Topic 01

Introduction

Problem Solving and Program Design

Prepared by: Suhaini Nordin

Acknowledgement: Some contents in this notes are edited from original notes prepared by Ban Kar Weng (William)

Topic 1

Introduction to Computing and Problem Solving

Questions and Answers

How do we communicate with the computer?

- Programming languages

How do we get computers to perform complicated tasks?

- Tasks are broken down into a sequence of instructions

Why Python?

- Powerful, easy to download, write and read

How did the language Python get its name?

- Named for the British comedy group Monty Python (really!)

What is IDLE?

- Stands for Integrated DeveLopment Environment
- Python Shell and text editor that we will use to write the program

Questions and Answers

What is an interpreted language?

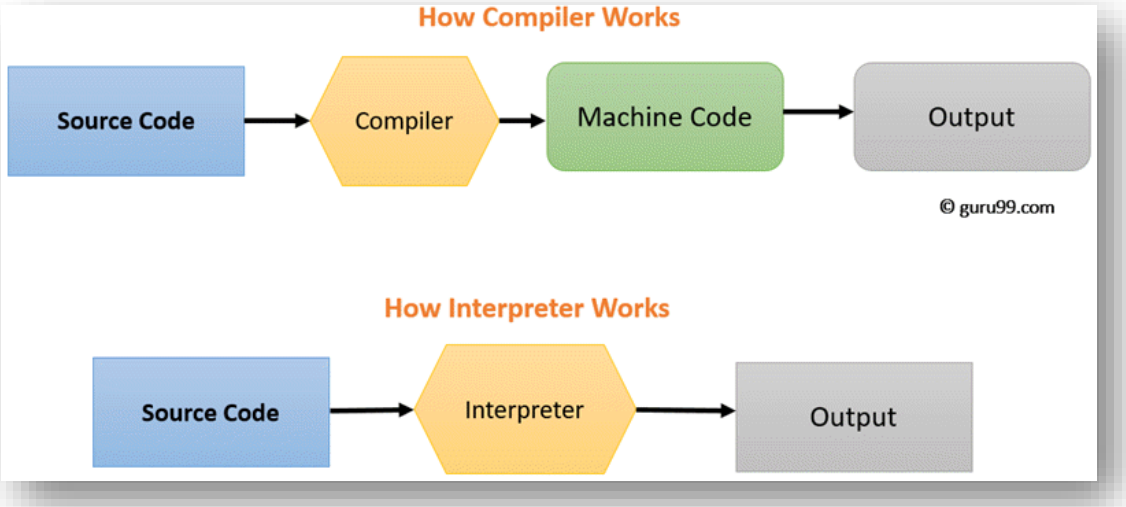
- Uses an interpreter, translates high-level language one statement at a time into machine language and then runs

What are the meanings of the terms “programmer” and “user”?

- Programmer: A person who solves problems by writing programs on a computer
- User: Any person who runs a program

What is the meaning of the term “code”?

- Python statements that the programmer writes



Questions and Answers

Are there certain characteristics that all programs have in common?

- Input, processing, output

What are the meanings of the terms “hardware” and “software”?

- Hardware: Physical components of the computer
- Software: The program

How are problems solved with a program?

- Step-by-step procedure devised to process given data and produce requested output.

What is a zero-based numbering system?

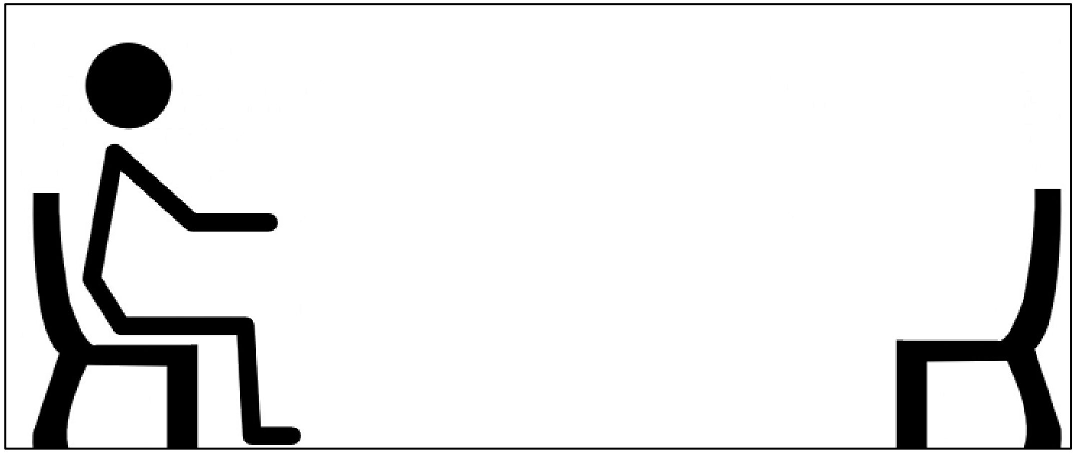
- Numbering begins with zero instead of one

Prerequisites to learning Python?

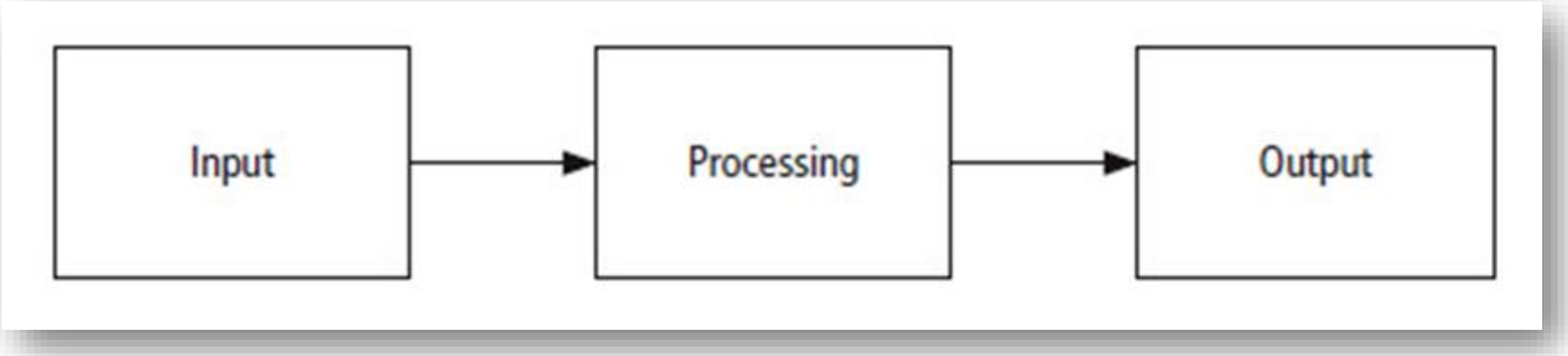
- Be familiar with how folders and files are managed

Test Your Problem Solving Skill

- Jojo is a robot that has a very limited knowledge base. He can only store 4 instructions at a time.
- List down 4 instructions that Jojo can obey and based on Jojo's knowledge base, write the instruction to make Jojo (in sitting position) to walk 4 steps forward and sit on a chair at the opposite end.

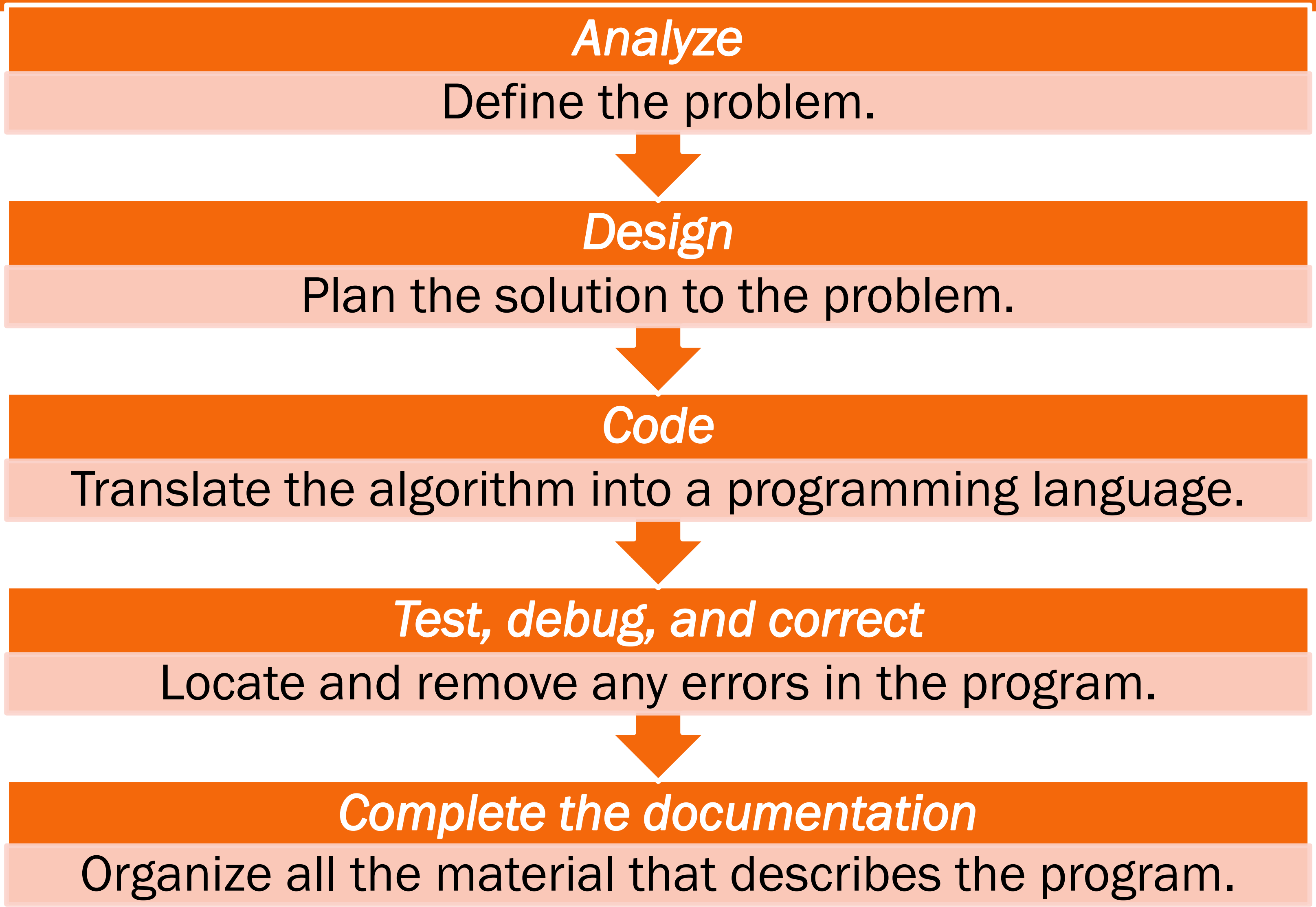


Program Development Cycle



The problem-solving process

Program Planning



Analyze

Define the problem.

Design

Plan the solution to the problem.

Code

Translate the algorithm into a programming language.

Test, debug, and correct

Locate and remove any errors in the program.

Complete the documentation

Organize all the material that describes the program.

Problem Solving in Mathematics

What is 5! ?

Problem Solving in Mathematics

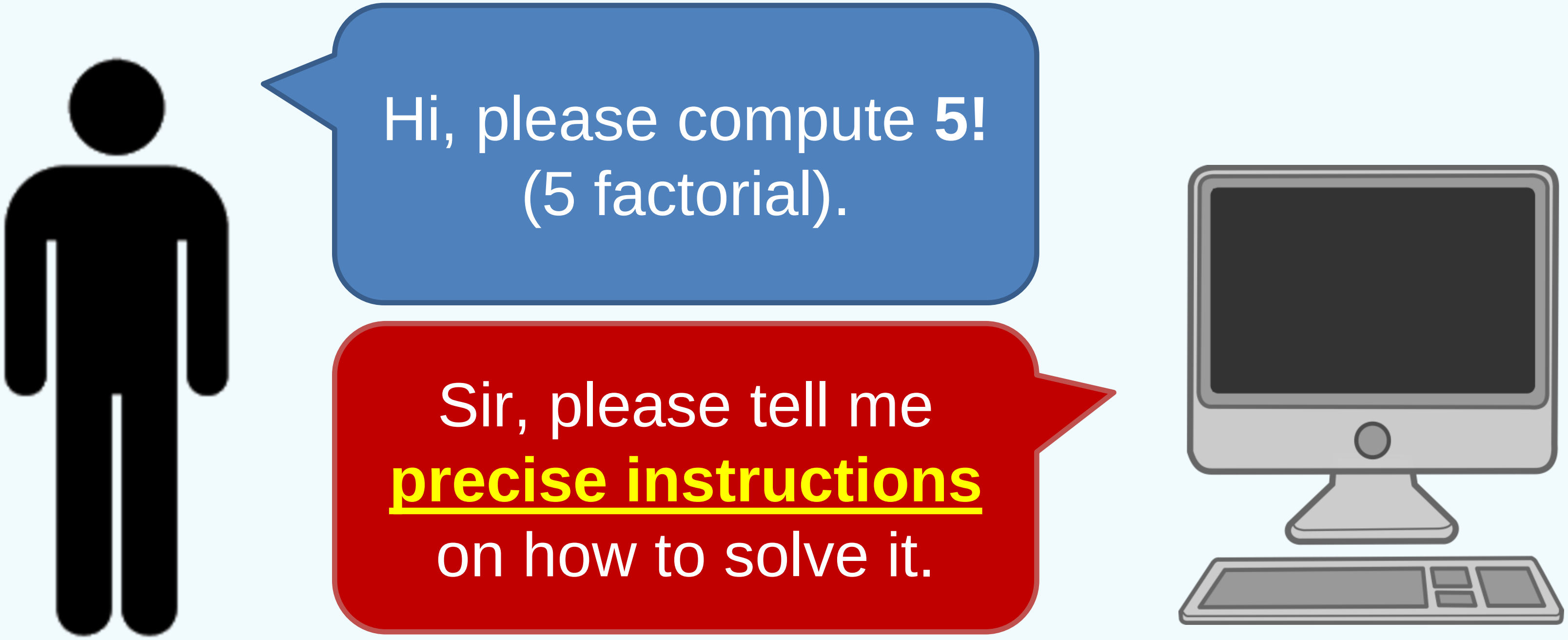
What is 5! ?

$$5! = 5 \ \times 4 \times 3 \times 2 \times 1$$

$$= 120$$

What is Programming?

Tell a computer to perform a task for us.

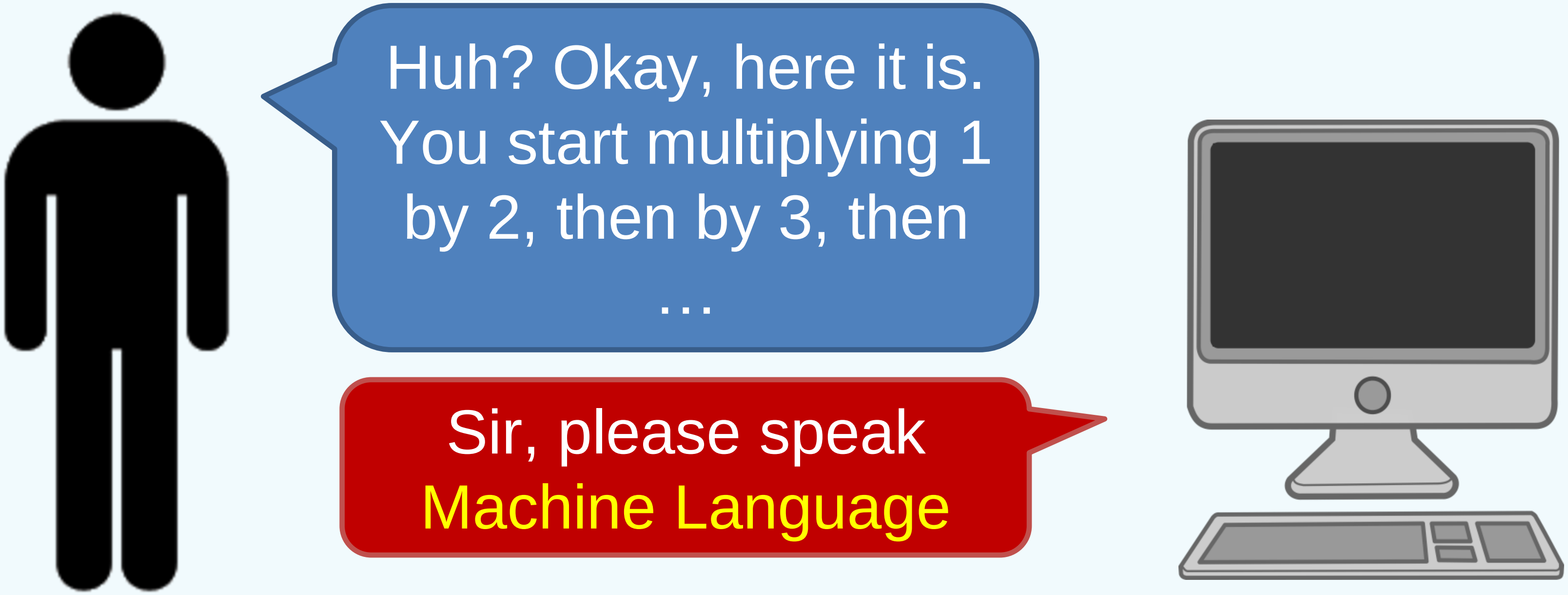


Hi, please compute **5!** (5 factorial).

Sir, please tell me **precise instructions** on how to solve it.

What is Programming?

Tell a computer to perform a task for us.

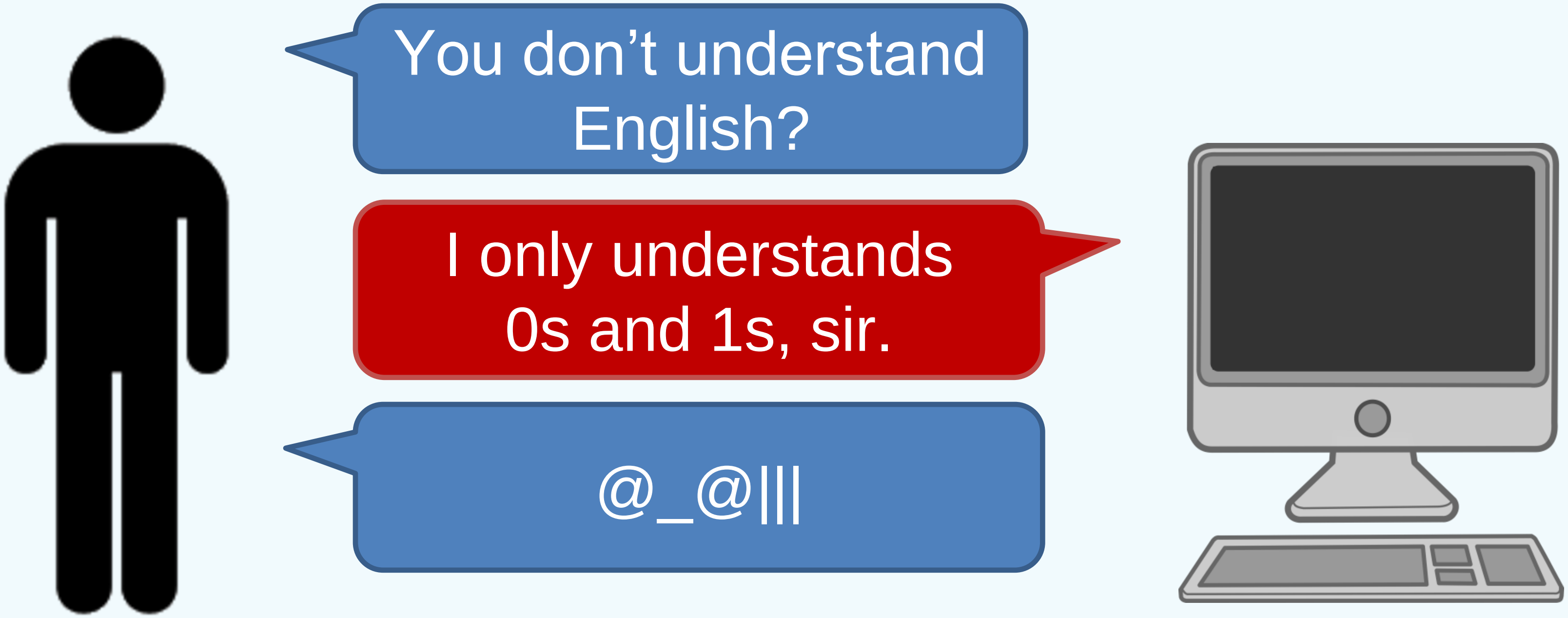


Huh? Okay, here it is. You start multiplying 1 by 2, then by 3, then …

Sir, please speak Machine Language

What is Programming?

Tell a computer to perform a task for us.



You don’t understand English?

I only understands 0s and 1s, sir.

@_@|||

Problem in Communicating with Machine

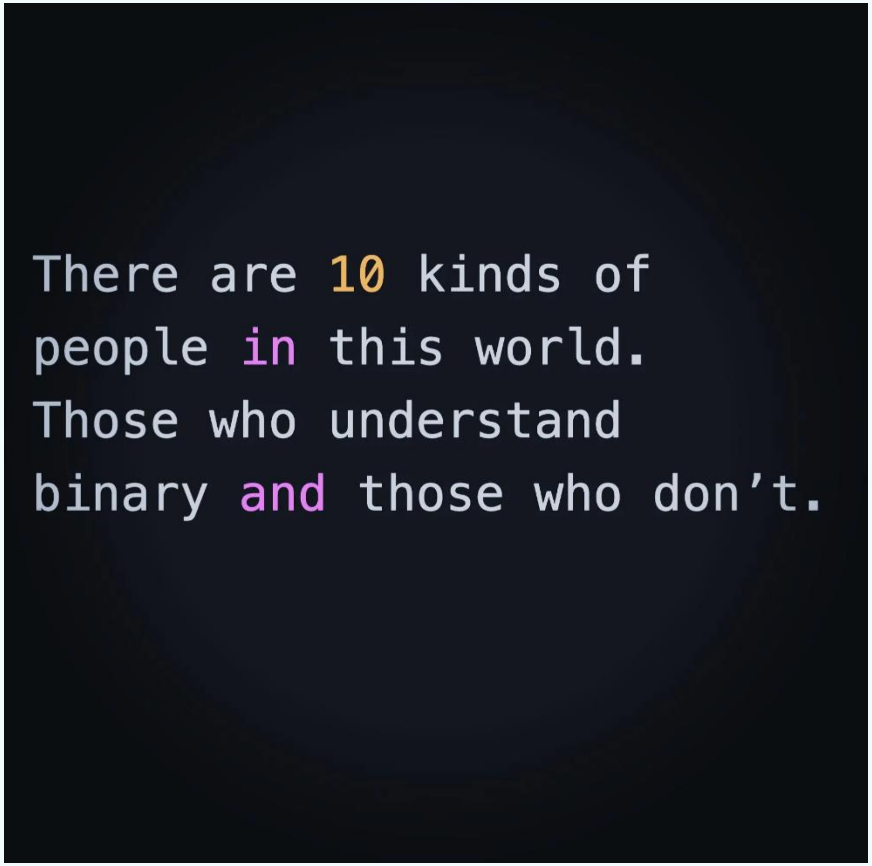
Need

Precise

Instructions

in

Machine Language



There are 10 kinds of
people in this world.
Those who understand
binary and those who don’t.

Precise Instructions

Example: How to find 5!

1) Start by multiplying 1 and 2
2) Then multiplying the result of step (1) by 3
3) Then multiplying the result of step (2) by 4
4) Then multiplying the result of step (3) by 5

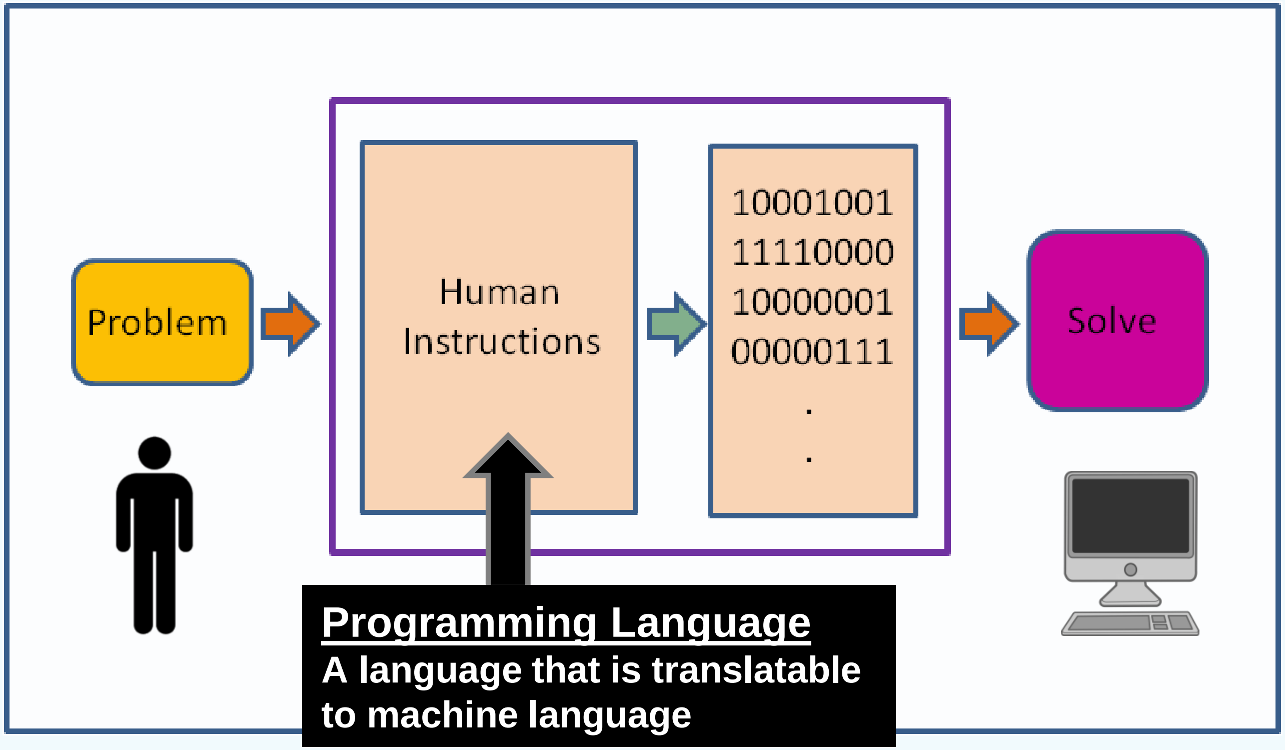
Machine Language = **Binary Code**

- Computer can only understand machine language.
- Pretty much impossible for human to use to communicate with the machine.

Example: How to find 5!

1) 0000111010010010101011000011010
2) 0110101000110000101110000100100
3) 0011001101101111100010101001011
4) 0011100111000001111011100001011
5) 1110100101111010101001101101111

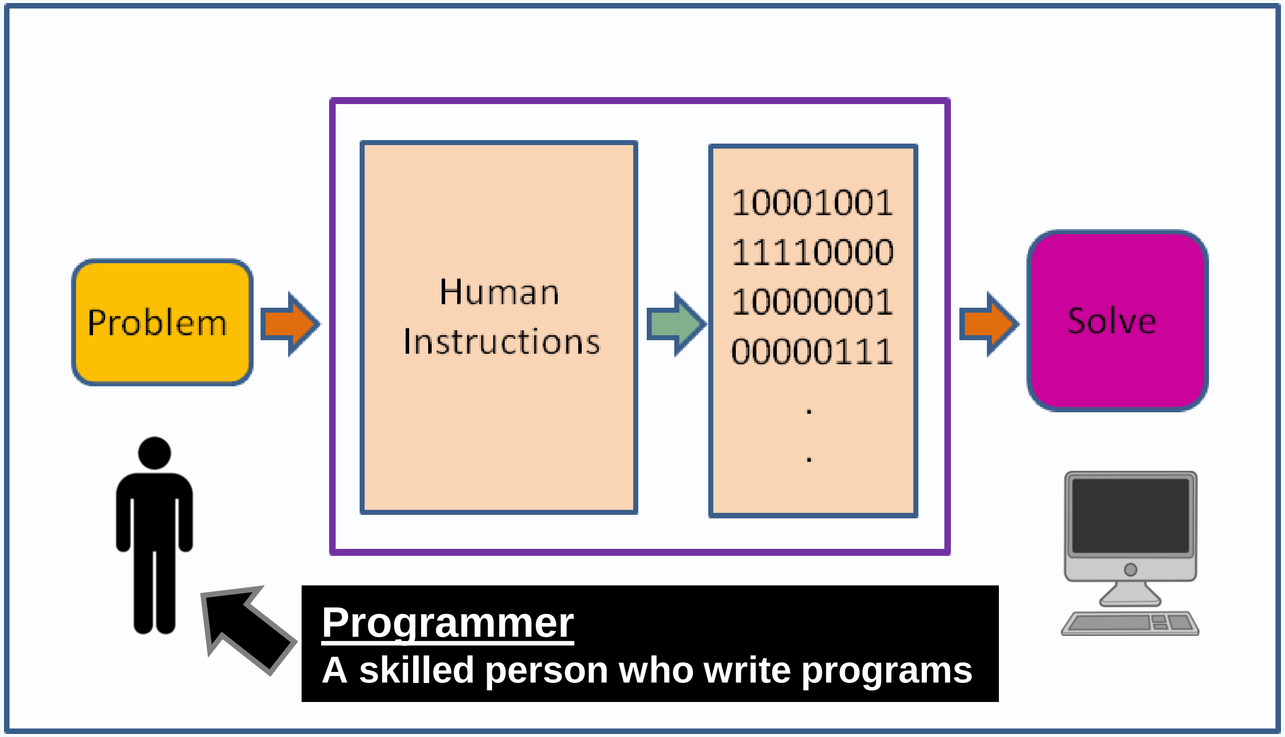
Programming Language



Programming Language
A language that is translatable to machine language

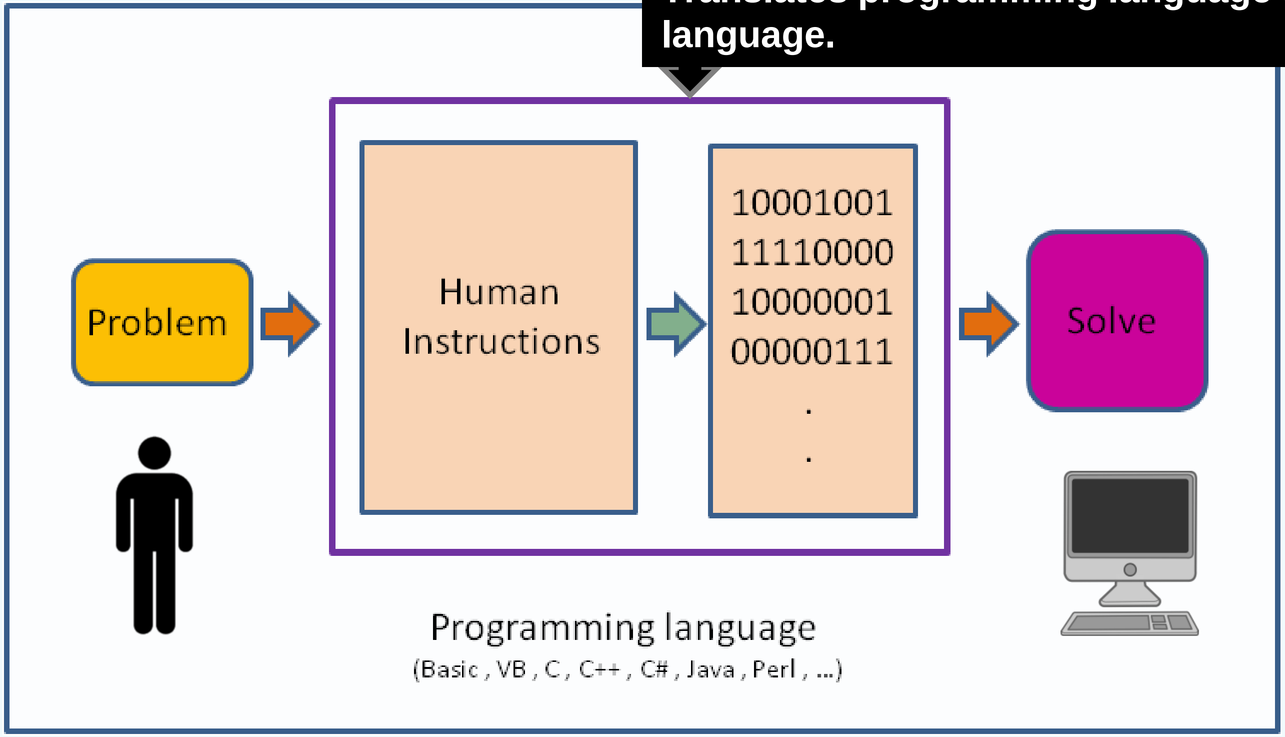
Source : https://upload.wikimedia.org/wikipedia/commons/2/2d/Programming_language.png

Programmer



Source : https://upload.wikimedia.org/wikipedia/commons/2/2d/Programming_language.png

Translator



Translator
Translates programming language to machine language.

Source : https://upload.wikimedia.org/wikipedia/commons/2/2d/Programming_language.png

Programming Tools

Algorithms

Flowcharts

Pseudocode

Algorithm Development

Problem: To determine number of stamps for a letter

- Rule of thumb: 1 stamp for every 5 sheets of paper

Algorithm to determine number of stamps for a letter

- Rule of thumb: 1 stamp for every 5 sheets of paper

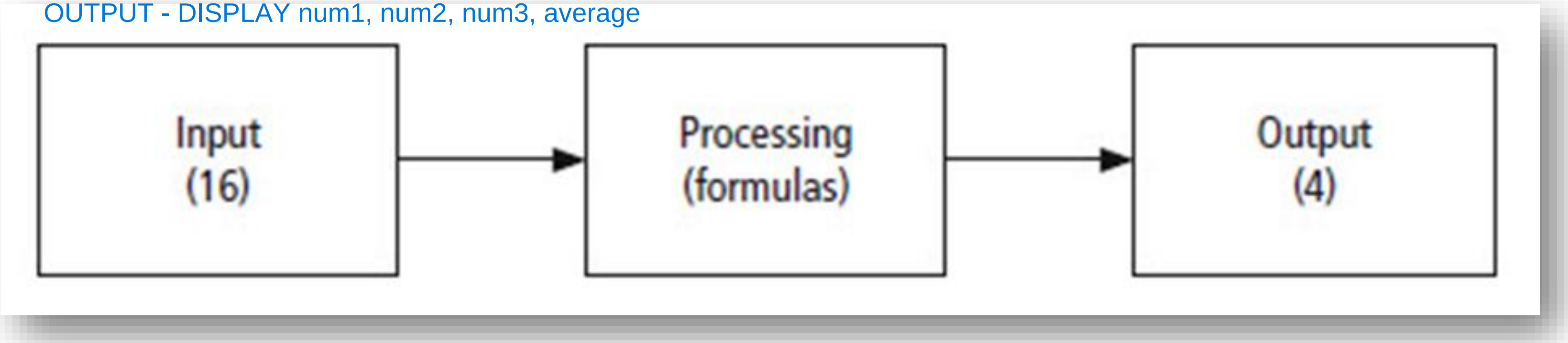
1. Get how many sheets of paper INPUT
2. Divide sheets of paper by 5 PROCESS
3. Round the answer from (2) quotient up to next whole number and store it as number of stamps PROCESS
4. Display the number of stamps OUTPUT

Problem Solving for Stamps

```
1. INPUT - GET num1, num2
   PROCESS - result = num1 - num2
   OUTPUT - DISPLAY result
```

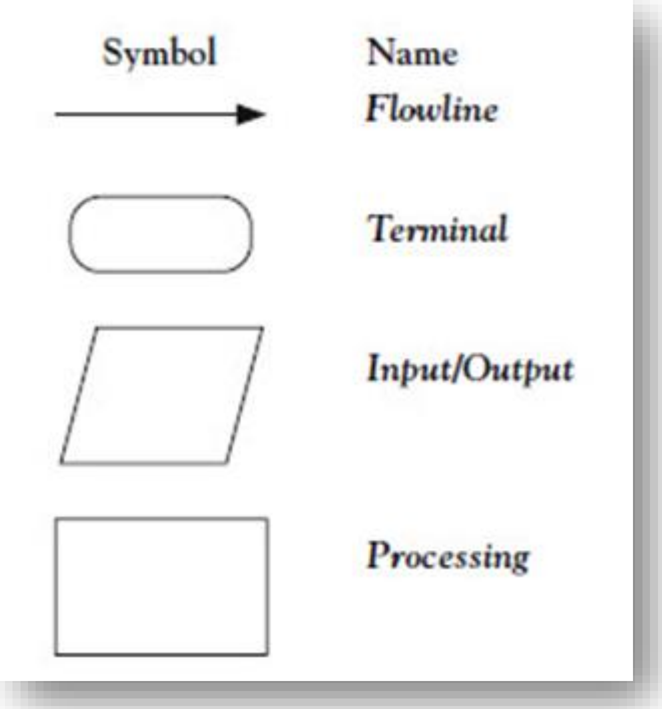
```
2. INPUT - GET num1, num2, num3
   PROCESS - result = num1*num2*num3
   OUTPUT - DISPLAY result
```

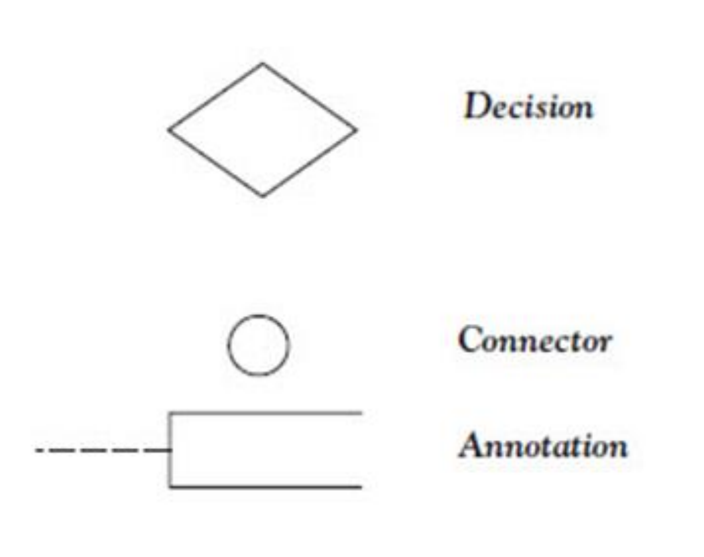
```
3. INPUT - GET num1, num2, num3
   PROCESS - average = (num1+num2+num3)/3
   OUTPUT - DISPLAY num1, num2, num3, average
```



The problem-solving process for the stamp problem for 16 sheets of paper

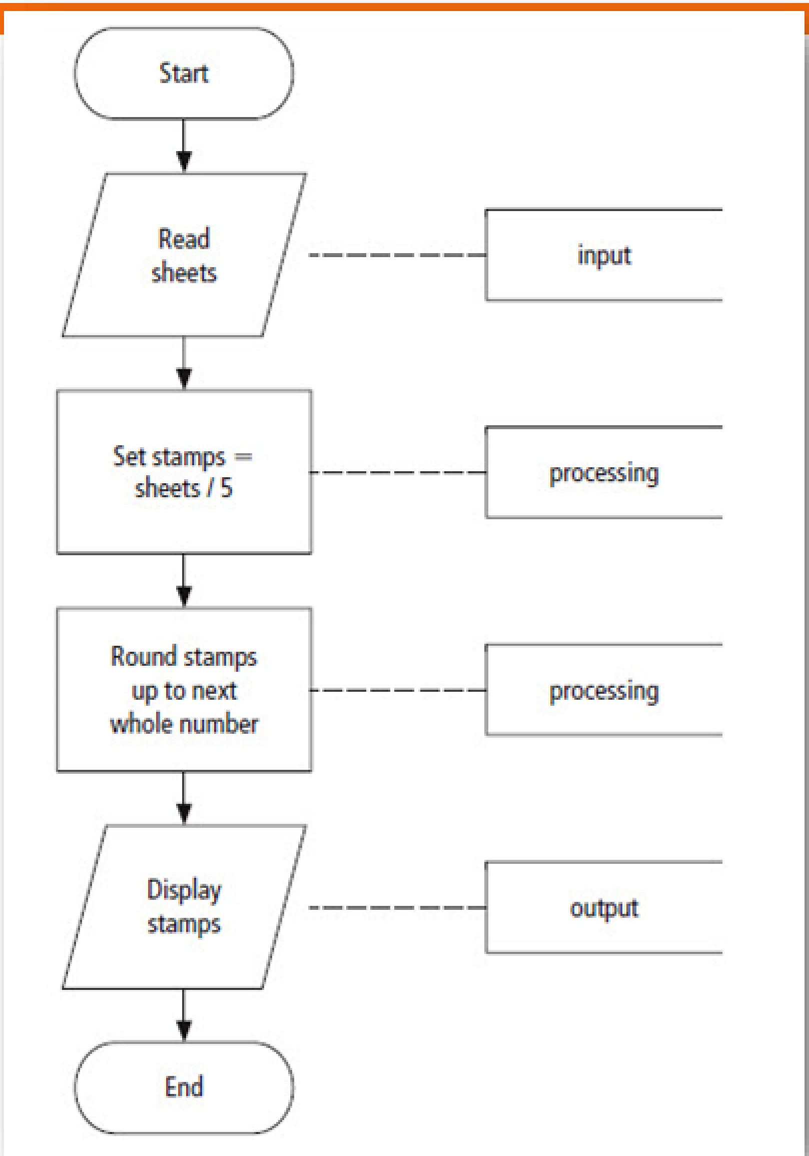
Flowcharts


Symbol
Name
Flowline
Terminal
Input/Output
Processing


Decision
Connector
Annotation

Example Flowchart

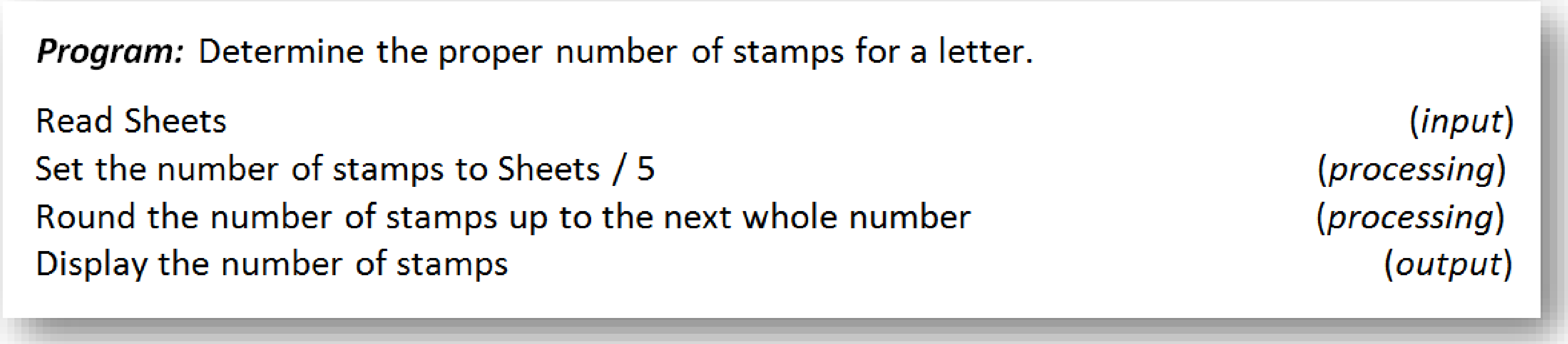
Flowchart for the postage-stamp problem.



Pseudocode

- Abbreviated plain English version of actual computer code
- Symbols used in flowcharts replaced by English-like statements
- Allows programmer to focus on steps required to solve problem

Example Pseudocode



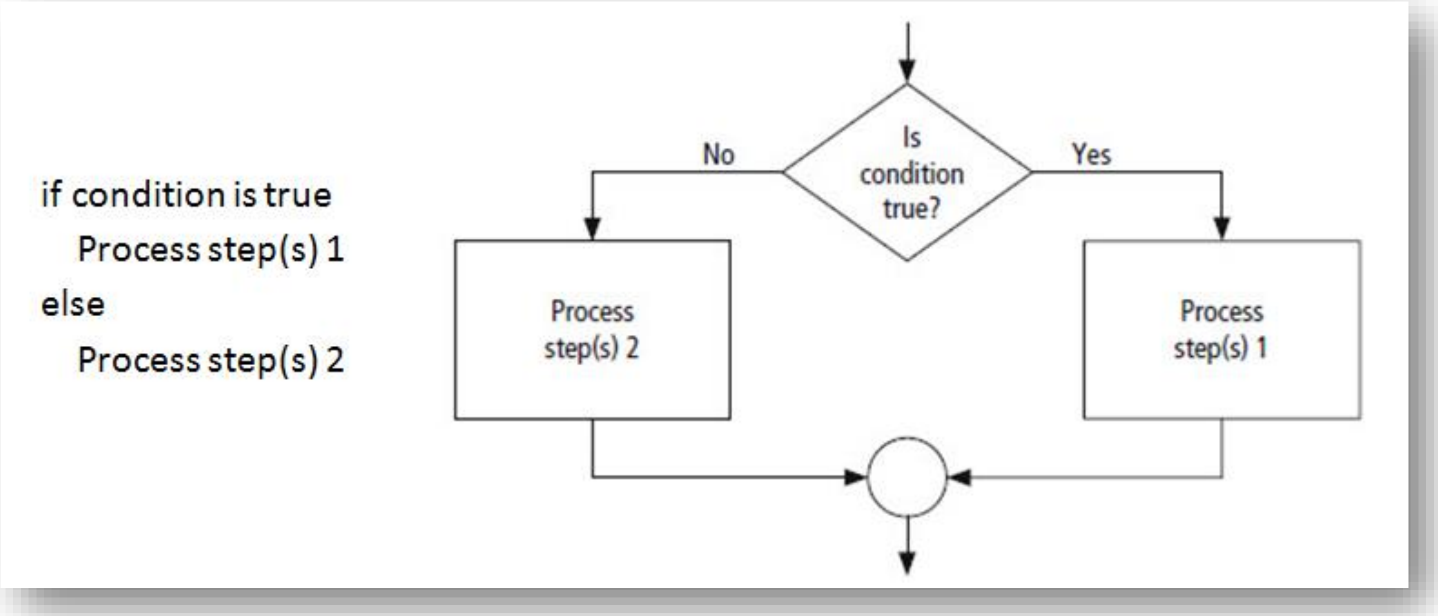
Program: Determine the proper number of stamps for a letter.

```
Read Sheets                                                       (input)
Set the number of stamps to Sheets / 5                       (processing)
Round the number of stamps up to the next whole number       (processing)
Display the number of stamps                                     (output)
```

Pseudocode for the postage stamp problem.

Decision Structure

Pseudocode and flowchart for a decision structure.



```
if condition is true
   Process step(s) 1
else
   Process step(s) 2
```

Direction of Numbered NYC Streets Algorithm

Problem

- Given street number of one-way street in New York City, decide direction of street, either eastbound or westbound.

Discussion

- There is a simple rule to tell the direction of a one-way street in New York City:
  - Even-numbered streets run eastbound.

Direction of Numbered NYC Streets IPO

Input:

- Street number.

Processing:

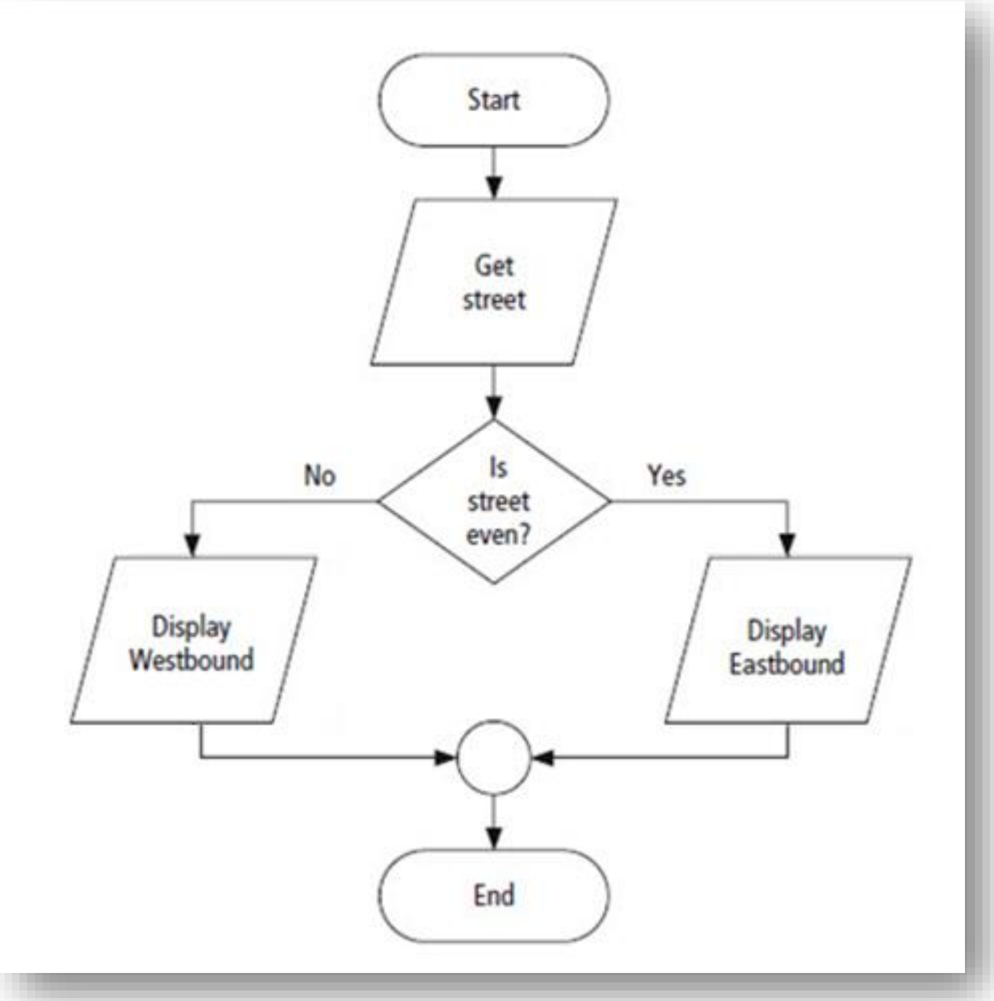
- Decide if the street number is divisible by 2.

Output:

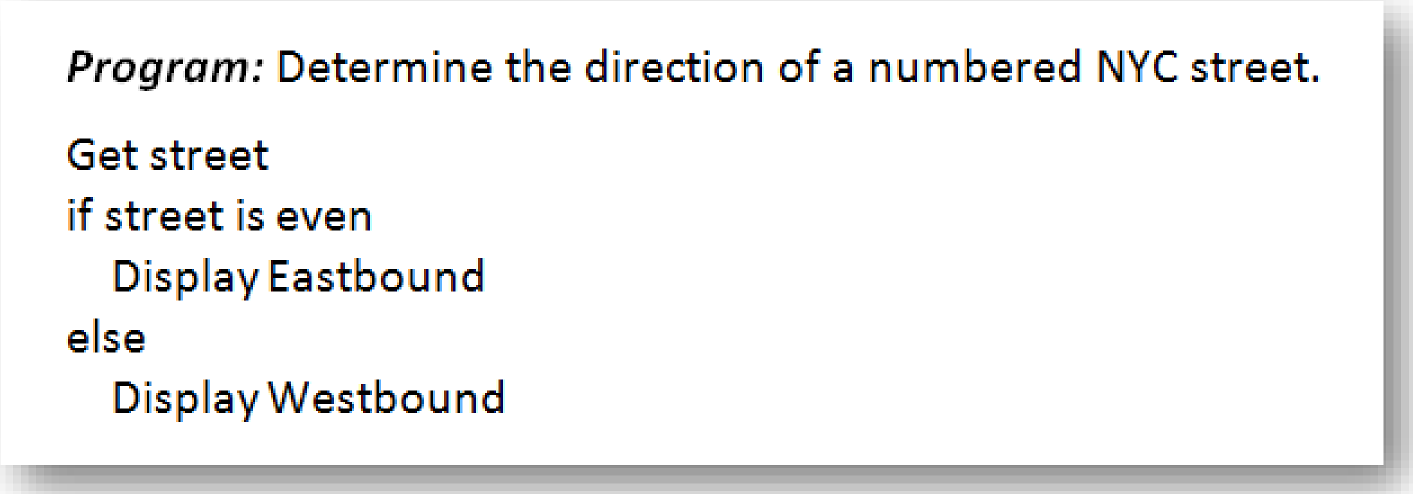
- “Eastbound” or “Westbound”.

Direction of Streets Flowchart

Flowchart for the numbered New York City streets problem.



Direction of Streets Pseudocode



Program: Determine the direction of a numbered NYC street.

```
Get street
if street is even
    Display Eastbound
else
    Display Westbound
```

Pseudocode for the numbered New York City streets problem.

Repetition Structure

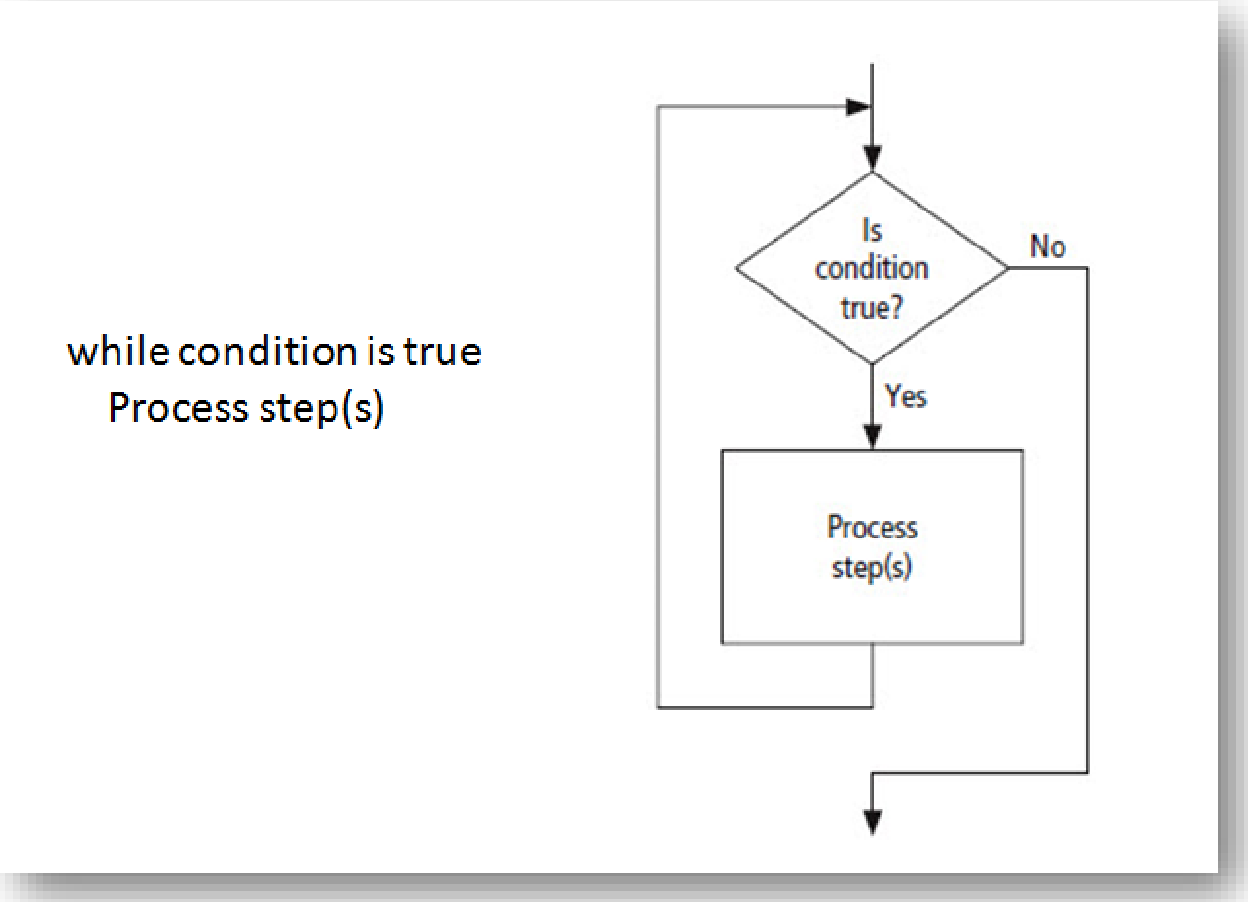
A programming structure that executes instructions many times

- Repetition structure
- Looping structure

Need a test (or condition) to tell when the loop should end

- Check condition before each pass-through loop

Repetition Structure



```
while condition is true
   Process step(s)
```

Pseudocode and flowchart for a loop.

Class Average Algorithm

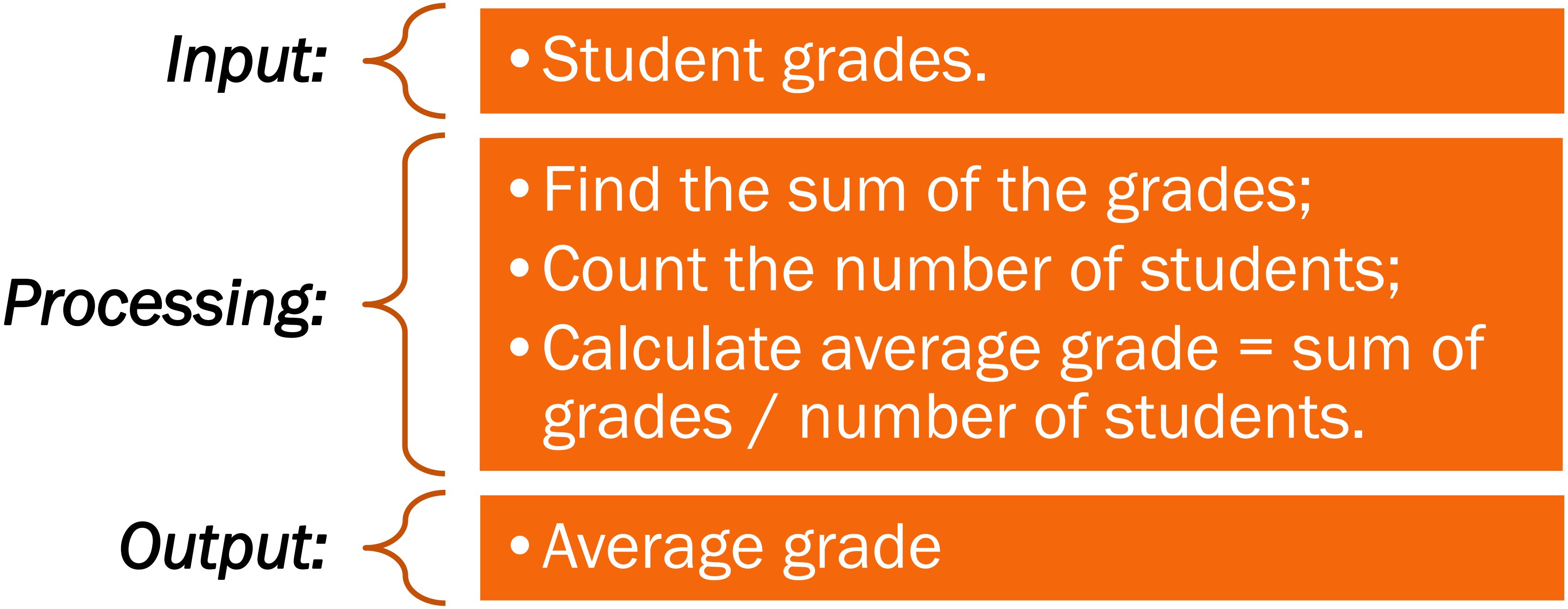
Problem:

- Calculate and report the average grade for a class.

Discussion:

- Average grade equals sum of all grades divided by number of students.
  - Need loop to read and then add (accumulate) grades for each student in class.
  - Inside the loop, we also need to total (count) number of students in class.

Class Average IPO



Input:

- Student grades.

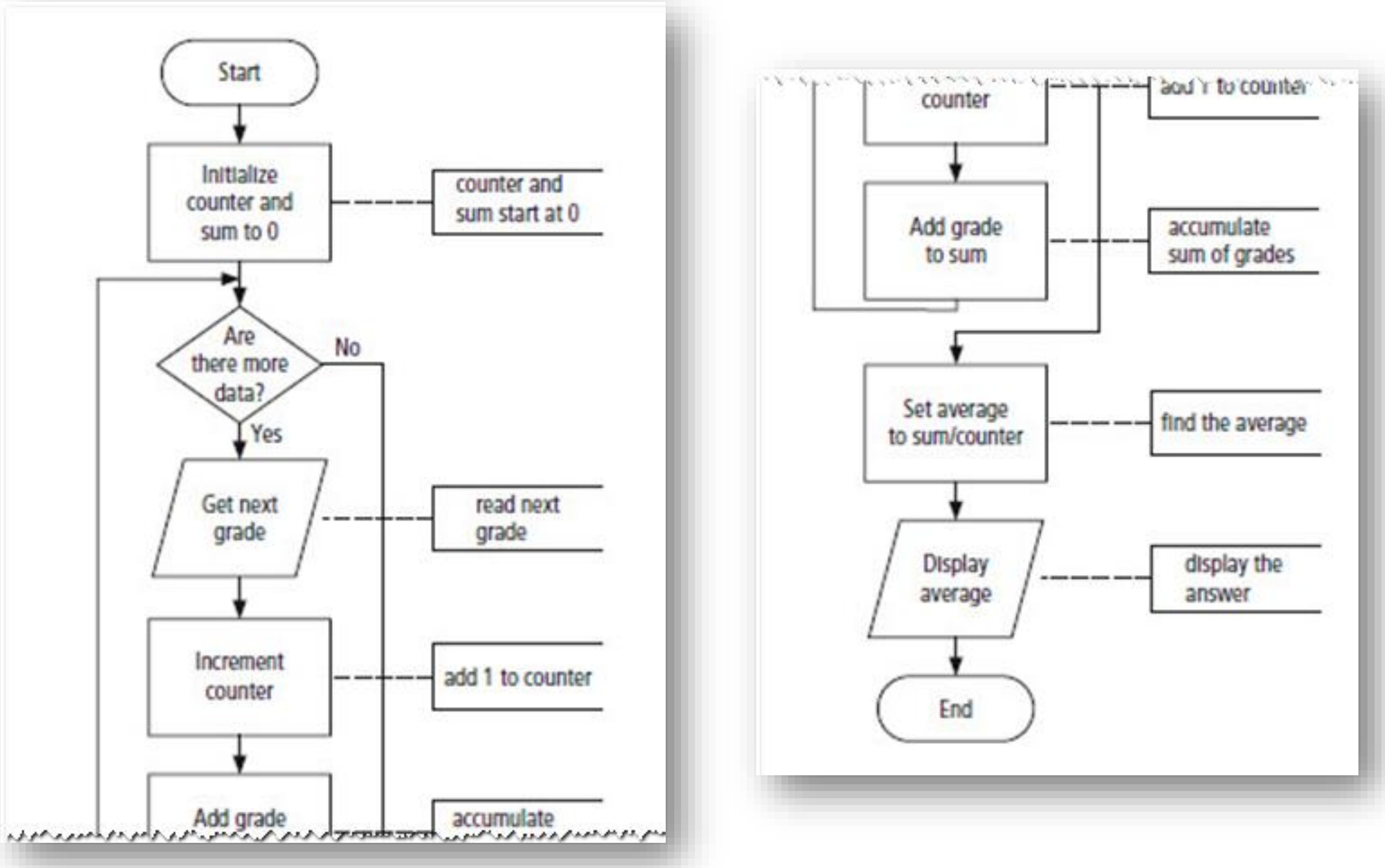
Processing:

- Find the sum of the grades;
- Count the number of students;
- Calculate average grade = sum of grades / number of students.

Output:

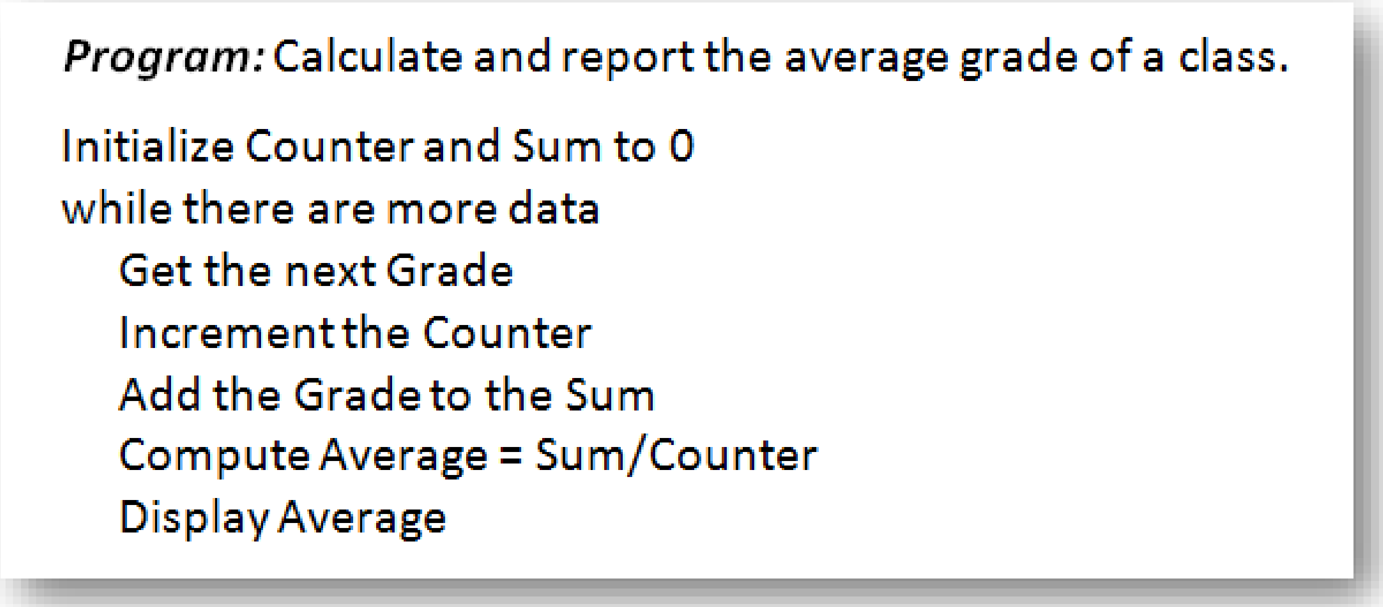
- Average grade

Class Average Flowchart



Flowchart for the class average problem.

Class Average Pseudocode



Program: Calculate and report the average grade of a class.

```
Initialize Counter and Sum to 0
while there are more data
    Get the next Grade
    Increment the Counter
    Add the Grade to the Sum
    Compute Average = Sum/Counter
    Display Average
```

Pseudocode for the class average problem.

Python

Python

- A **general-purpose programming language**.
- Conceived by **Guido van Rossum** in 1980s.
- Named after one of Guido’s favourite comedy sketch show “**Monty Pythons Flying Circus**”

Python

django

PYTORCH

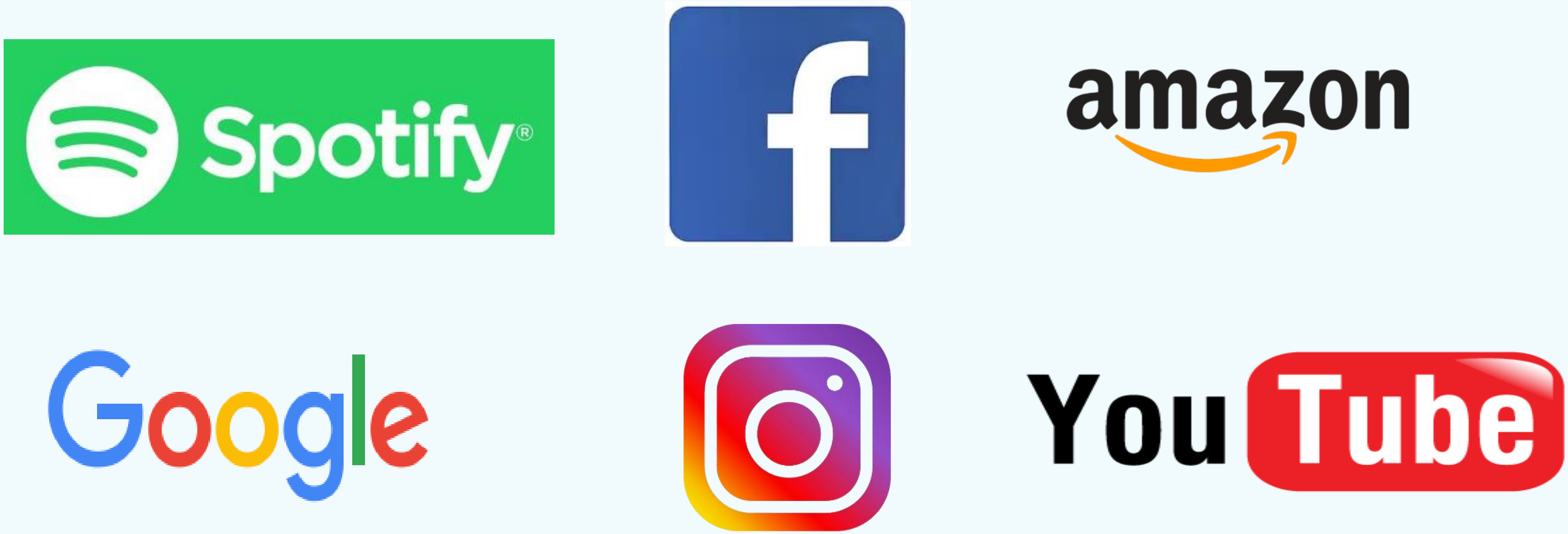
pandas

Why Python is gaining interest in recent years?

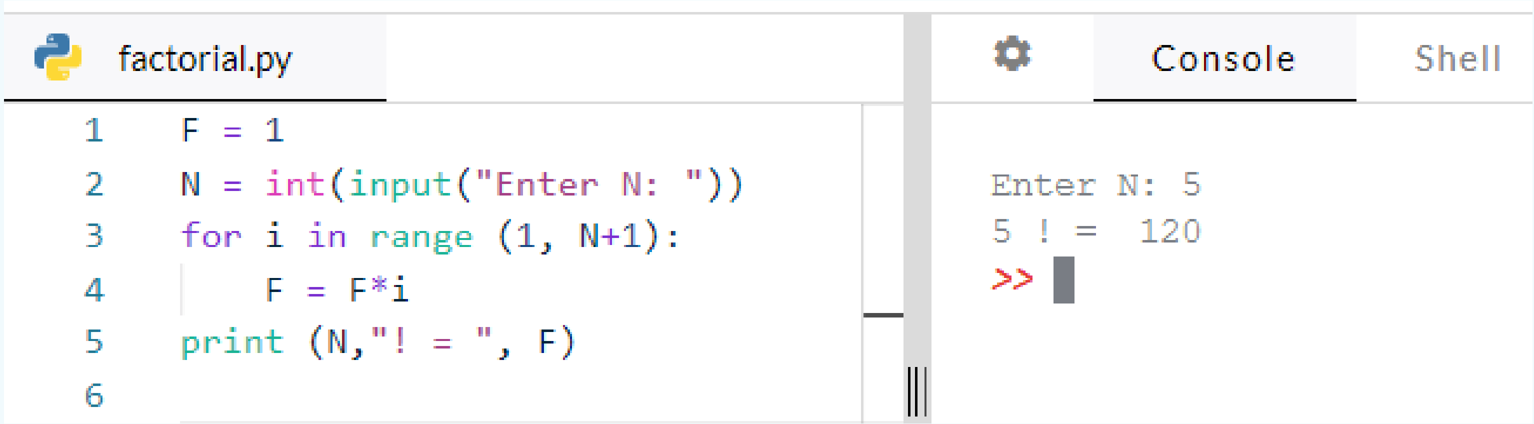
- Simplicity and Speed of Development
- Used by **Data Science** and **Machine Learning** community
- Many libraries that extends the basic features of Python.

Python

Some of the big companies that uses Python



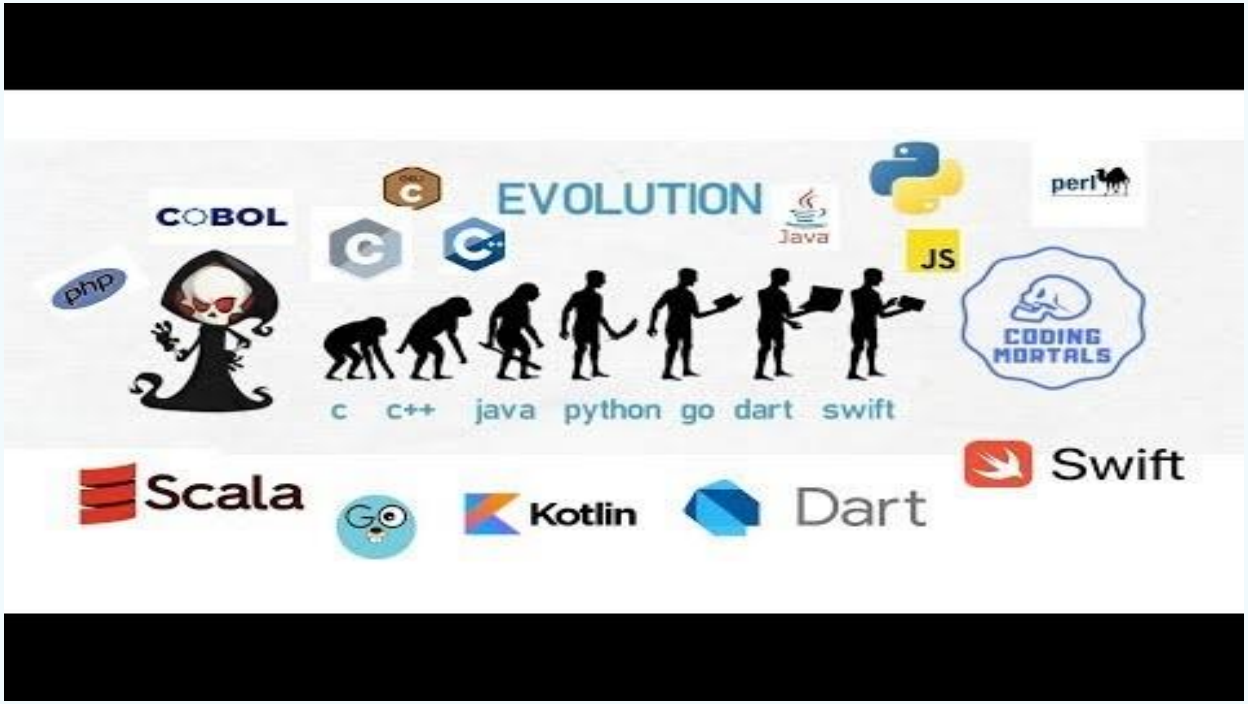
A Python Program for Computing N!



```
F = 1
N = int(input("Enter N: "))
for i in range (1, N+1):
    F = F*i
print (N,"! = ", F)
```

```
Enter N: 5
5 ! =  120
>>
```

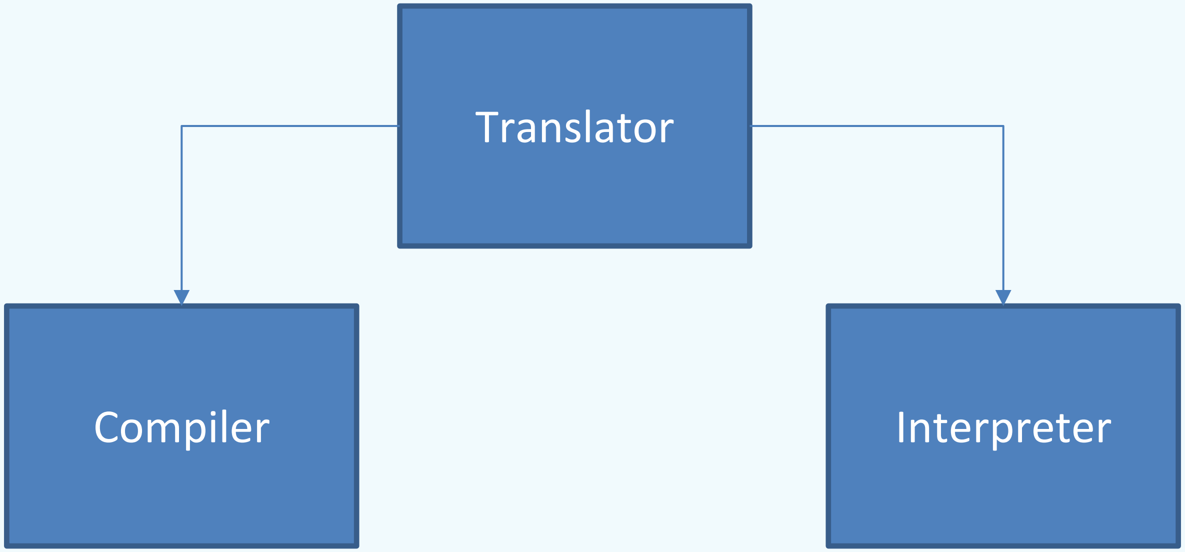
Evolution of Programming Languages



https://www.youtube.com/watch?v=Pn5znSOGHcs&t=92s

Translator

Translator



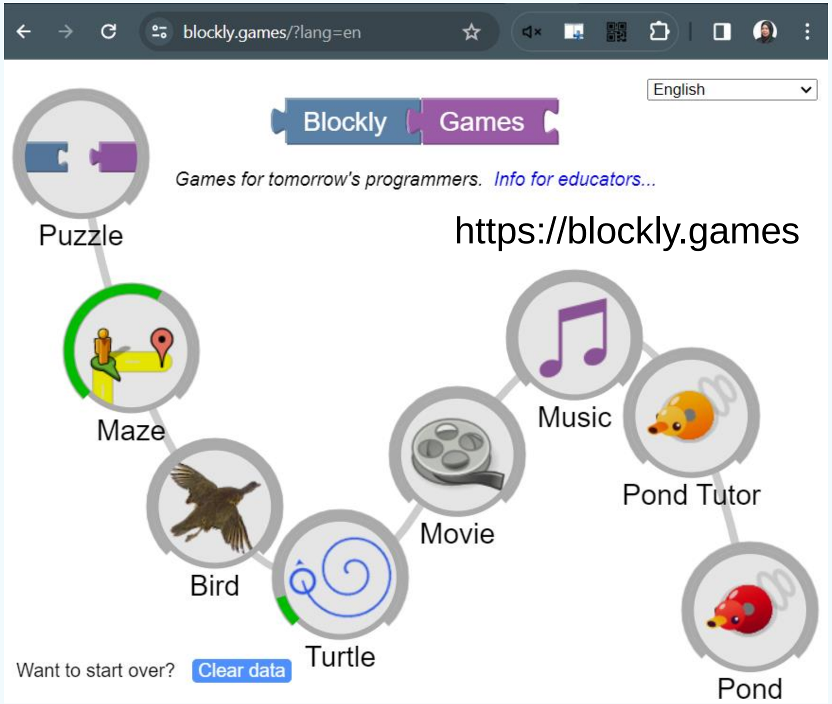
Interpreter

- Reads the program, parses the source codes, and interprets the instructions **on the fly**.
- Example of programming languages that use interpreter
  - Python
  - Javascript
  - PHP

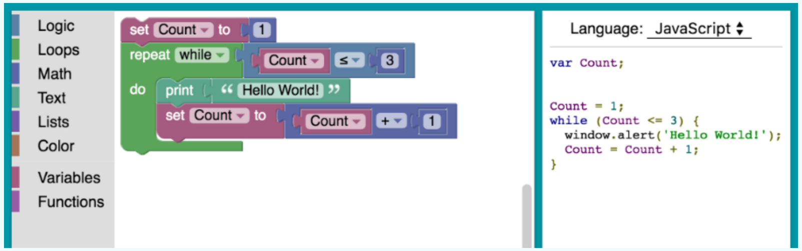
Compiler

- Translate the entire program into machine language, and put the resulting machine language into a file for later execution (e.g. a .exe file or a .dll file).
- Example of programming languages that use compiler
    - C
    - C++

Programming without Coding



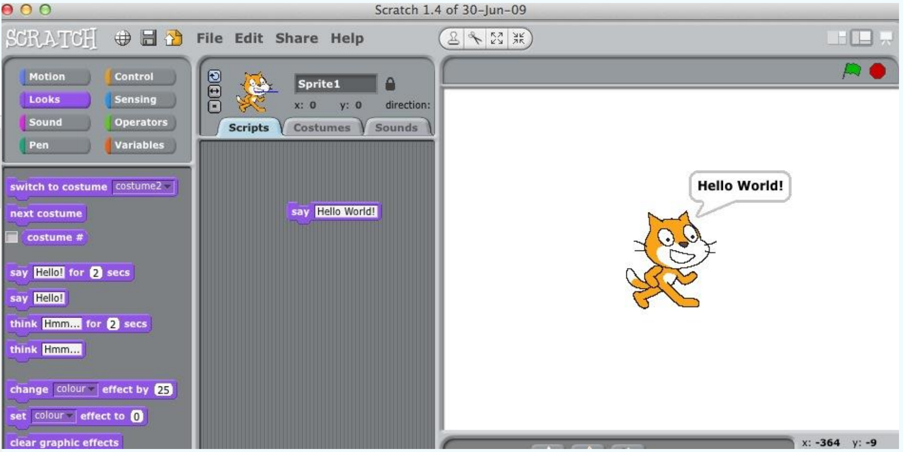
https://blockly.games



```
set Count to 1
repeat while Count ≤ 3
do  print " Hello World! "
    set Count to Count + 1
```

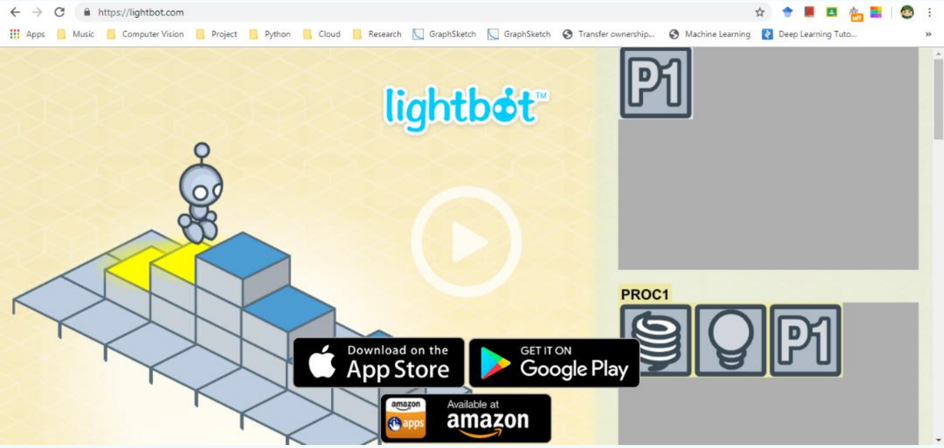
```
var Count;


Count = 1;
while (Count <= 3) {
  window.alert('Hello World!');
  Count = Count + 1;
}
```

https://developers.google.com/blockly



```
say Hello World!
```

https://scratch.mit.edu/



https://lightbot.com/