WEEKLY PRESENTATION

Presented By : Shruthi R K

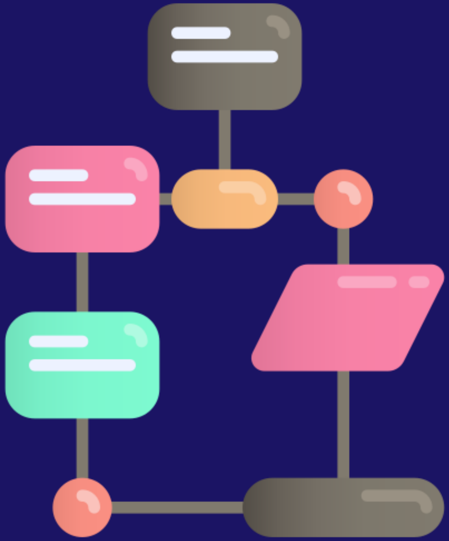
Introduction To Algorithms

1. A set of steps to accomplish a task.
2. We do use algorithms in day-to-day life.
3. Algorithms can be expresses diagrammatically using flowcharts.

Algorithm restaurant management system

- step 1: Start
- step 2: receive the menu
- step 3: search for the good food and cost
- step 4: cost min 100
- step 5: order
- step 6: receive another menu
- step 7: cost>100
- step 8: order
- step 9: stop

Flowcharts



There is saying that a picture is worth thousand words, likewise in programming,
a solution can be well expressed using flowcharts.

Start/stop: A flowchart terminator used at the beginning and end of the algorithm

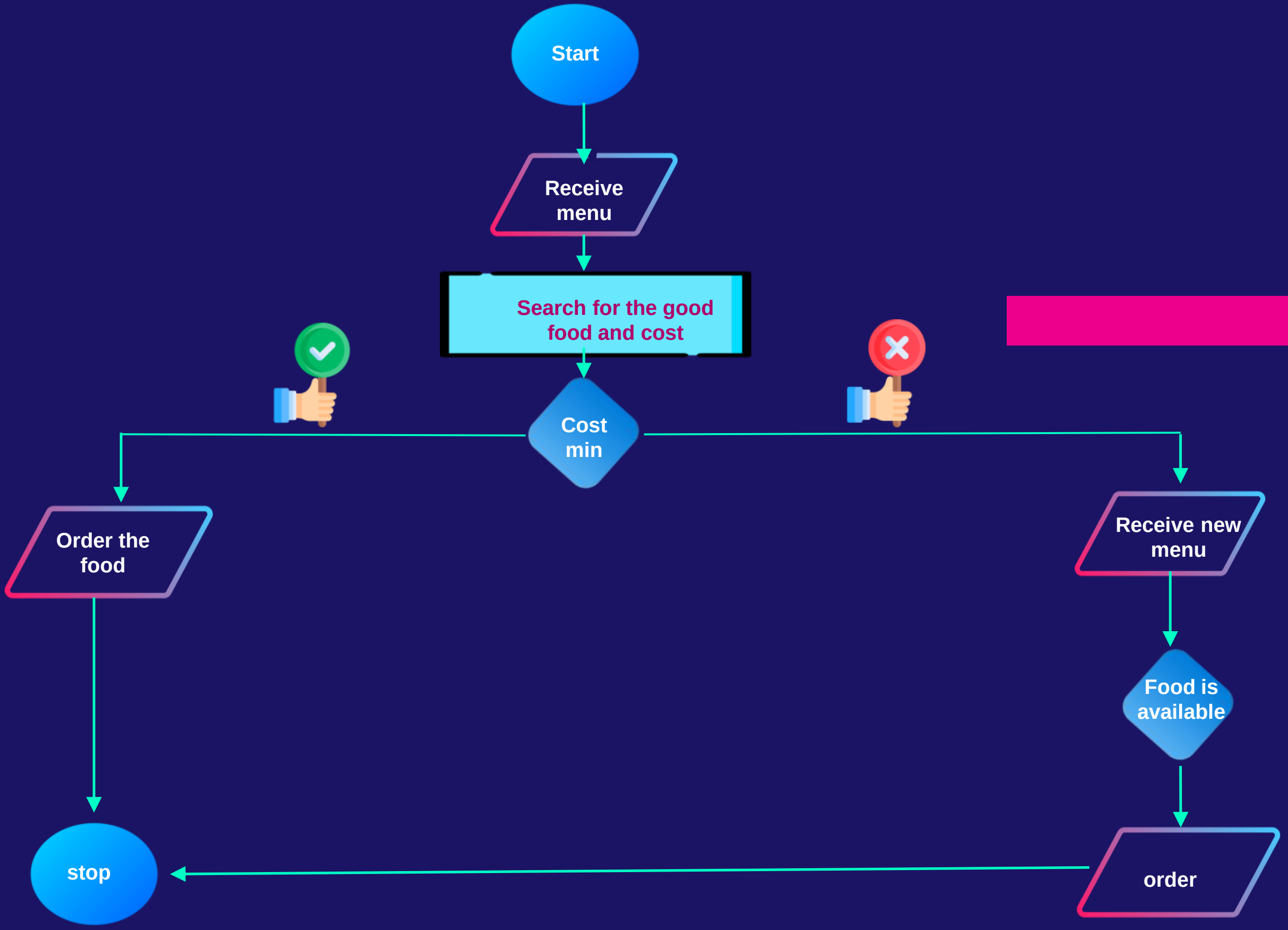
Arrow: A line connector that shows the logical flow of the process.

Input/Output: A parallelogram used for denoting program inputs and outputs.

Process: A rectangle, which indicates logic blocks with instructions.

Decision: A diamond that stands for decision statements in a program, where answer is either Yes or No.

Looping: Repeats the process multiple times.

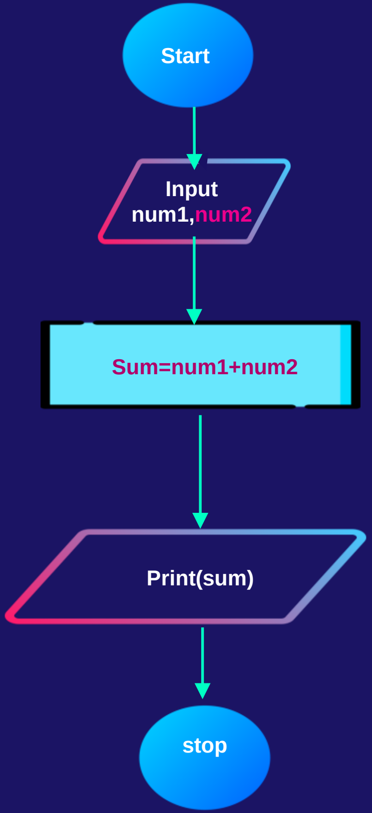

Start
Receive menu
Search for the good food and cost
Cost min
Order the food
Receive new menu
Food is available
order
stop

PSEUDO CODE

- Pseudocode is a text-based algorithm to instruct a computer to perform various tasks.
- It is expressed in an informal language. which is usually English.

EXAMPLE

```
//Program to find the sum of two numbers.
Begin
Numeric num1,num2
Print("enter the num1,num2")
Input num1
Input num2
Sum=num1+num2
Print(sum)
```


Start
Input num1,num2
Sum=num1+num2
Print(sum)
stop

FLOW: //Arithmetic operation

```
begin
numeric num1,num2,sum,difference,product,quotient
print("enter the num1,num2 value”)
input num1,num2
sum=num1+num2
difference=num1-num2
product=num1*num2
quotient=num1/num2
print(“The Addition of” +num1 “and” +num2 “is” +sum)
print((“The Subtraction of” +num1 “and” +num2 “is” +difference)
print(product)
print(quotient)
end
```

CODE

//Sum of two number

```
class Exampleprogram1
{
  public static void main(String args[ ])
    {
      int num1=10,num2=20,sum;
      sum=num1+num2;
      System.out.println(sum);
    }
}
```

IF STATEMENT

Use if to specify a **block of code** to be executed, if a specified condition is true or false.

FLOW: //Voting Eligibility Check

```
begin
numeric age
print("Enter the age")
input age
if(age>=18)
print("Eligible to vote")
else
print("Not Eligible to vote")
end
```

CODE://Taking input from the user

```
import java.util.Scanner;
class Week1pratice
{
public static void main (String args[ ])
{
Scanner s=new Scanner(System.in);
int age;
System.out.println("Enter the age");
age=s.nextInt();
if(age>18)
{
  System.out.println("Eligible for voting");
}
else
{
  System.out.println("Not Eligible for voting");
}
}
}
```

IF ELSE STATEMENT

Use else if to specify a new condition to test, if the first condition is false

FLOW://Even or odd check

- begin
- numeric num
- print(“Enter the number")
- input num
- if(num%2==0)
- print(“The number is even")
- else
- print(“The number is odd")
- end

CODE

```
import java.util.Scanner;
class Week1pratice
{
public static void main (String args[ ])
{
Scanner s=new Scanner(System.in);
int num=0;
System.out.println("Enter the num");
num=s.nextInt();
if(num%2==0)
{
  System.out.println("Even number");
}
else
{
  System.out.println("Odd number");
}
}
}
```

WHILE LOOP

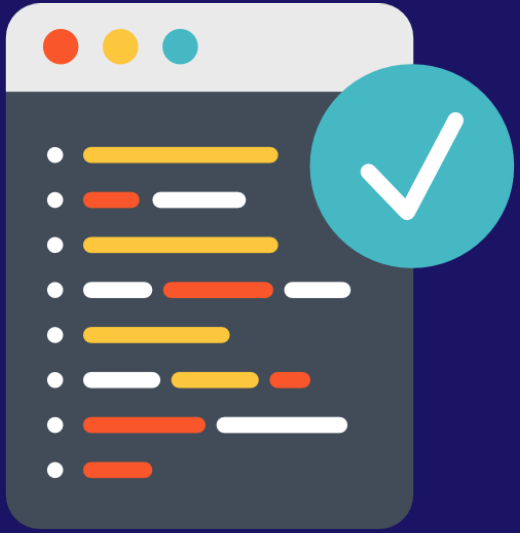
The Java *while loop* is used to iterate a part of
the program repeatedly until the specified Boolean condition is true

FLOW.. //printing even numbers

```
begin
numeric num=1
while(num<=n)
{
if(num%2==0){
print(num)
}
num++
}
end
```

DO WHILE LOOP

Java do-while loop is called an exit control loop. Therefore, unlike while loop and for loop, the do-while check the condition at the end of loop body. The Java *do-while loop* is executed at least once because condition is checked after loop body.



FLOW

```
begin
numeric num
print("enter the num")
input num
do
{
print(num)
num++
}
while(num<=10)
end
```

CODE

```
import java.util.Scanner;
class Week1pratice{
public static void main (String args[ ])
{
Scanner s=new Scanner(System.in);
int s1,s2,s3;
System.out.println("enter the marks");
s1=s.nextInt();
System.out.println("enter the marks");
s2=s.nextInt();
System.out.println("enter the marks");
s3=s.nextInt();
int sum=0;
int total=0;
if(s1>90&&s2>80&&s3>50)
  sum=s1+s2+s3;
  System.out.println(sum);
    total=(70*s1/100)+(20*s2/100)+(10*s3/100);
  System.out.println(total);
}}
```

PRINTING THE START AND STOP VALUE

```
begin
numeric startvalue,stopvalue
print("enter startvalue and stopvalue)
input startvalue
input stopvalue
while(startvalue<=stopvalue){
print ("startvalue")
startvalue++
}
end
```

PRINTING THE MID VALUE

```
begin
numeric startvalue,stopvalue, s
print("enter the start value")
input start value
print("enter the stop value")
input stop value
while(stop value<=10)
{
s=stop value-start value/2
print(s)
}
end
```

Java is a **high-level**, **class-based**, **object-oriented** programming language ,Java is a platform-independent language. Java achieves this using JVM and Byte Code. Java compiler converts the programming code into byte code.

INTRODUCTION TO JRE, JVM, JDK AND JIT

JRE

- **Java Virtual Machine (JVM)** is an abstract computing machine.

JVM

Java Runtime Environment (JRE) is an implementation of the JVM

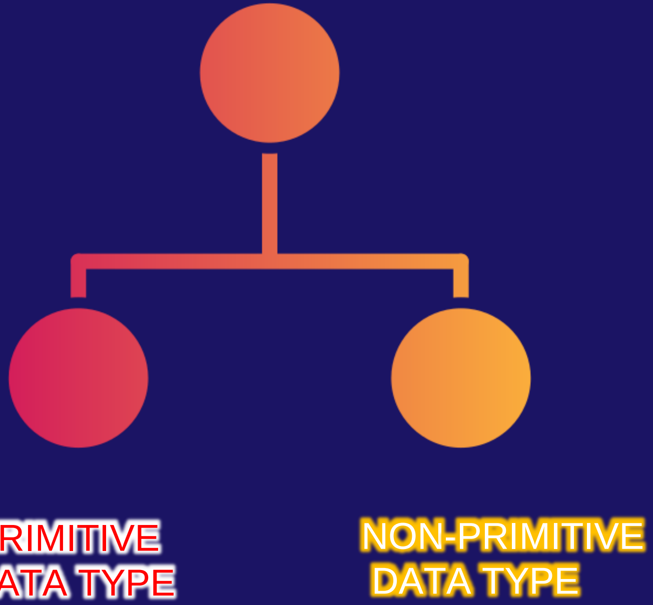
JDK

Java Development Kit (JDK) contains JRE along with various development tools

JIT

Just In Time compiler (JIT) is runs after the program has started executing, on the fly.

DATA TYPES



PRIMITIVE DATA TYPES

byte 1 byte

Stores whole numbers from -128 to 127

short 2 bytes

Stores whole numbers from -32,768 to 32,767

int 4 bytes

Stores whole numbers from -2,147,483,648 to 2,147,483,647

long 8 bytes

Stores whole numbers from -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807

float 4 bytes

Stores fractional numbers. Sufficient for storing 6 to 7 decimal digits

double 8 bytes

Stores fractional numbers. Sufficient for storing 15 decimal digits

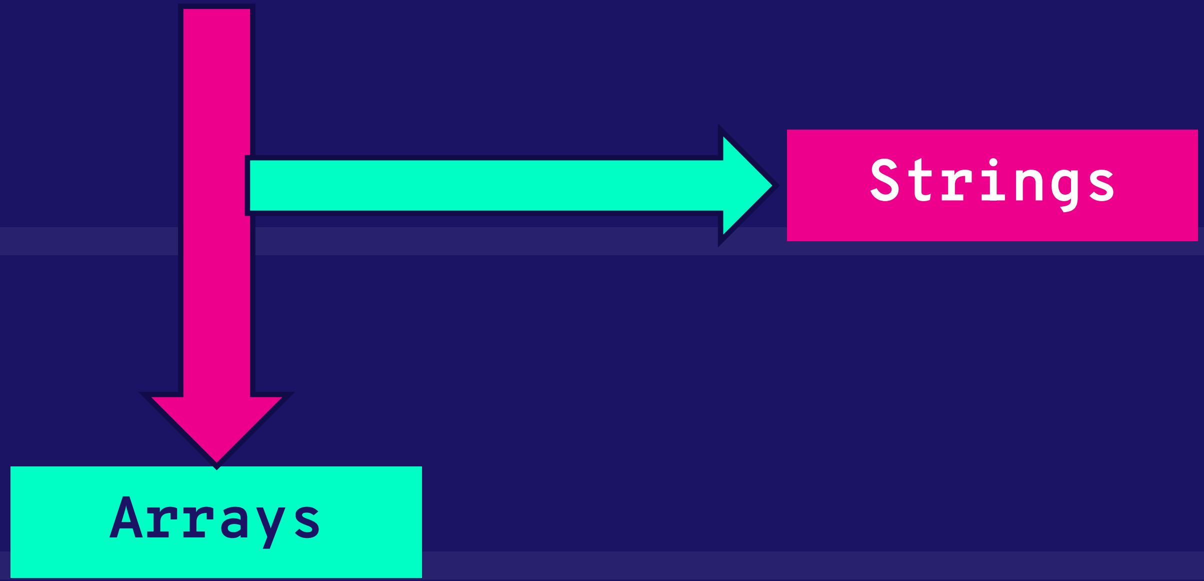
boolean 1 bit

Stores true or false values

char 4 bytes

Stores single character values or ASCII

NON PRIMITIVE DATA TYPES



THANK

YOU

Java