of 0s and 1s.
III. In assembly language, each statement is one instruction but in high level each statement can be further broken into multiple steps.

Six basic computer instructions

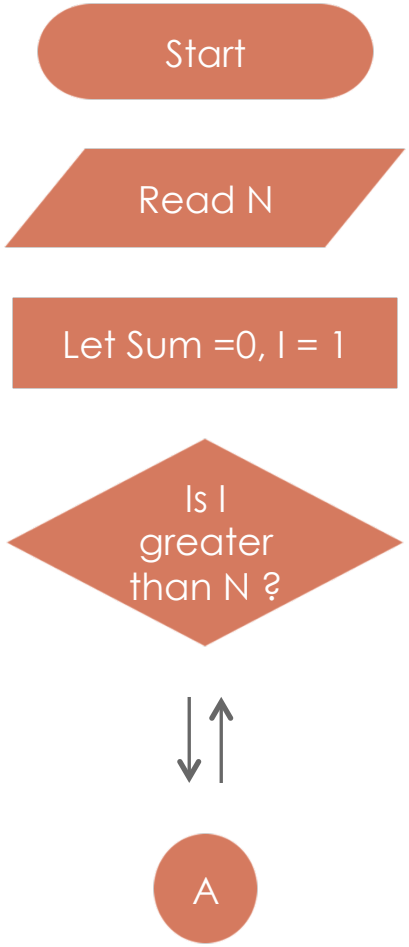
I. Reading/Receiving some information
II. Outputting/Printing some information
III. Performing arithmetic operation
IV. Assigning a value to a variable or memory location
V. Conditional Execution
VI. Repeat a group of actions

Time for Flowcharts!

What is a flowchart?

I. Diagrammatic representation illustrating a solution to a given problem.
II. Allows you to break down any process into smaller steps and display them in a visually pleasing way
III. It shows steps as boxes of various kinds, and their order by connecting them with arrows.
IV. It helps your audience to see the logical flow and relationship between steps.

Flowchart components



Initializer / Terminator

Input / Output

Process

Decision

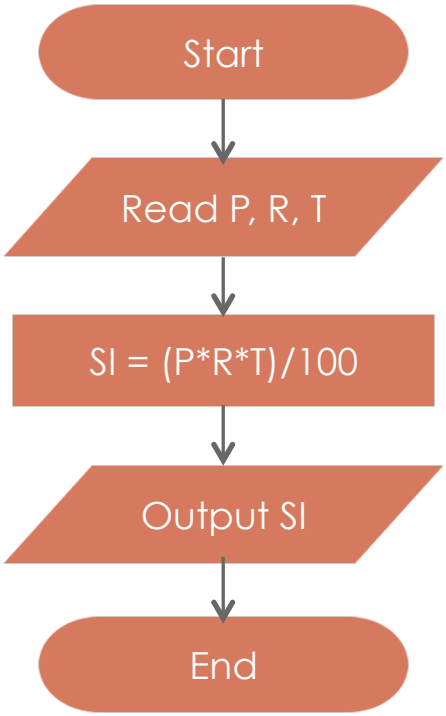
Arrow

Connector

CODING BLOCKS

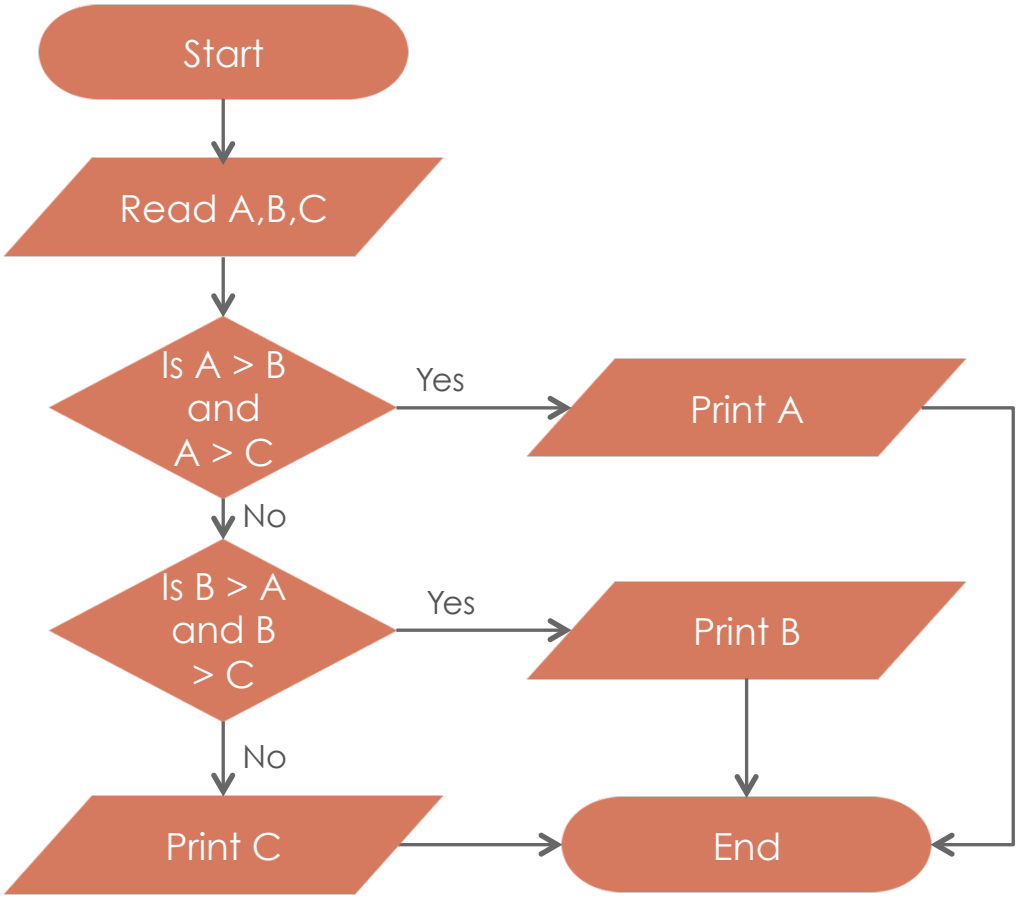
Lets look at few problems and their flowcharts!

Read Principal, Rate & Time and Print SI



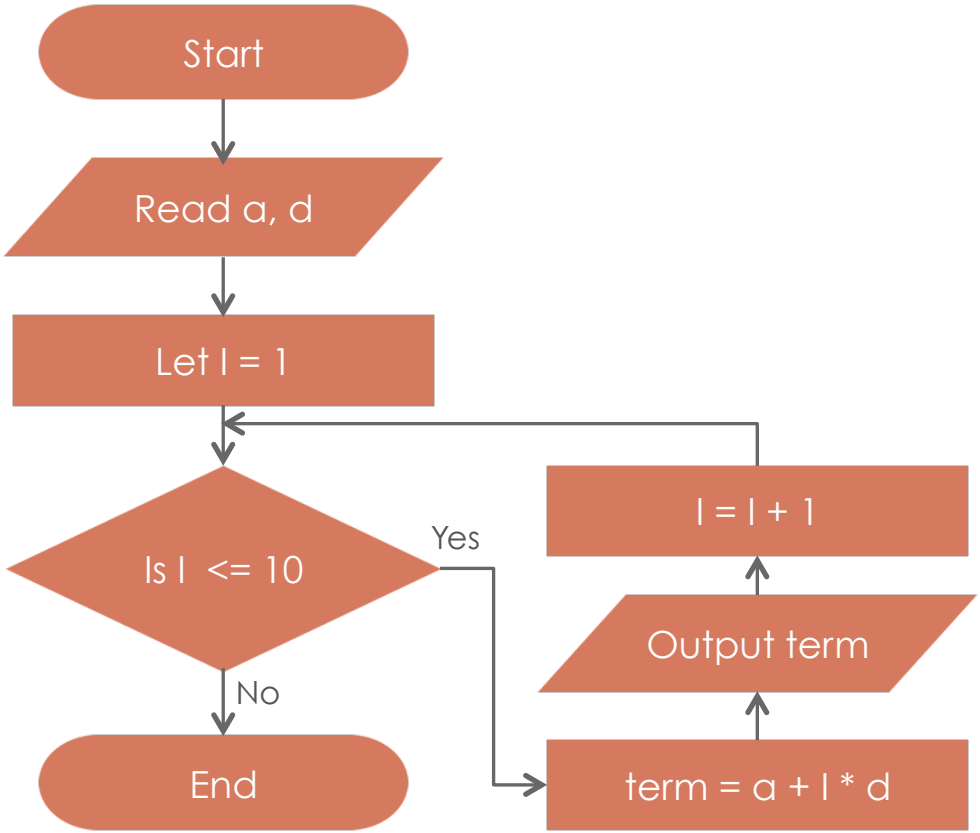
CODING BLOCKS

Find largest of three numbers



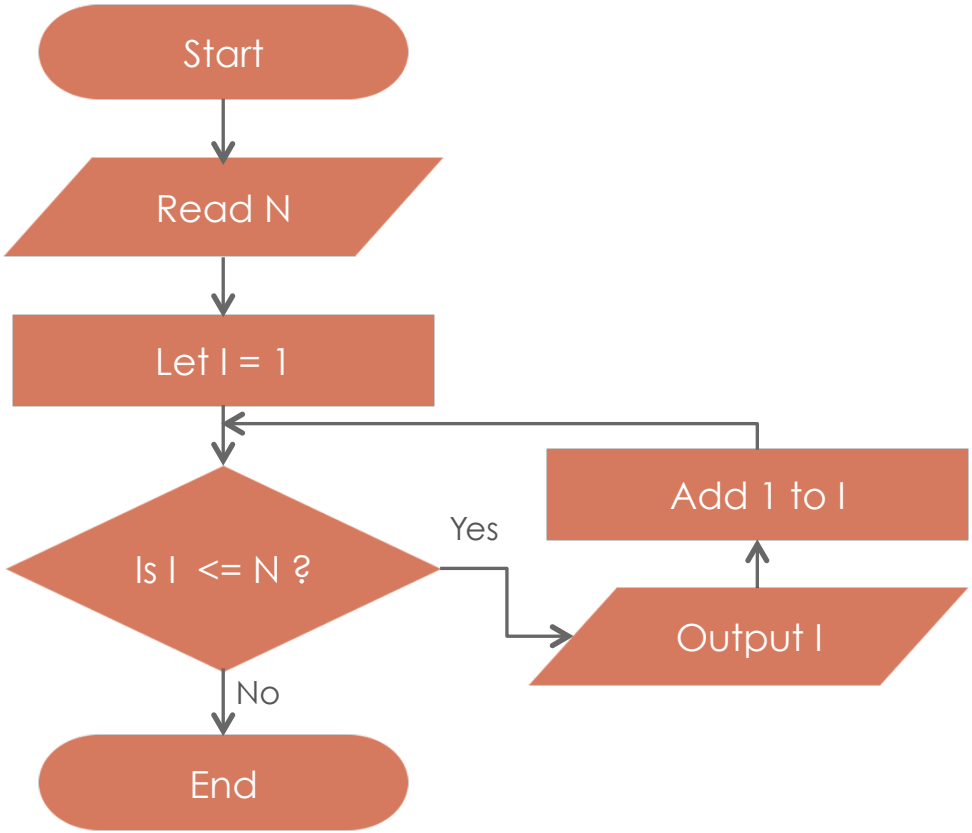
CODING BLOCKS

Read a & d, print 10 numbers of form a+d, a+2d, a+3d...



CODING BLOCKS

Given N, print all numbers from 1 to N

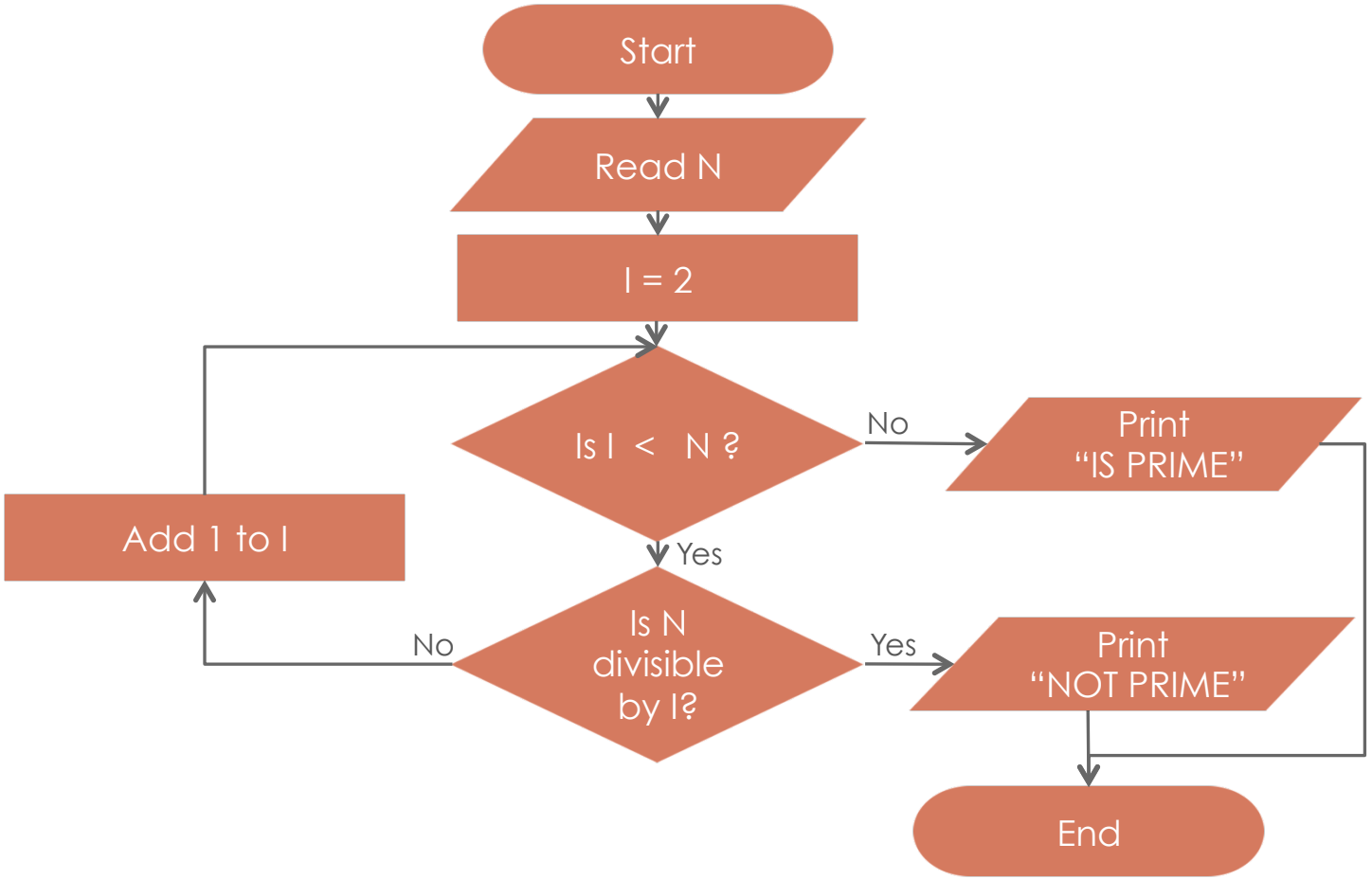


CODING BLOCKS

Time to try?

I. Read five numbers and print their average
II. Given N, find sum of even numbers from 1 to N
III. Given N, check if its prime or not

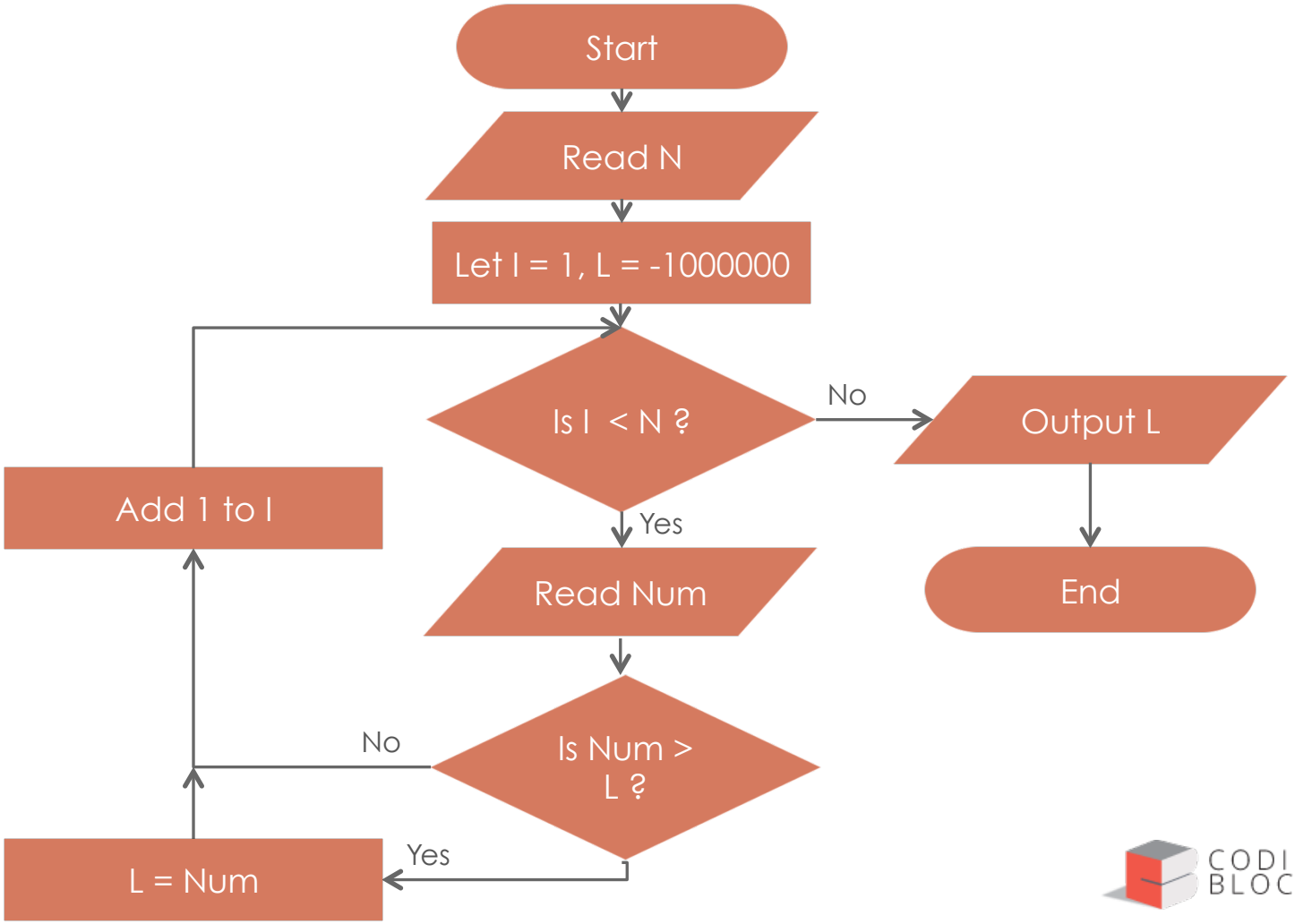
Check if a number is prime or not?



CODING BLOCKS

Some more examples!

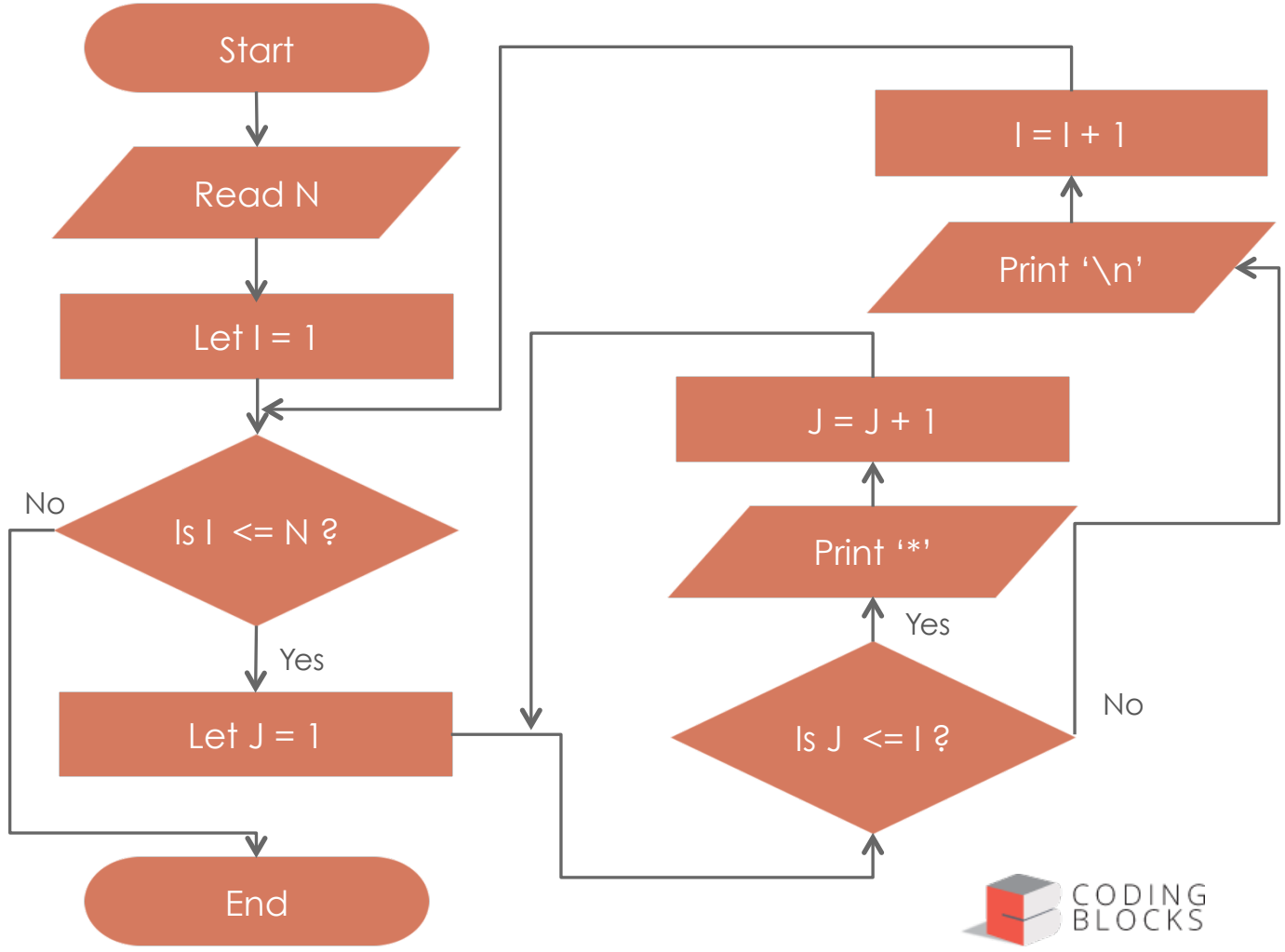
Find largest of N numbers



CODING BLOCKS

Print the below pattern

```
*
**
***
****
*****
```



Time to try?

I. Given a list of N integers, find mean, maximum and minimum value. You would be given first N, and then N integers of the list.

II. Given a number check if it is a member of Fibonacci sequence or not

III. Read N, and print the following pattern

```
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
```

Pseudocode!

What is pseudocode?

I. A notation resembling a simplified programming language, used in program design
II. It allows designers or lead programmers to express the design in great detail and provides programmers a detailed template for the next step of writing code in a specific programming language

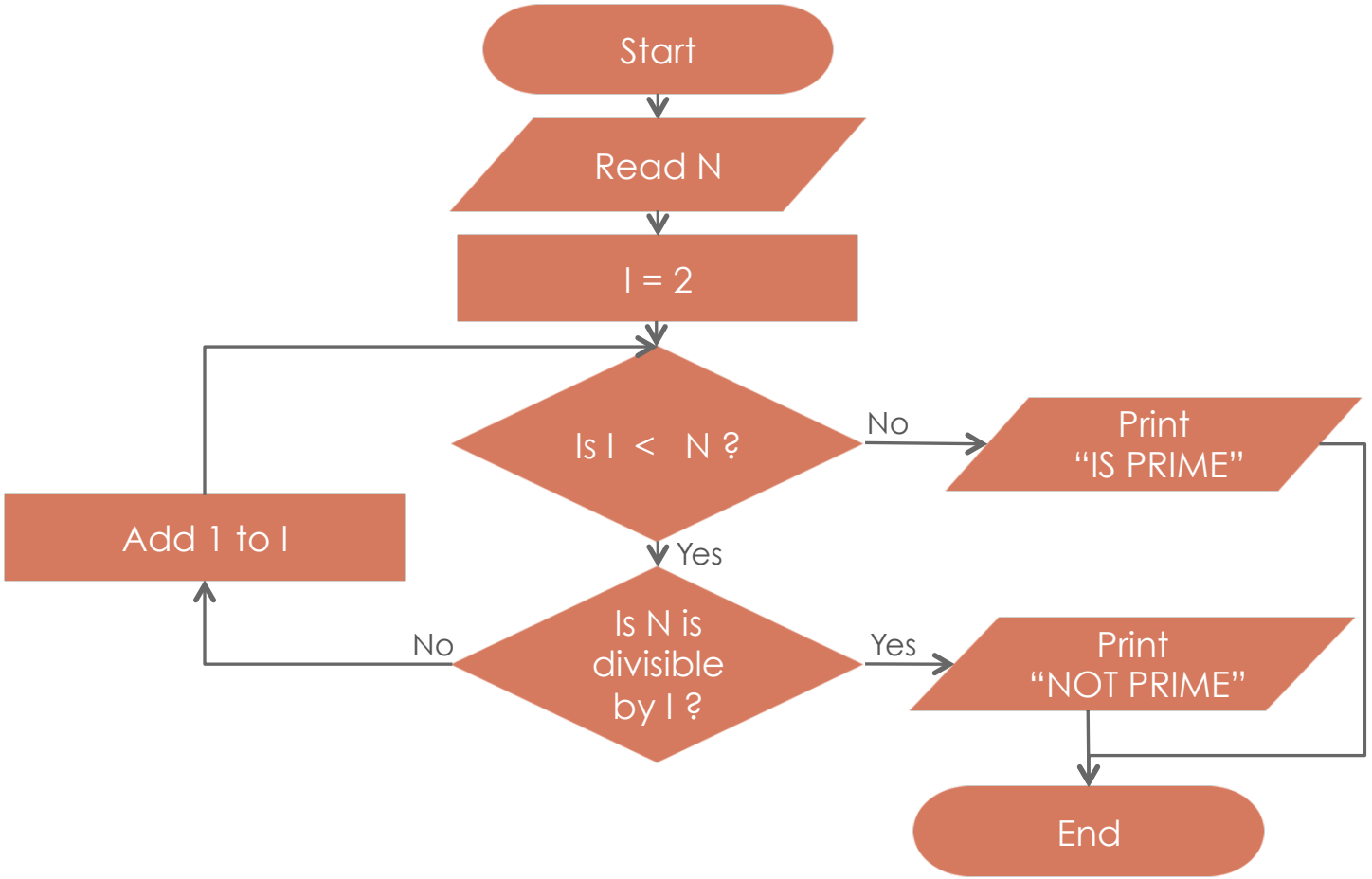
Notation for those basic six!

I. Reading/Receiving [read N]
II. Outputting/Printing [print Sum, print "Coding Blocks", print 1, print '\n']
III. Assignment [Sum ← 5]
IV. Arithmetic operators [a+b, a * 5, sum + i]
V. If Else [if I < N then … end else then … end]
VI. While Loop [while I < N do … end]

CODING BLOCKS

Lets convert some flowcharts into pseudocode!

Check if a number is prime or not?

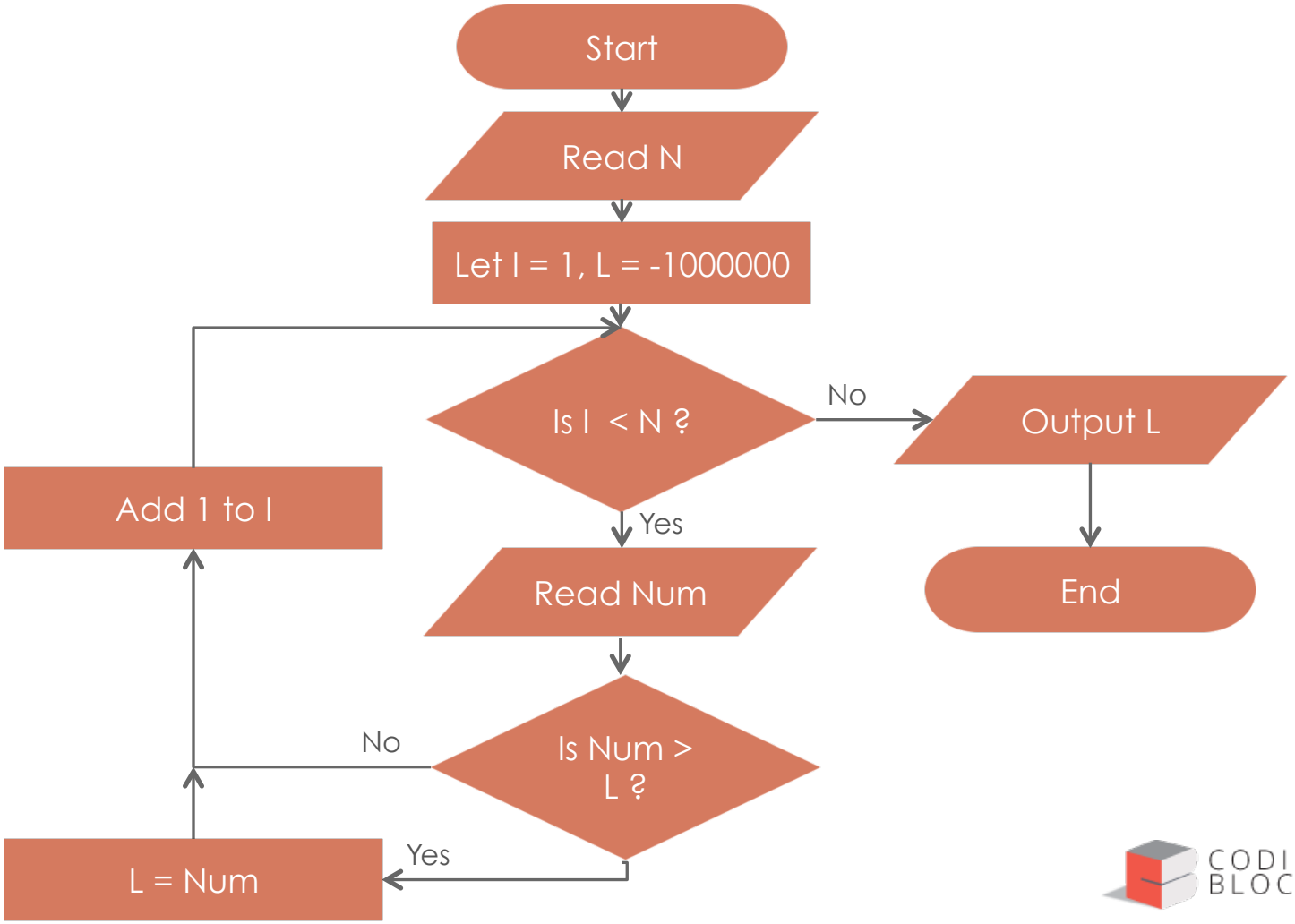


CODING BLOCKS

Pseudocode- Check if N is prime?

```
read N
I ← 2
While I < N do
        if N is divisible by I then
                print "NOT PRIME"
                exit
        end
I ← I + 1
end
print "IS PRIME"
exit
```

Find largest of N numbers



CODING BLOCKS

Pseudocode –Largest of N numbers

```
read N
I ← 1, L ← -100000
while I < N do
        read num
        if num > L then
                L ← num
        endif
        I ← I + 1
end
print L
exit
```

Print following pattern

```
1
2 3
4 5 6
7 8 9 10
```

```
read N
row ← 0
value ← 1
While row < N do
        col ← 0
        while col <= row do
                print value
                value ← value + 1
                col ← col + 1
        end
        print "\n"
        row ← row + 1
end
exit
```

Lets try one more pattern!

```
    1
   123
  12345
 1234567
123456789
```

```
read N
i ← 0
while i < N do
        j ← 0
        while j < n-i-1 do
                print ' '
                j ← j + 1
        end
        j ← 0, value ← 1
        while j < 2*i + 1 do
                print value
                value ← value + 1
                j ← j + 1
        end
        print '\n'
        i ← i + 1
end
```

Time to try?

I. Convert your flowcharts into pseudocode
II. Read N and print the below pattern

```
    1
   232
  34543
 4567654
567898765
```

III. Given a number N, find sum of its digits

Time for Brain Teasers!

BT - 1: Hour Glasses

You have two hourglasses: a 7 minute one and a 11 minute one. Using just two hourglass, **accurately time 15 minutes**.

BT – 2: Apples and Oranges

There are three closed and opaque cardboard boxes. One is labeled "APPLES", another is labeled "ORANGES", and the last is labeled "APPLES AND ORANGES". You know that the labels are currently misarranged, such that no box is correctly labeled. You would like to correctly rearrange these labels. To accomplish this, you may draw only one fruit from one of the boxes. **Which box do you choose, and how do you then proceed to rearrange the labels?**

BT – 3: Average Salary

Three coworkers would like to know their average salary. However, they are self-conscious and don't want to tell each other their own salaries, for fear of either being ridiculed or getting their houses robbed. **How can they find their average salary, without disclosing their own salaries?**

BT – 4: Criminal Cupbearers

An evil king has 1000 bottles of wine. A neighboring queen plots to kill the bad king, and sends a servant to poison the wine. The king's guards catch the servant after he has only poisoned one bottle. The guards don't know which bottle was poisoned, but they do know that the poison is so potent that even if it was diluted 1,000,000 times, it would still be fatal. Furthermore, the effects of the poison take one month to surface. The king decides he will get some of his prisoners in his vast dungeons to drink the wine. **Rather than using 1000 prisoners each assigned to a particular bottle, this king knows that he needs to murder no more than 10 prisoners to figure out what bottle is poisoned, and will still be able to drink the rest of the wine in 5 weeks time. How does he pull this off?**

What is next class about?

Programming Fundamentals - 1

I. Basic syntax of C++
II. Datatypes/Variables
III. Constants
IV. If else
V. while

CODING BLOCKS

Thank you

Aman Bahl
09908124628