Problem Solving Introduction 🚀

Introduction

- Basics of Problem Solving
- Flowcharts
- Psuedocode

About Me

Prateek Narang

I ❤️ teaching code and

...

Co-founded Coding Blocks

Co-founded Coding Minutes

Udemy Instructor

Ex-Google Engineer

Engineering Lead @ Scaler & Instructor

Hour Glass Puzzle

An eccentric professor used a unique way to measure time for a test lasting 15 minutes.
He used just two hourglasses. One measured 7 minutes and the other 11 minutes.

Flowcharts

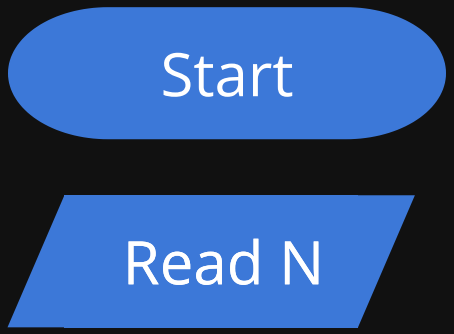
- Diagrammatic representation illustrating a solution to a given problem.

- Allows you to break down any process into smaller steps and display them in a visually appealling way.

Flowchart Components

Terminator

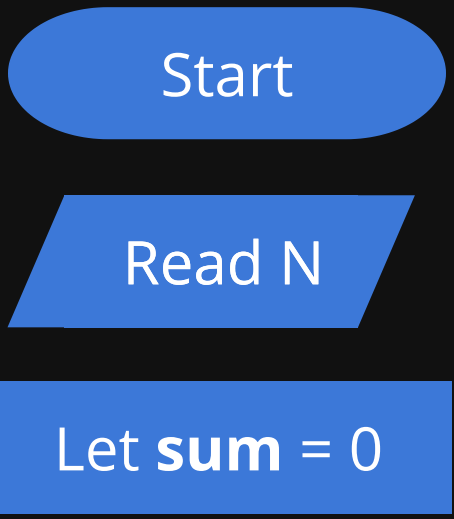
Flowchart Components



Terminator

Input/Output

Flowchart Components

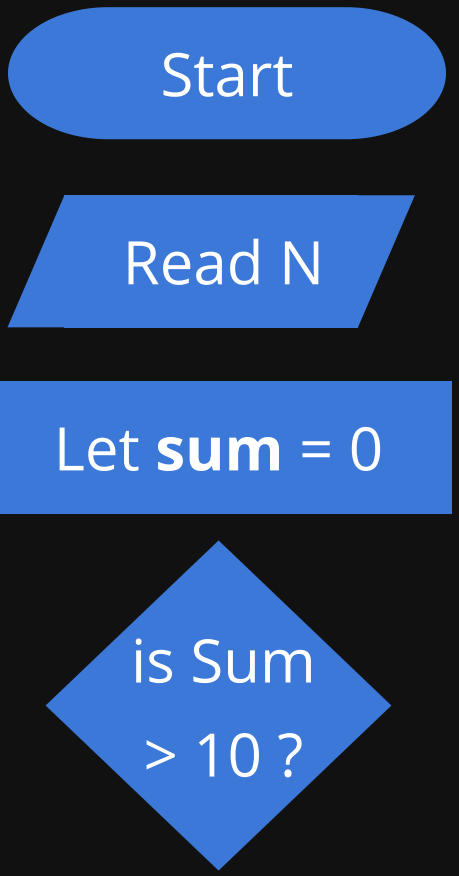


Terminator

Input/Output

Process

Flowchart Components



Terminator

Input/Output

Process

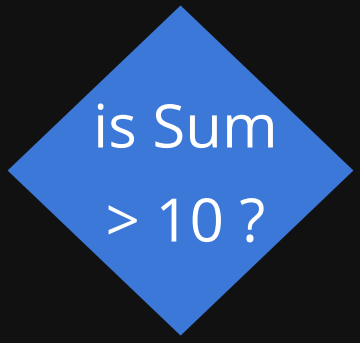
Decision

Flowchart Components

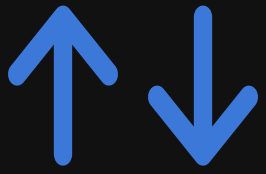
Terminator

Input/Output

Process



Decision



Arrows

Examples

Lets look at few problems and their flowcharts

Simple Interest Calculator

Read Principal, Rate & Time and print Simple Interest.

Sample Input

P = 100

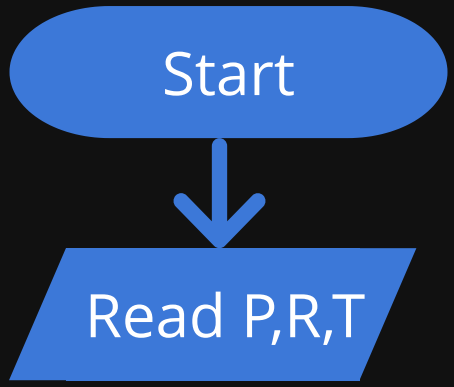
R = 5

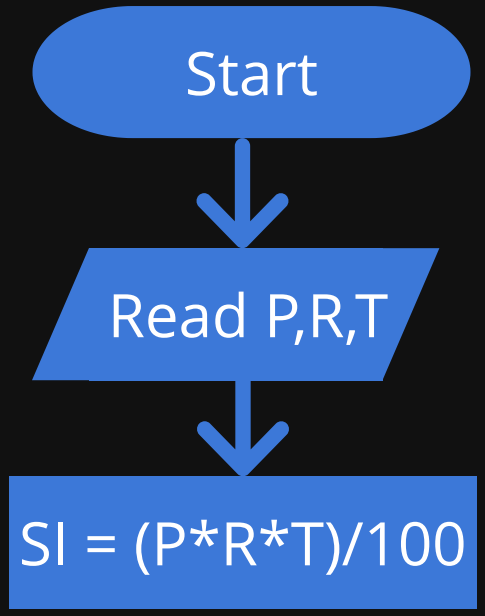
T = 2

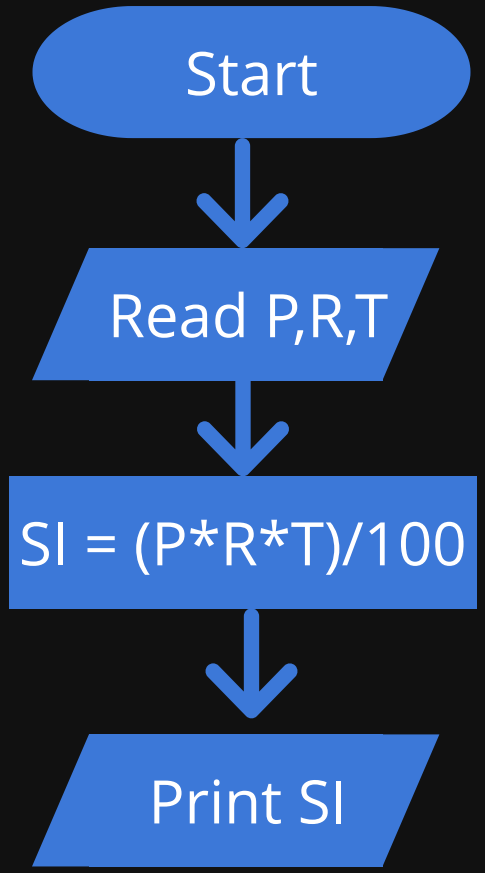
Sample Output

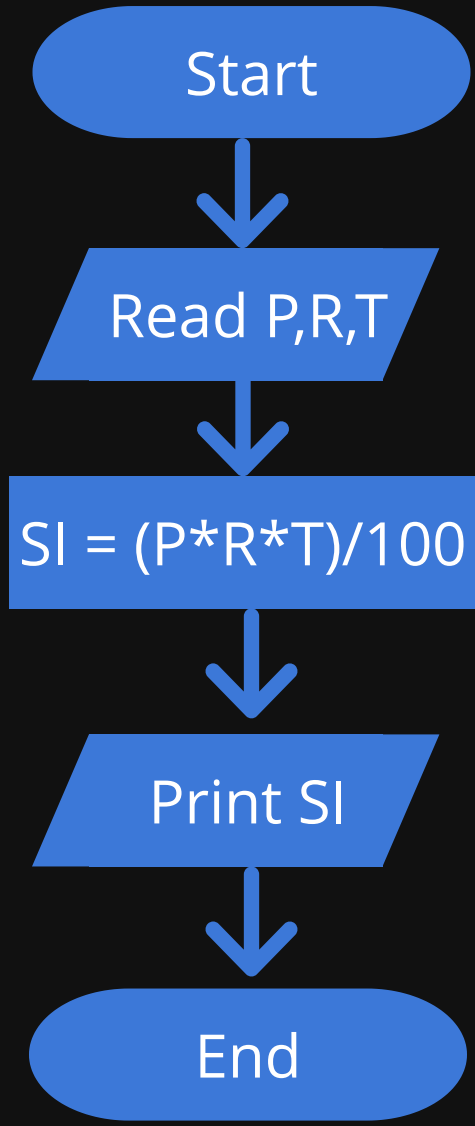
SI = 10

Start


Start
Read P,R,T


Start
Read P,R,T
SI = (P*R*T)/100


Start
Read P,R,T
SI = (P*R*T)/100
Print SI


Start
Read P,R,T
SI = (P*R*T)/100
Print SI
End

Largest Number

Given 3 numbers, find the largest number.

Sample Input

A = 10

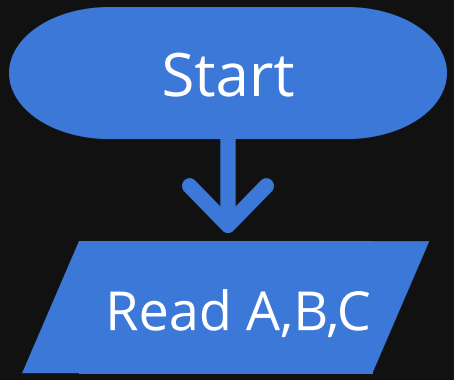
B = 30

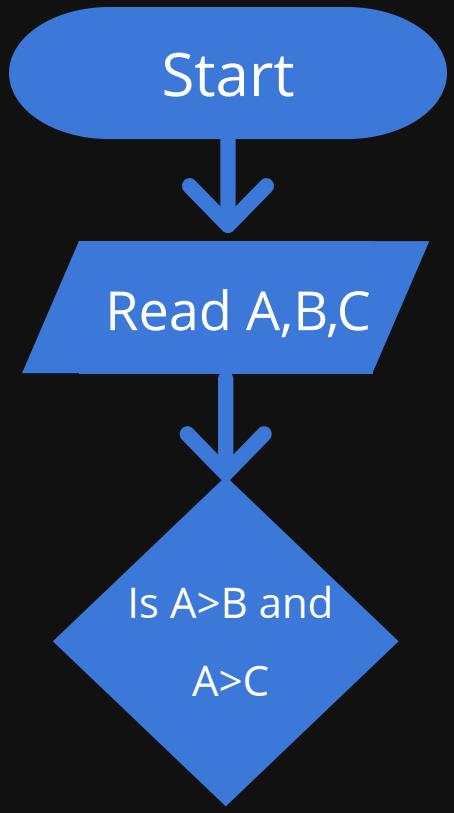
C = 20

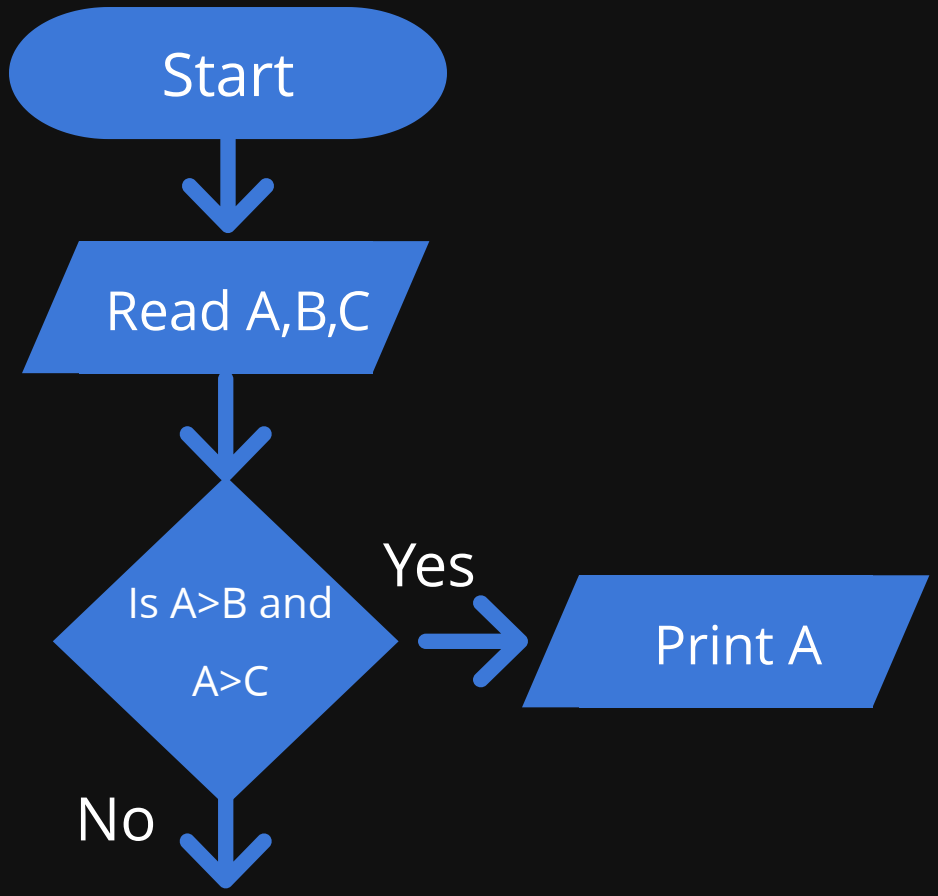
Sample Output

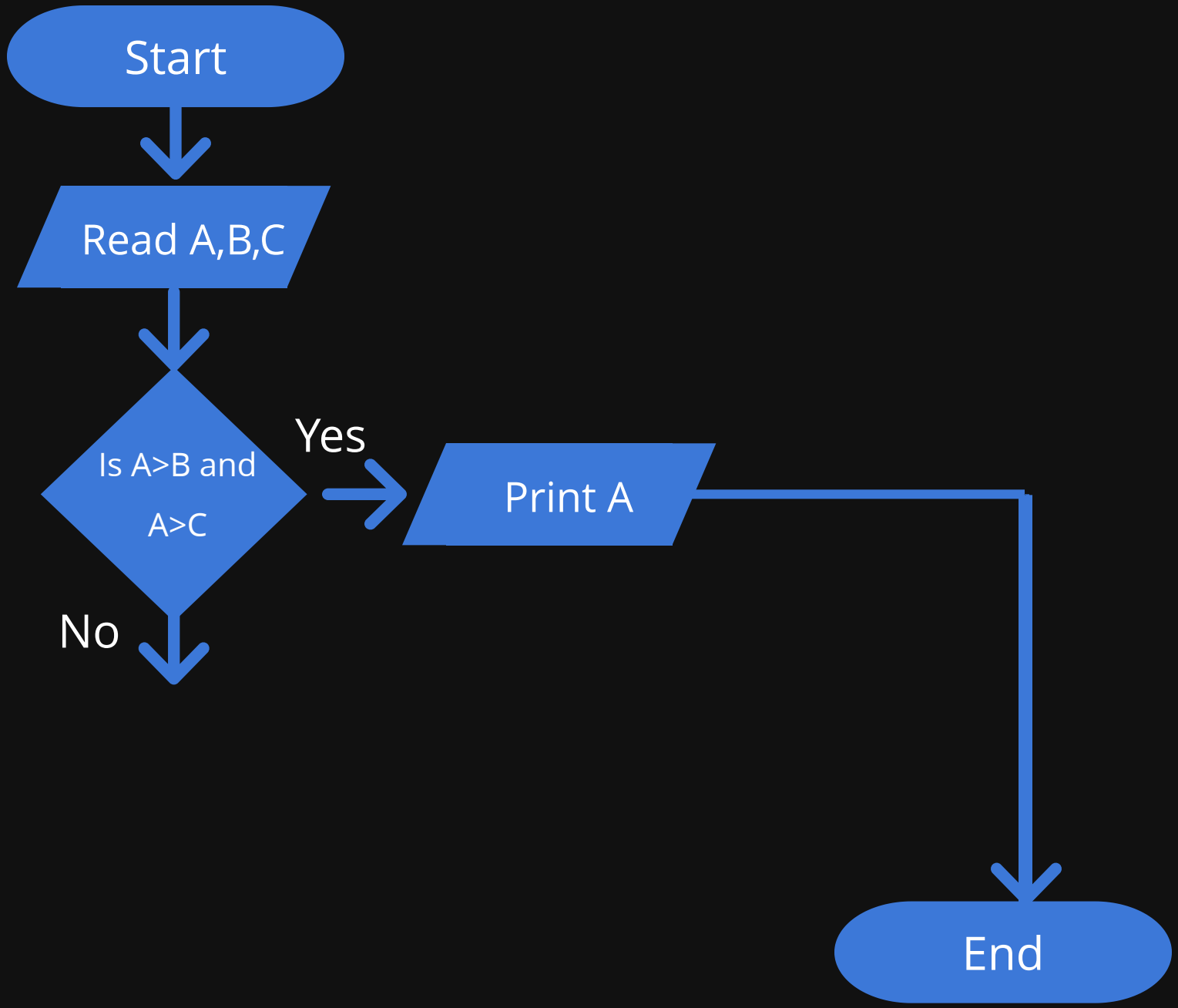
30

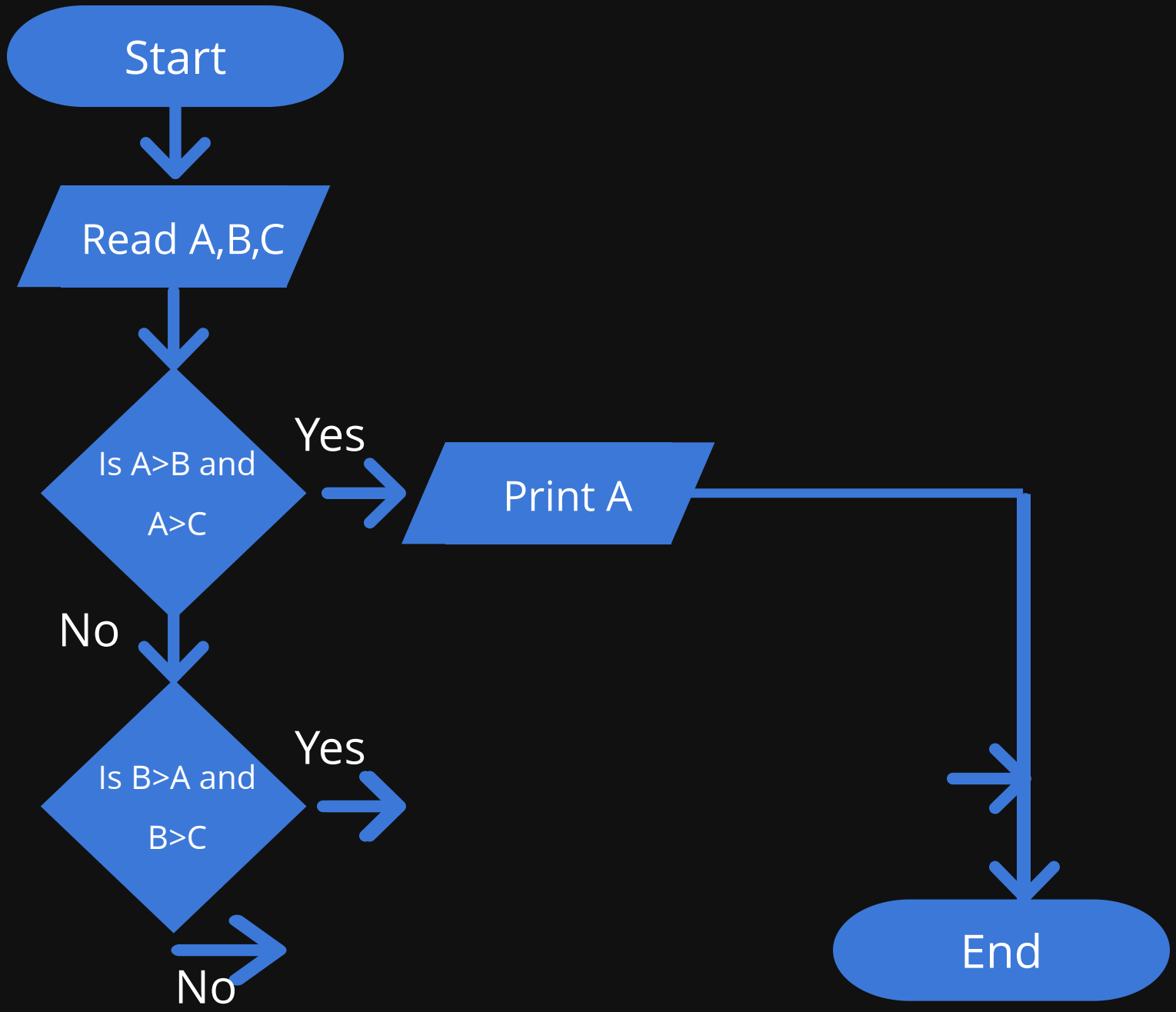
Start

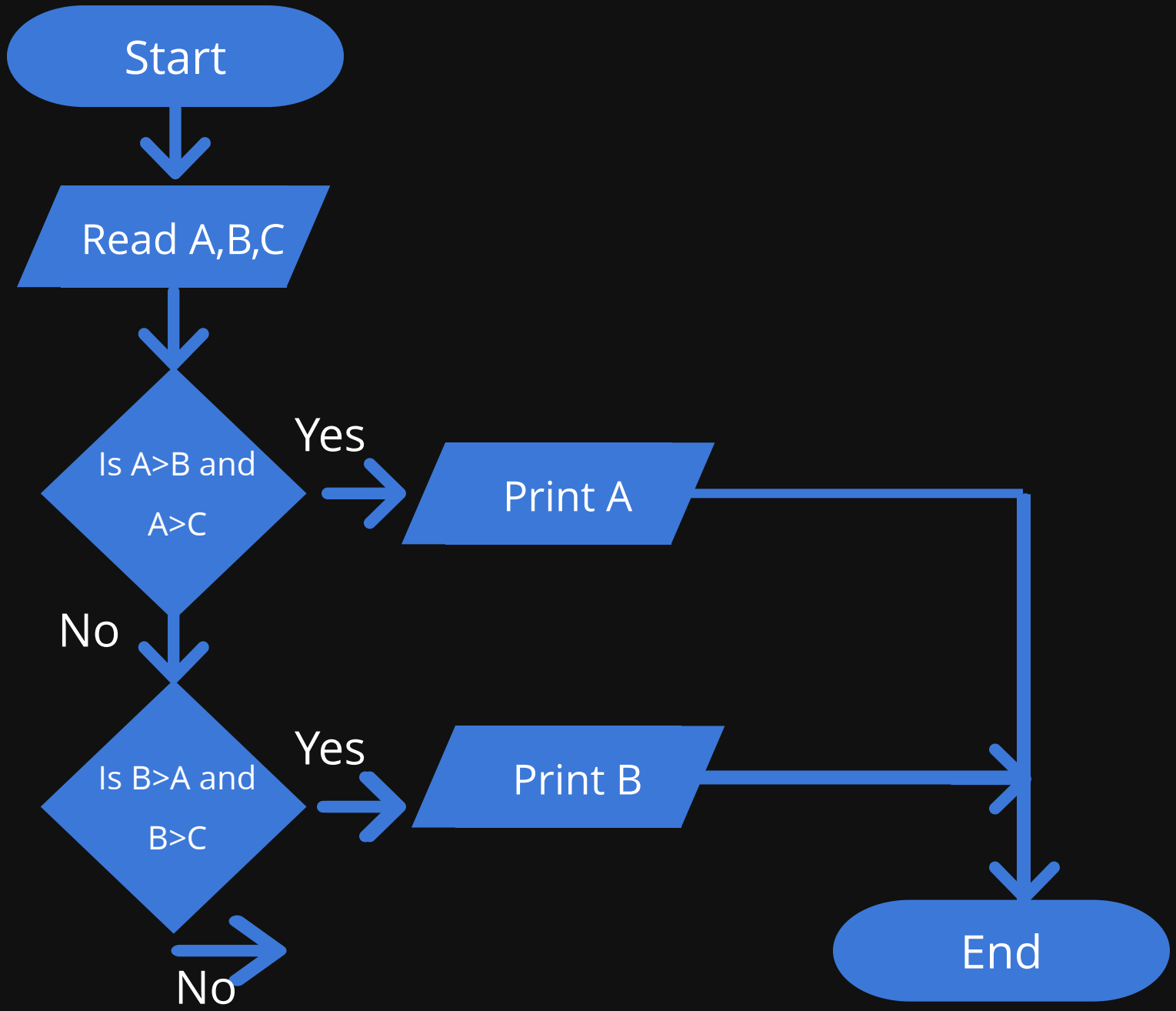

Start
Read A,B,C

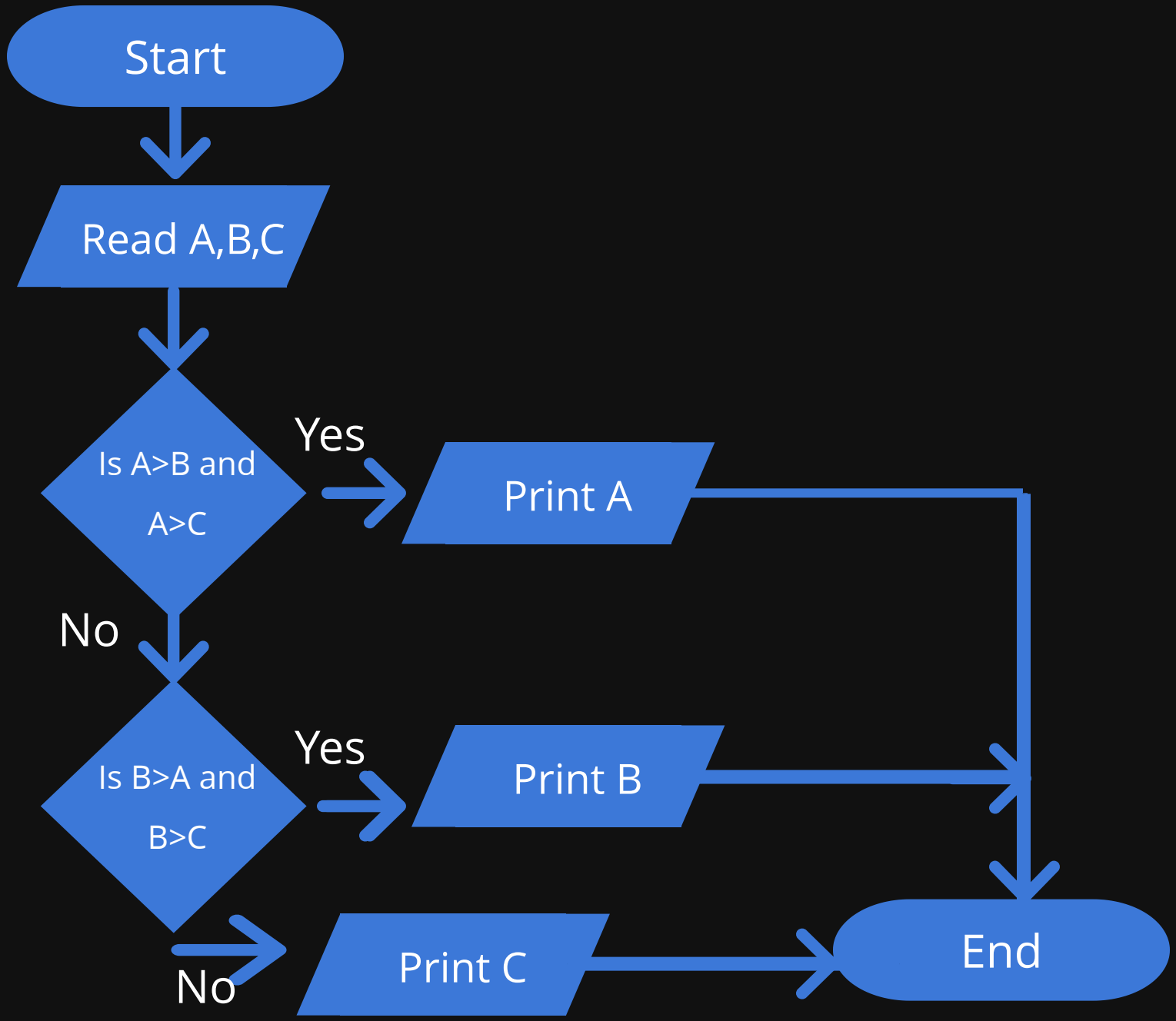

Start
Read A,B,C
Is A>B and
A>C


Start
Read A,B,C
Is A>B and A>C
Yes
Print A
No


Start
Read A,B,C
Is A>B and A>C
Yes
No
Print A
End


Start
Read A,B,C
Is A>B and A>C
Yes
Print A
No
Is B>A and B>C
Yes
No
End


Start
Read A,B,C
Is A>B and A>C
Yes
Print A
No
Is B>A and B>C
Yes
Print B
No
End


Start
Read A,B,C
Is A>B and A>C
Yes
Print A
No
Is B>A and B>C
Yes
Print B
No
Print C
End

Number Sum

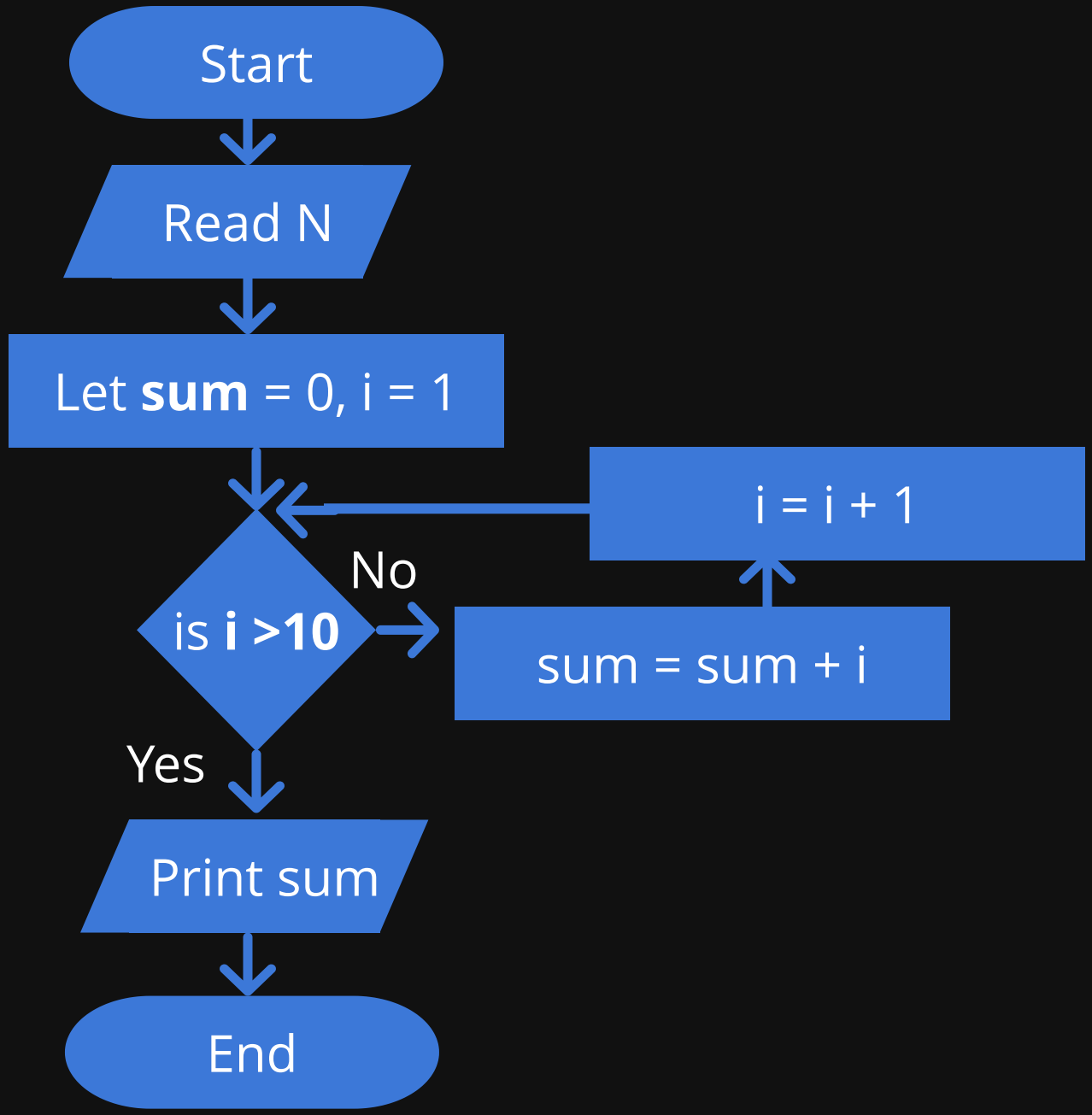
Find the sum of numbers from 1 to N

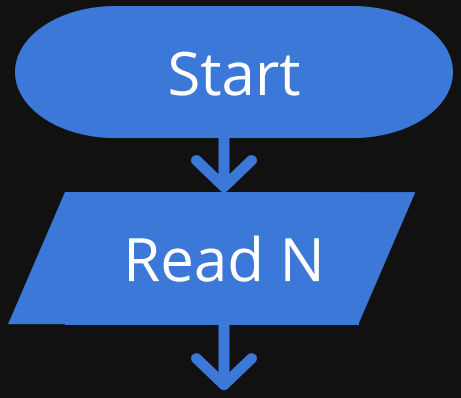
Example

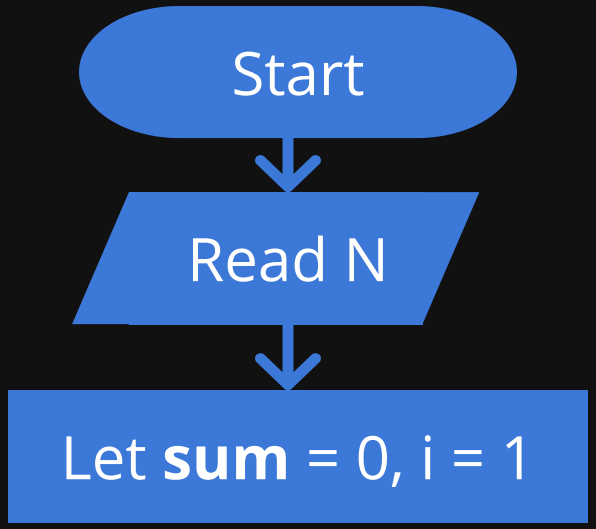
N = 4

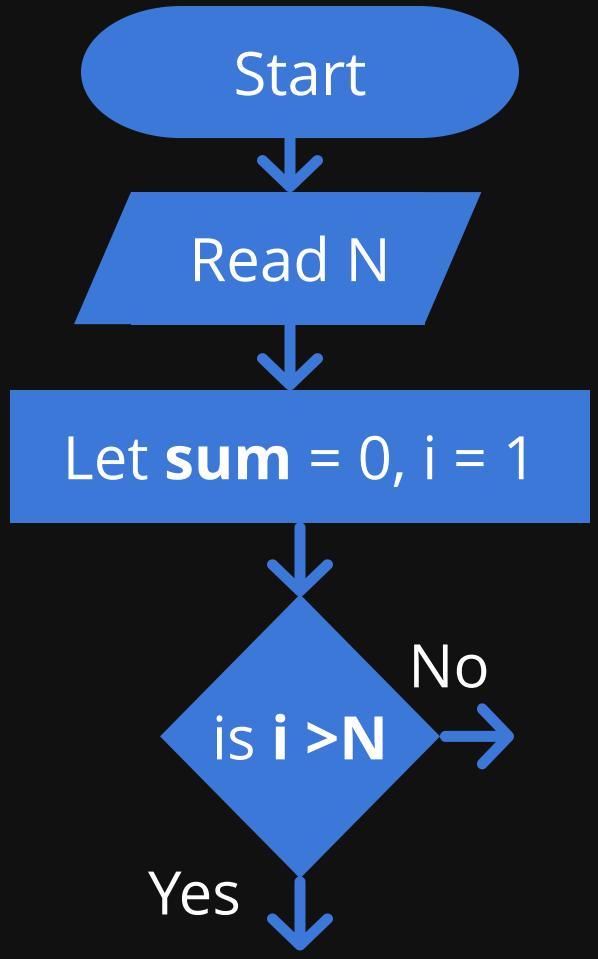
Output

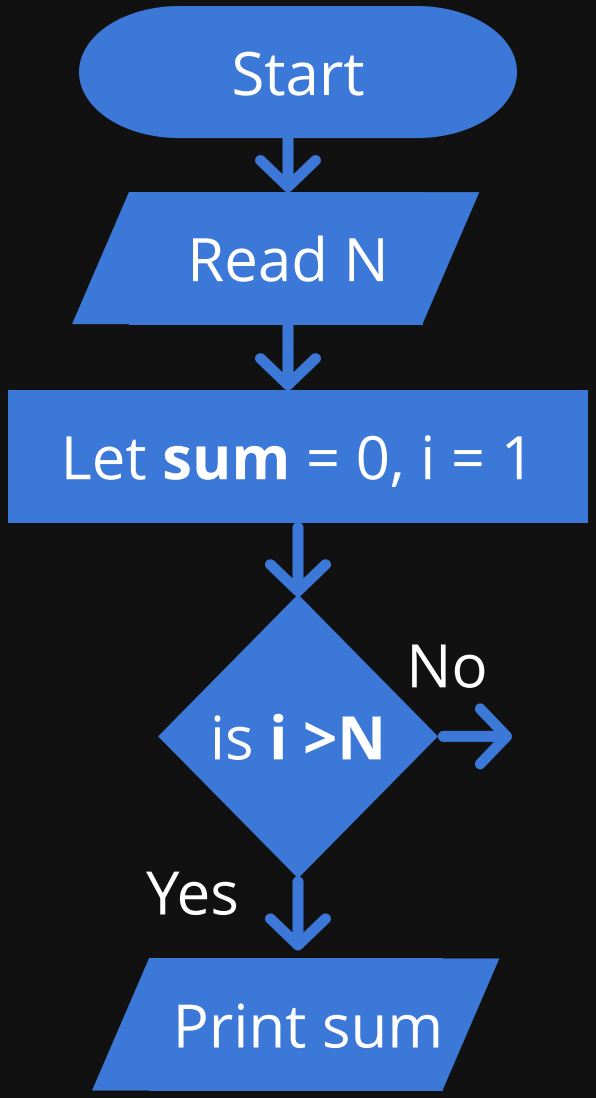
10

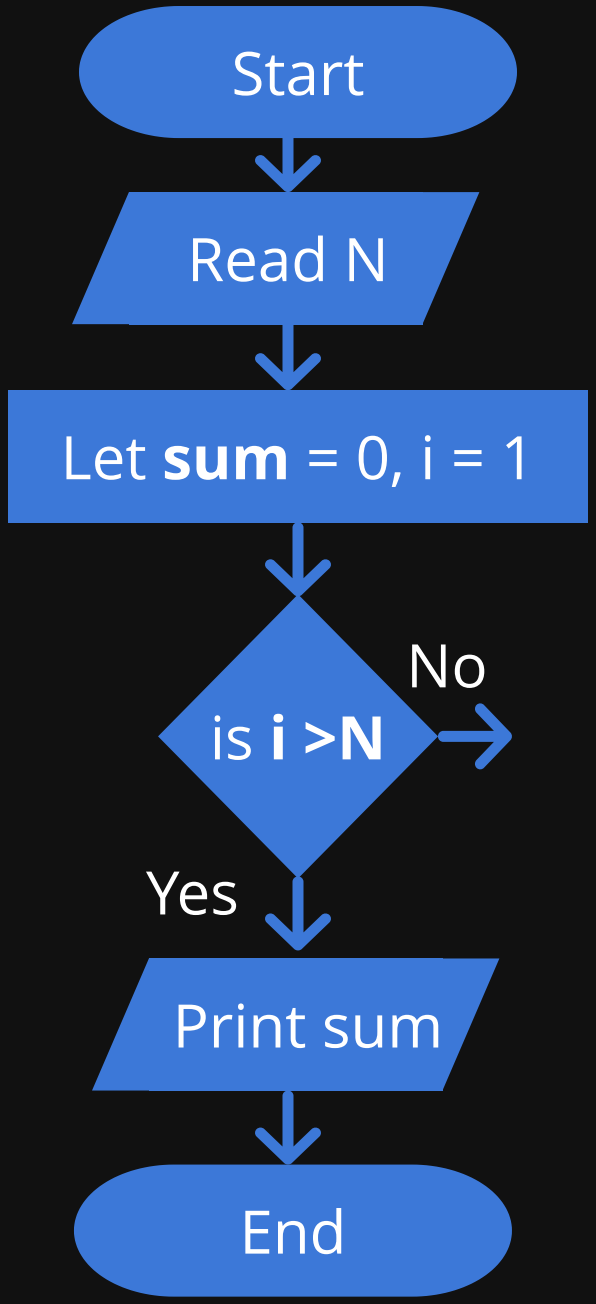

Start
Read N
Let **sum** = 0, i = 1
i = i + 1
No
is **i >10**
sum = sum + i
Yes
Print sum
End

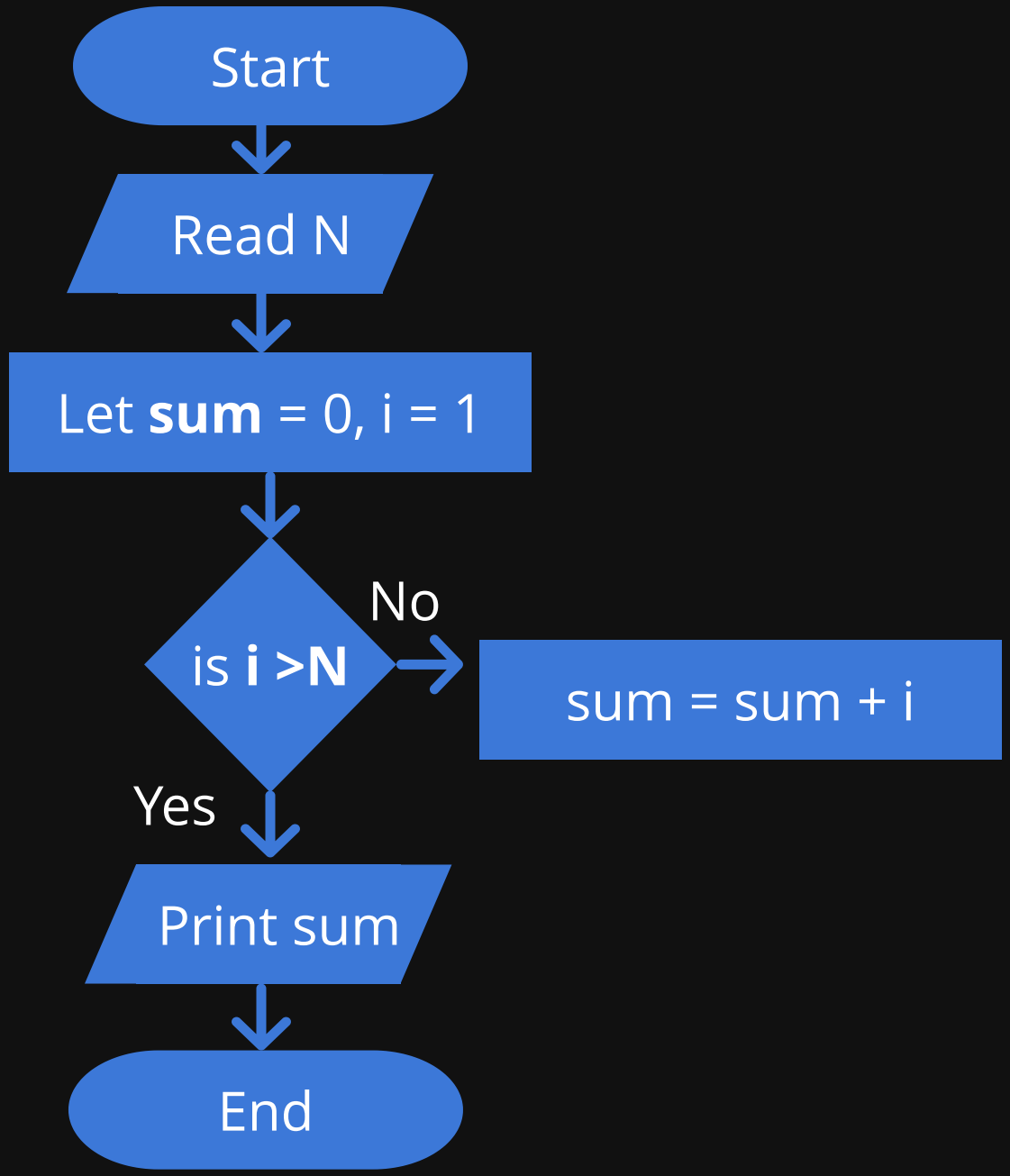

Start
Read N

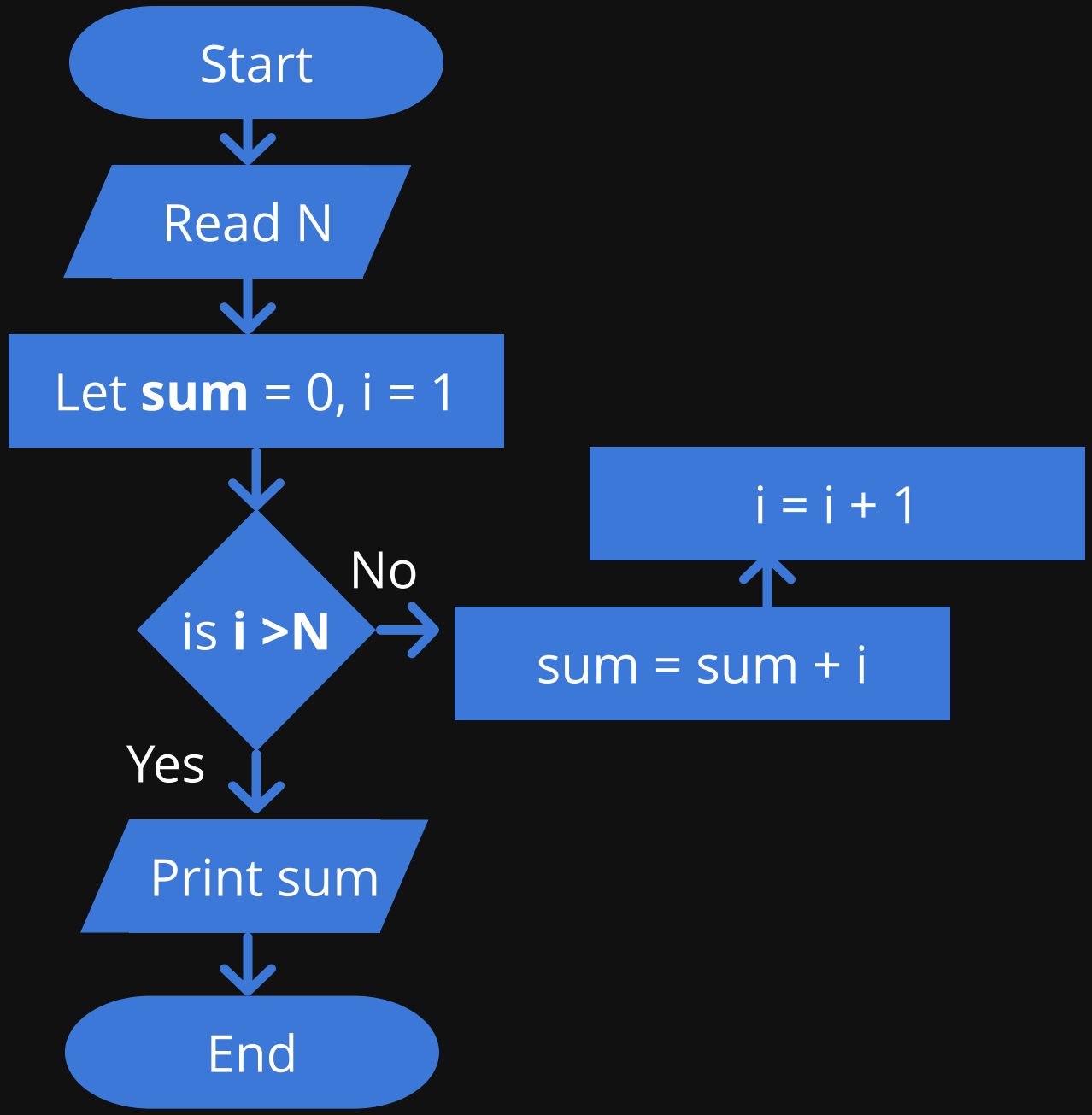

Start
Read N
Let **sum** = 0, i = 1

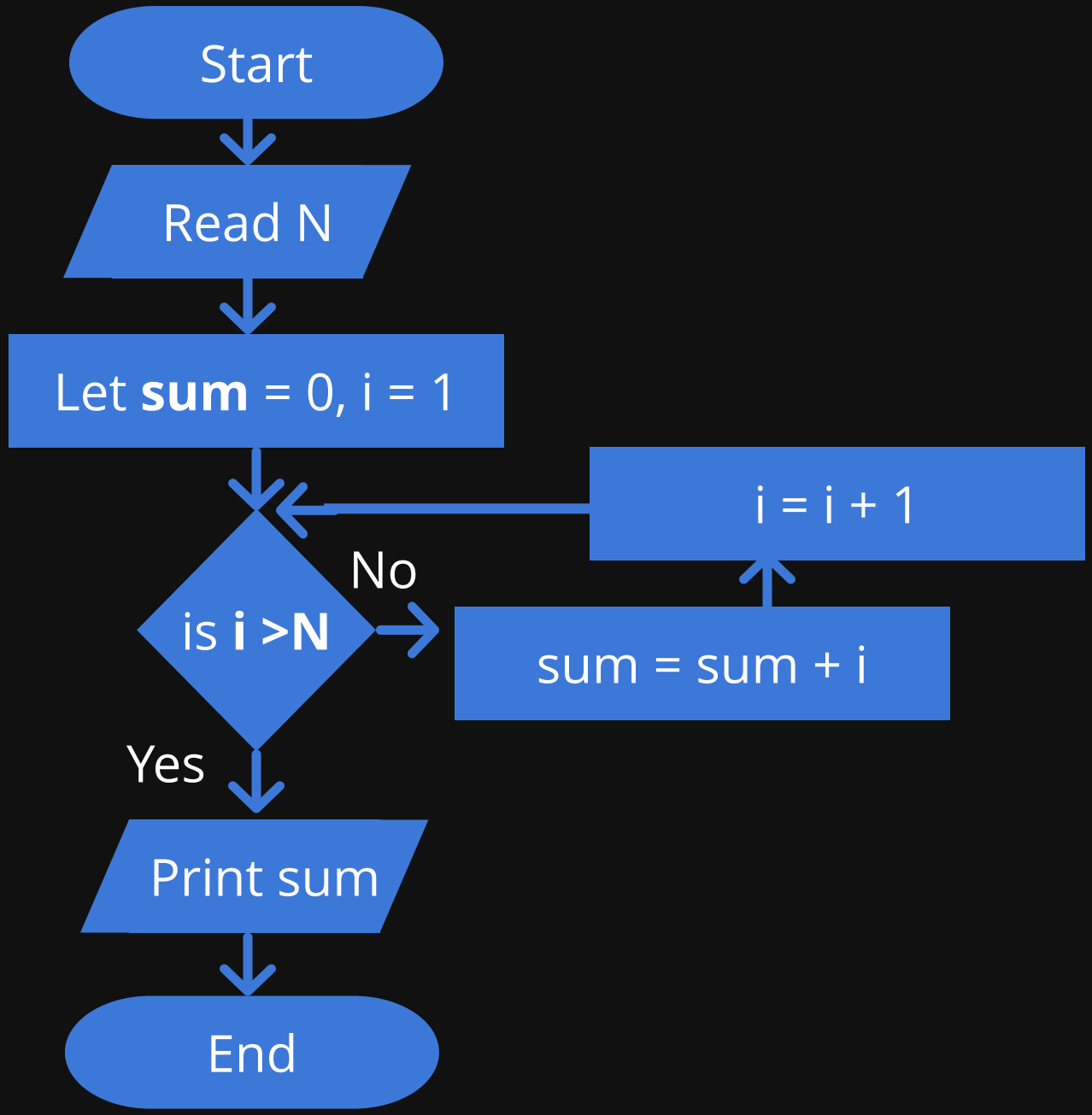

Start
Read N
Let **sum** = 0, i = 1
is **i >N**
No
Yes


Start
Read N
Let **sum** = 0, i = 1
is **i >N**
No
Yes
Print sum


Start
Read N
Let **sum** = 0, i = 1
is **i >N**
No
Yes
Print sum
End


Start
Read N
Let **sum** = 0, i = 1
is **i >N**
No
sum = sum + i
Yes
Print sum
End


Start
Read N
Let **sum** = 0, i = 1
is **i >N**
No
Yes
sum = sum + i
i = i + 1
Print sum
End


Start
Read N
Let **sum** = 0, i = 1
i = i + 1
No
is **i >N**
sum = sum + i
Yes
Print sum
End

Time To Try!

8 Mins

Prime Number

Given a Number, check if it is Prime or Not!

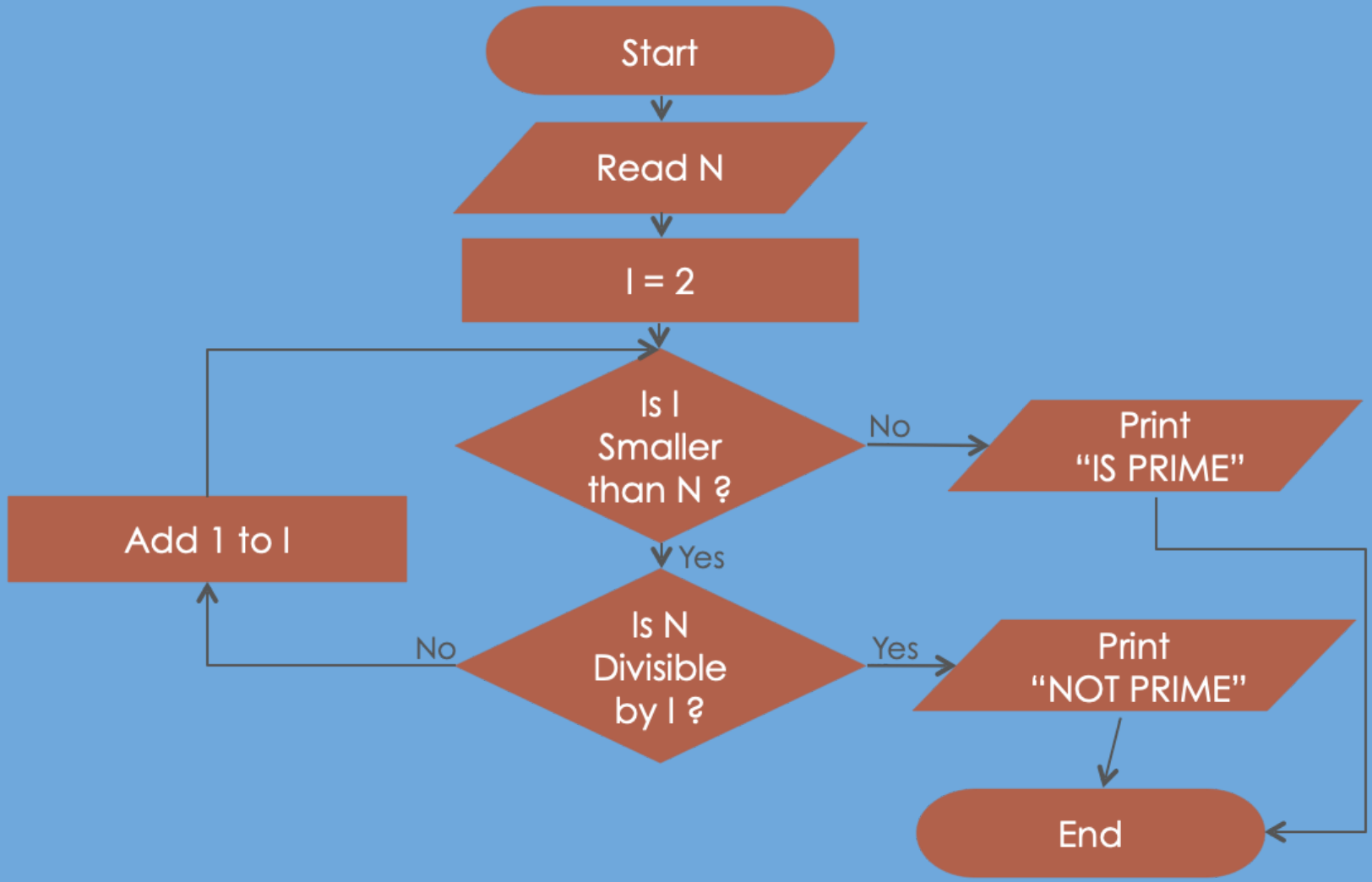
Input

11

Output

Yes

Let's Try


Start
Read N
I = 2
Is I Smaller than N ?
No
Print "IS PRIME"
Yes
Add 1 to I
Is N Divisible by I ?
No
Yes
Print "NOT PRIME"
End

A Real World Example!

Bank

A Bank is open till 6 p.m. and the banker needs to process requests from customers

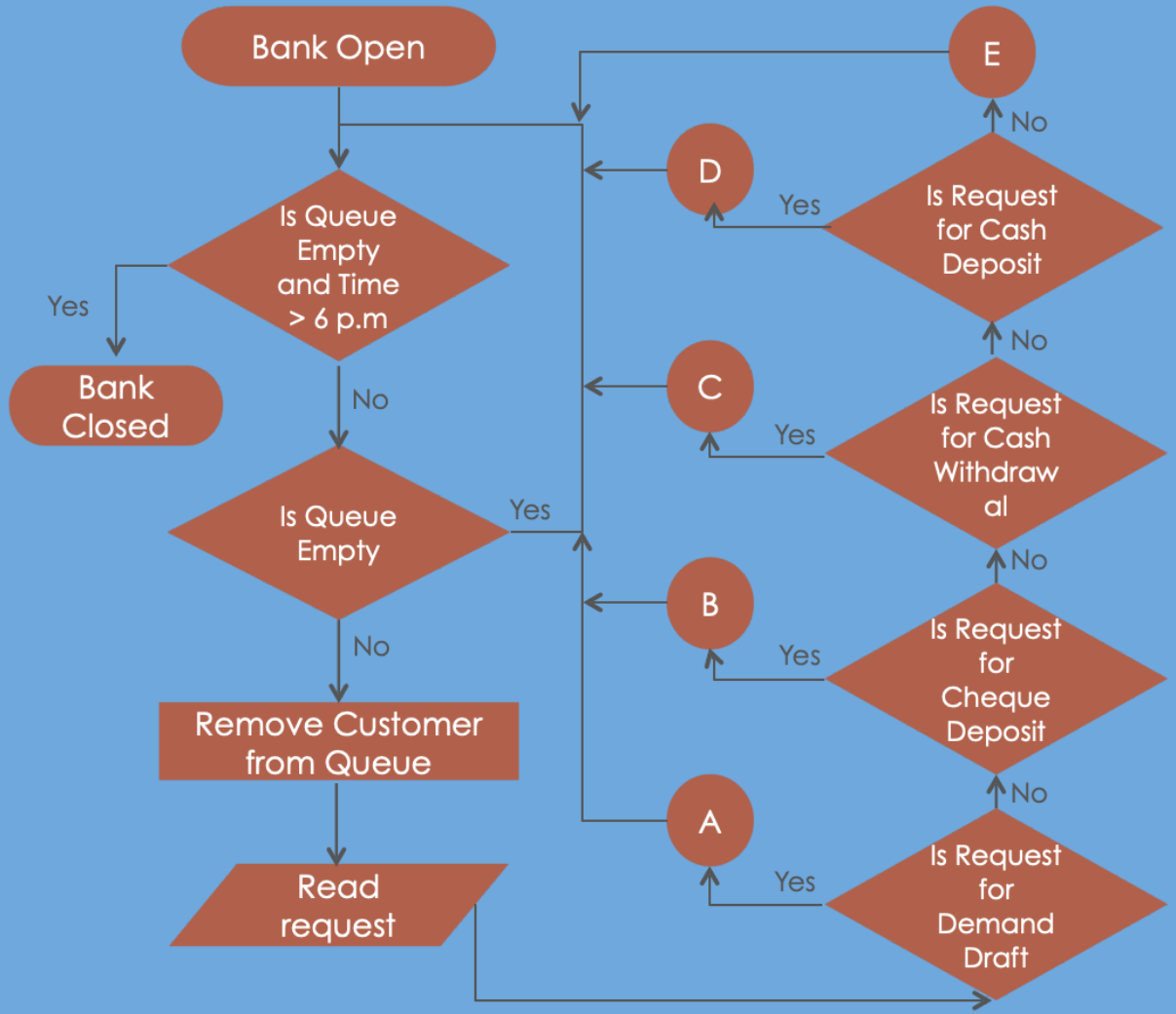
Requests can be one of the four types:

- Cash Deposit
- Cash Withdrawal
- Cheque Deposit
- Making A Demand Draft

Draw the workflow of the bank employee.

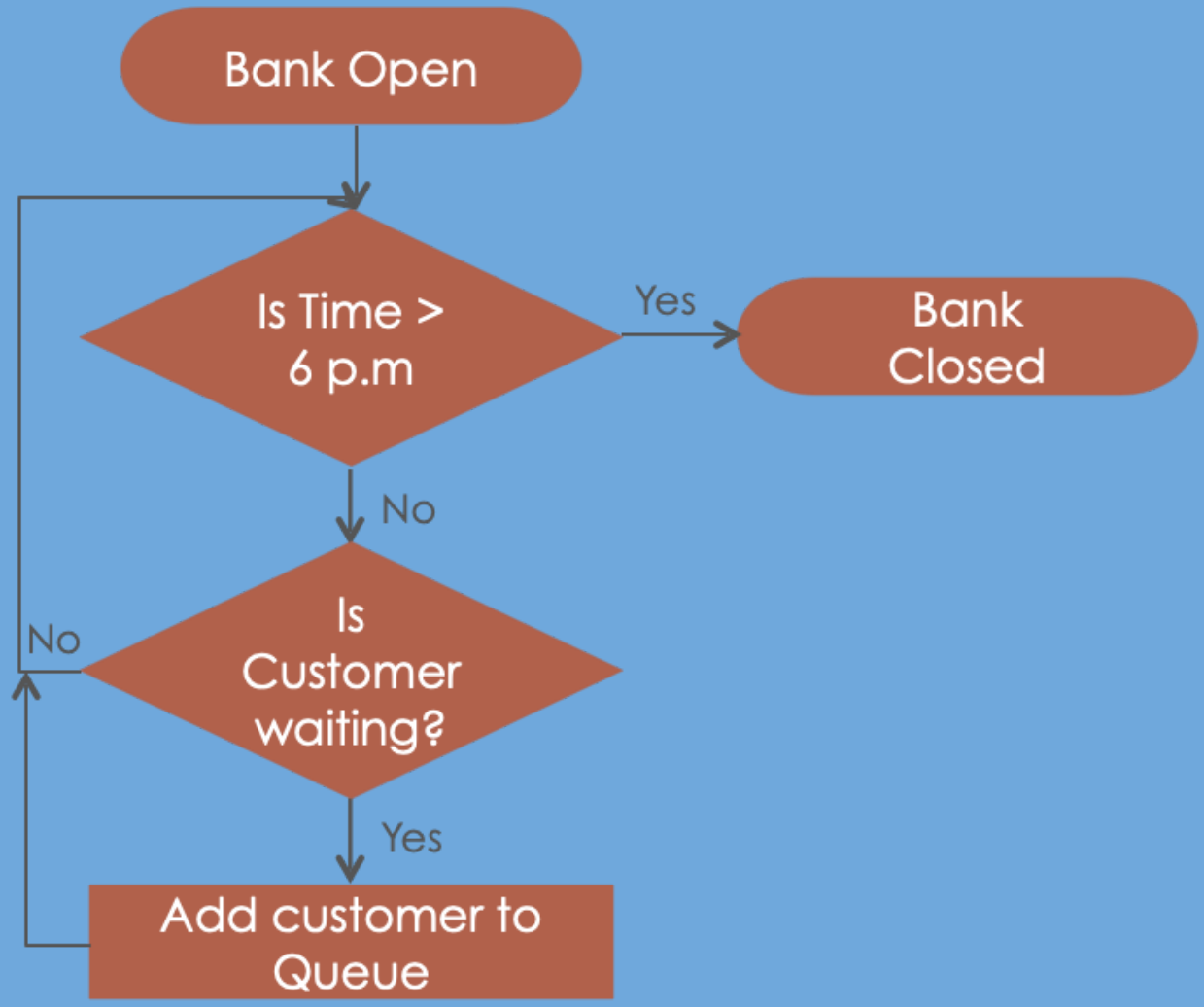
Let's Try

Solution


Bank Open
Is Queue Empty and Time > 6 p.m
Yes
Bank Closed
No
Is Queue Empty
Yes
No
Remove Customer from Queue
Read request
E
No
D
Yes
Is Request for Cash Deposit
No
C
Yes
Is Request for Cash Withdraw al
No
B
Yes
Is Request for Cheque Deposit
No
A
Yes
Is Request for Demand Draft

Flowchart for **Bank Security Guard?**

Solution


Bank Open
Is Time > 6 p.m
Yes
Bank Closed
No
Is Customer waiting?
No
Yes
Add customer to Queue

More Problems

- Factorial of a Number

- Find GCD of two numbers

Homework

Take input N followed by N more numbers, find the sum of those numbers.

eg N = 4

10, 20, 30 , 50

Output : 110

Homework

Find LCM of two numbers A and B example -

A = 10

B = 20

Output:

20

Homework (Optional)

Print all Primes upto N.

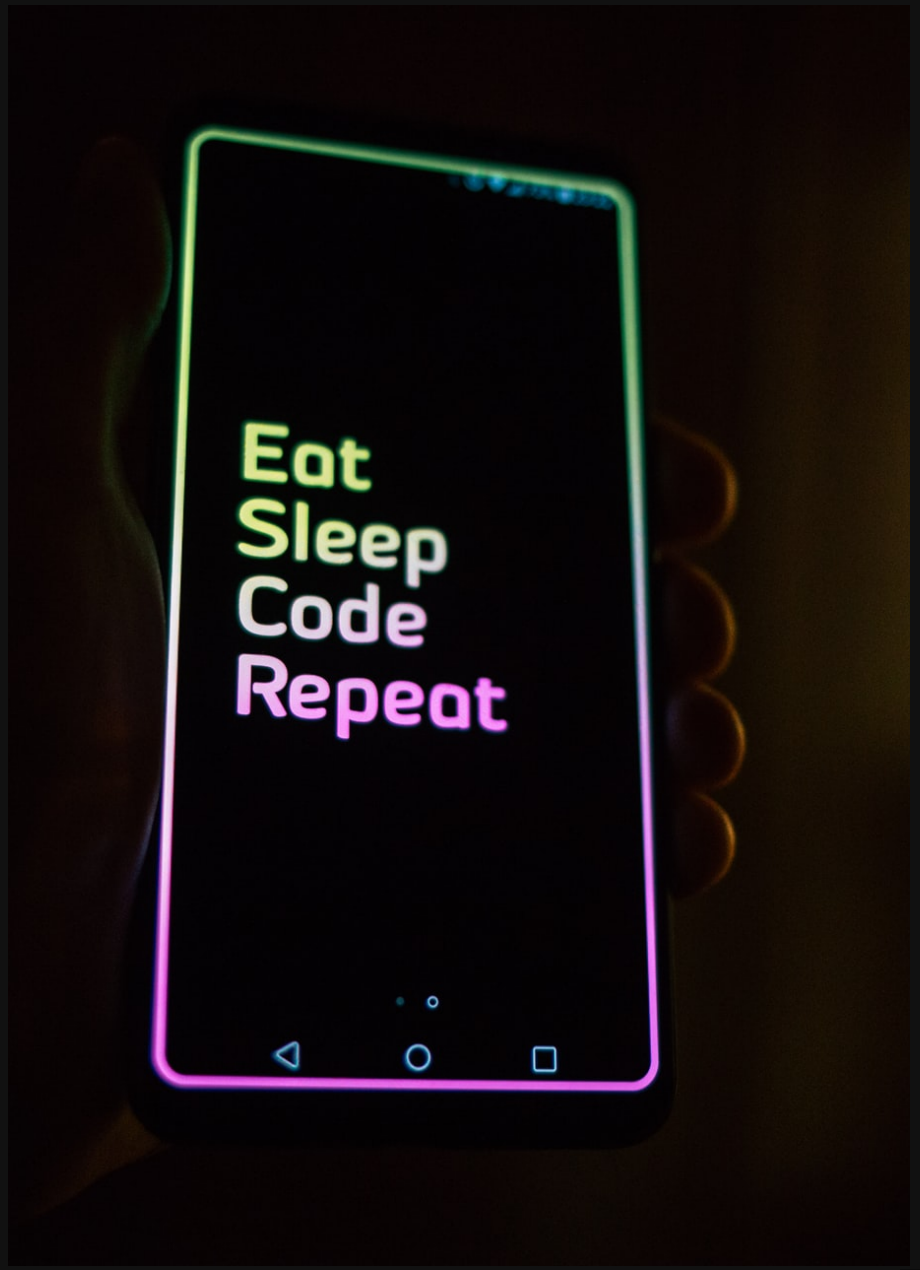
N = 10

Output:
2, 3, 5, 7

Lecture - 2

Pseudocode

Human readable description of an algorithm



It's helpful!

- Language Independent.
- Structure your code before writing it.
- Fastest way to verify / get a review.

Pseudocode

Let's define our own instruction set

- Input [read N]
- Assignment [Sum = 0]
- Output [print Sum]
- If Else
  - [if I < N then ... end else then ... end]
- While Loop
  - [while I < N do ... end]
- Exit [exit]

Prime Number

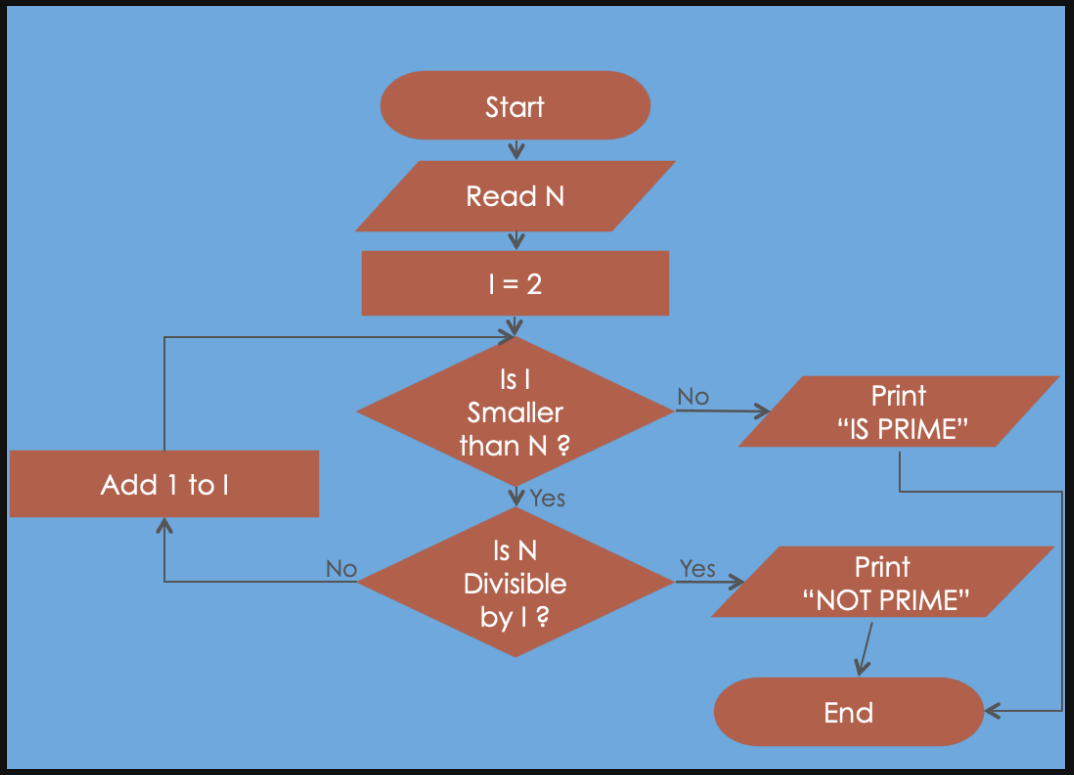
Pseudocode to check if number is prime or not.

Input

11

Output

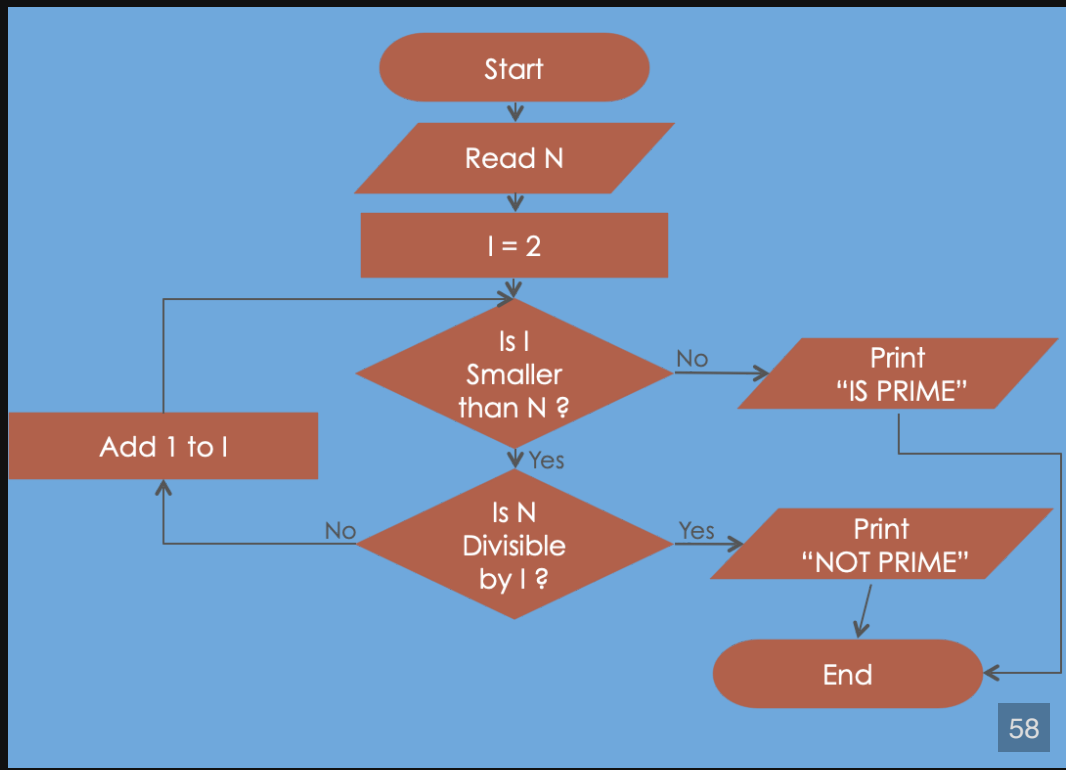
Yes


Start
Read N
I = 2
Is I Smaller than N ?
No
Print "IS PRIME"
Yes
Add 1 to I
Is N Divisible by I ?
No
Yes
Print "NOT PRIME"
End

```
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
```



Number Pattern

Print the following pattern for a given N.

N = 4

1
2 3
4 5 6
7 8 9 10

1

2 3

4 5 6

7 8 9 10

```
read N
Row = 1
Value = 1

while Row <= N do
    Col = 1
    while Col <= Row do
        print Value
        Value = Value + 1
        Col = Col + 1
    end

    print “\n”
    Row = Row + 1
end
exit
```

Number Pattern - II

Write pseudocode to print the following pattern!

```
   1
  232
 34543
4567654
567898765
```

Time to Try!

- Sum of Digits of a Number
- Swap two numbers
- Half Diamond Pattern

N = 3

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

Ganesha's Pattern

Take as input N, an odd number (>=5) . Print the following pattern as given below for N = 7.

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

Next Lecture 3 Programming in JAVA

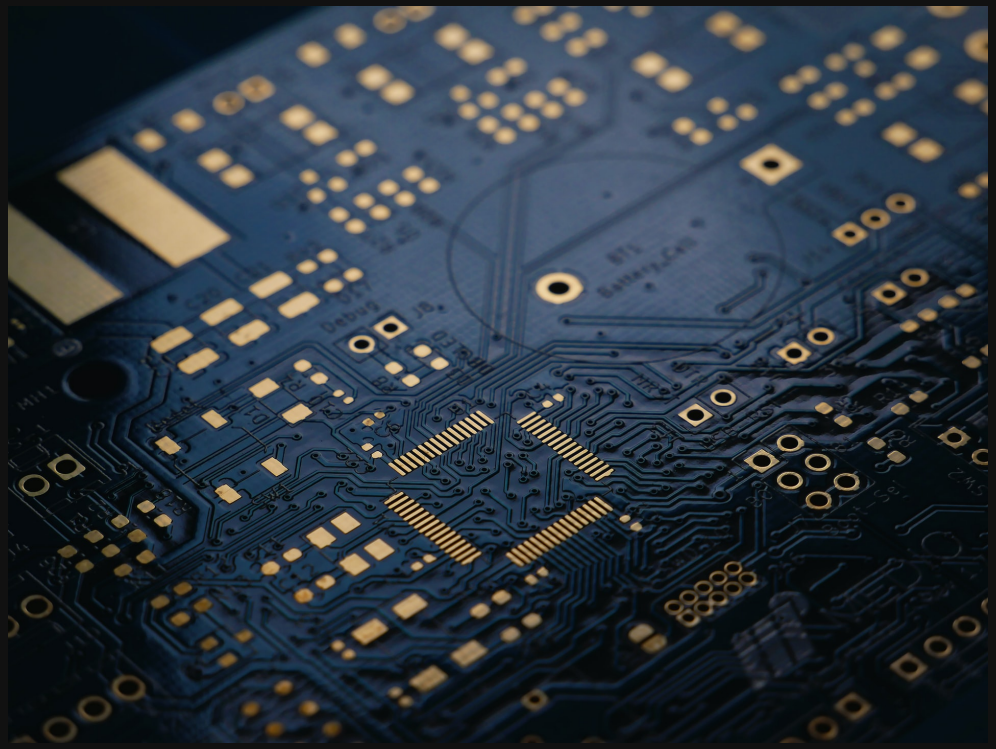
Binary Number System

Story behind

We humans use a decimal, or base-10, numbering system, presumably because people have 10 fingers

Early computers were designed around the decimal numbering system. This approach made the creation of computer logic capabilities unnecessarily complex and did not make efficient use of resources. (For example, 10 vacuum tubes were needed to represent one decimal digit.)

To deal with the basic electronic states of on and off, Von Neumann suggested using the binary numbering system

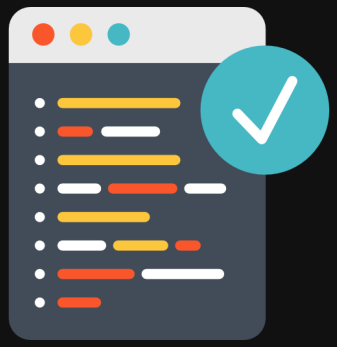


Binary Number System

Programming in JAVA

What happens behind the scenes?

Why Java is Platform independent.



.java file

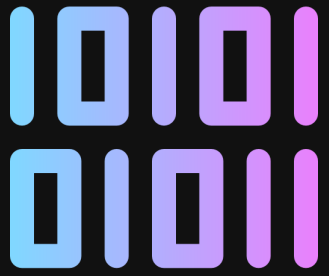
(source code)

compiler

.class file

(byte code)

interpreter

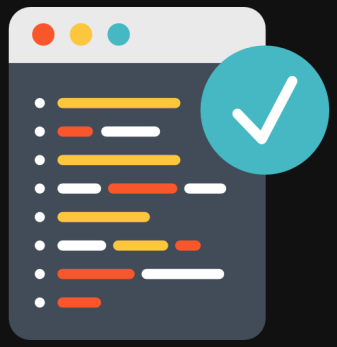


Machine Code

(0s & 1s)

What happens behind the scenes?

Why Java is Platform independent.



.java file

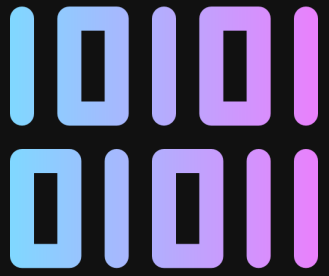
(source code)

compiler

.class file

(byte code)

interpreter



Machine Code

(0s & 1s)

Byte Code doesn't run directly, we need a JVM (Java Virtual Machine) to run this code

Platform Independence?

Byte-Code can run all operating systems

C++

C/C++ Compiler generates a .exe file which is platform dependent.

Python

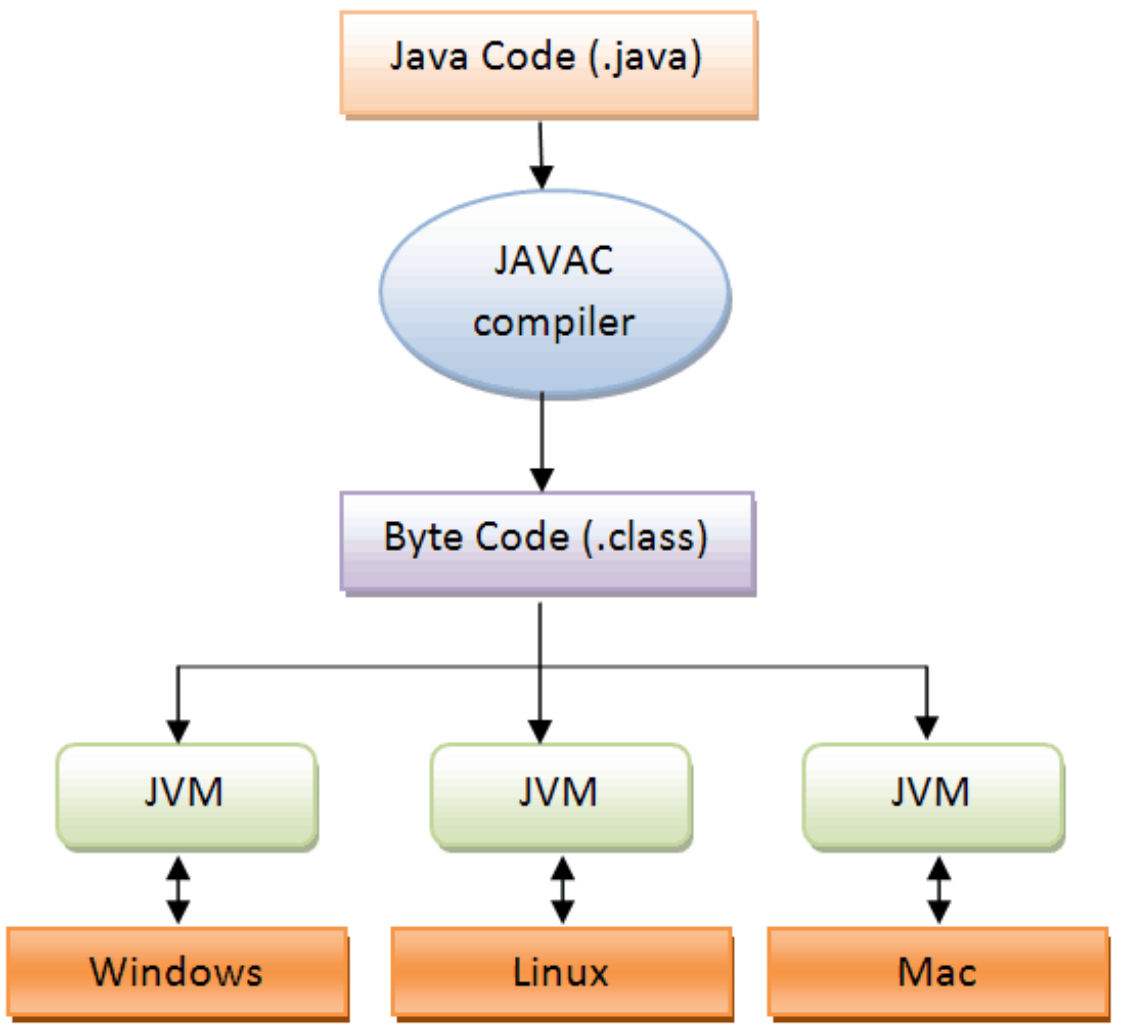
Python is interpreted language, instructions are executed line by line.

Java

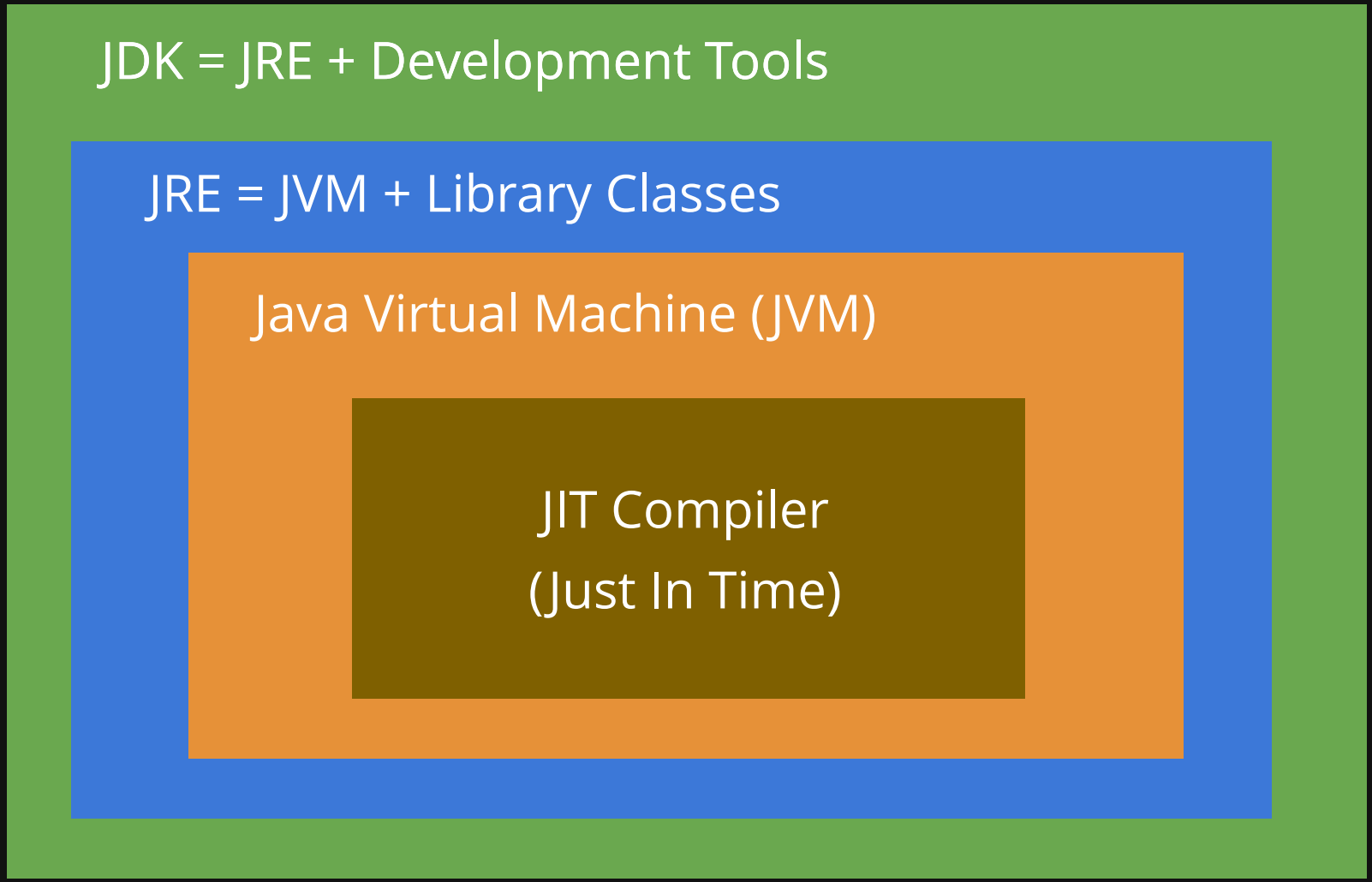
In Java we get bytecode, JVM converts this bytecode into machine code.

To run byte-code, we need a JVM installed on a machine.

Java is platform independent but JVM is platform dependent.


Java Code (.java)
JAVAC compiler
Byte Code (.class)
JVM
JVM
JVM
Windows
Linux
Mac

Architecture



JDK

Consists of Development tools & environment to run the Java program.

1. Development Tools
2. Compiler (javac)
3. Archiver (jar)
4. docs generator (javadoc)
5. interpreter / loader

JRE

Provides environment to only run the program.

JRE = JVM + Additional technologies & features

1. Deployment solutions
2. Development toolkits & libraries
3. Integration libraries
4. Language and utility libraries
   (such as collections framework)

JVM 🤔

Java Virtual Machine

JVM

Java Virtual Machine, or JVM, loads, verifies and executes Java bytecode. It is known as the interpreter or the core of Java programming language because it executes Java code.

- Class Loader Subsystem is responsible for loading, linking and initializing a Java class file (i.e., “Java file”), otherwise known as dynamic class loading.
- Runtime Data Areas contain method areas, PC registers, stack areas and threads.
- Execution Engine contains an interpreter, compiler and garbage collection area.

Summary

The JRE combines Java code created using the JDK with the necessary libraries required to run it on a JVM and then creates an instance of the JVM that executes the resulting program.

JVMs are available for multiple operating systems, and programs created with the JRE will run on all of them. In this way, the Java Runtime Environment is what enables a Java program to run in any operating system without modification.

Software Tools [OPTIONAL] 🤔

IntelliJ Idea (IDE)

https://www.jetbrains.com/idea/download

(Free Community Edition)

JDK

https://www.oracle.com/java/technologies/downloads/#java16

Recommended for beginners

Coding Minutes IDE

ide.codingminutes.com

No Installation Required!

Directly Run | Save | Share Codes!

First Java Program

Boilerplate Code

```
1 //Boilerplate Code
2 class Main
3 {
4         public static void main (String[] args)
5         {
6
7         }
8 }
```

First Java Program

Boilerplate Code

```
1 //Boilerplate Code
2 class Main
3 {
4         public static void main (String[] args)
5         {
6                 //Your Code goes here
7         }
8 }
```

First Java Program

Boilerplate Code

```
1 //Boilerplate Code
2 class Main
3 {
4         public static void main (String[] args)
5         {
6                 //Your Code goes here
7                 System.out.println("Hello World");
8         }
9 }
```

Let's Solve Problems & learn Java syntax!

1. Hello Java
2. Variables
3. Loops
4. Typecasting
5. Problems - Prime Numbers, Patterns, Simple Interest

Ternary Operator

```
variable = (condition) ? expressionTrue :  expressionFalse;
```