Introduction to Programming

Paradigms of Programming

Malay Bhattacharyya

Assistant Professor

Machine Intelligence Unit
Indian Statistical Institute, Kolkata

December, 2020

Let's play a game!!!

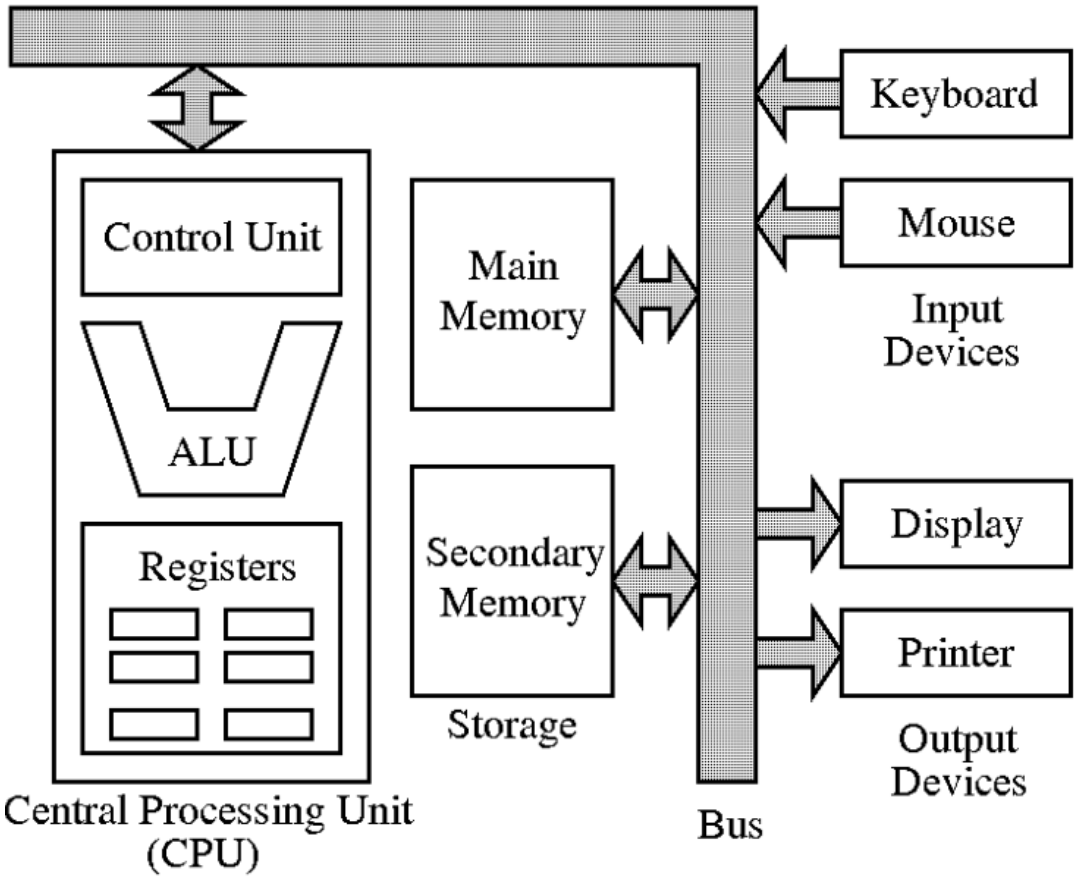
Suppose you are playing a game in turn with the computer. Total n number of sticks are to be picked up in this game. Whoever picks the last one loses the game. Neither the computer nor you can pick up more than 3 sticks at a time. Nobody can skip a turn, i.e. at least one stick is to be picked up in a turn. Write a program to ensure that the computer wins optimally (whenever there is a chance) irrespective of the turn.

What we learnt?

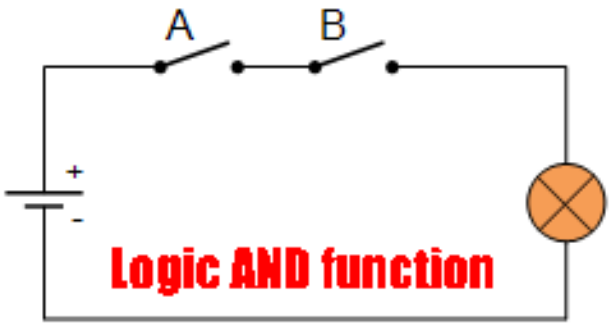
Solving a problem on a computer requires:

1. Understanding the logical actions (strategies) to take.
2. Choosing the best strategy.
3. Finding out the best implementation of the chosen strategy.

A digital computer – The internals

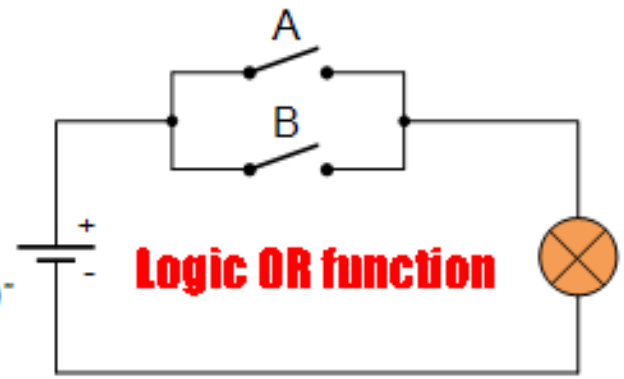


A digital computer – The circuitry



Switch A - Open = "0", Closed = "1"
Switch B - Open = "0", Closed = "1"

Lamp - ON = "1"
Lamp - OFF = "0"



Lamp - ON = "1"
Lamp - OFF = "0"

Switch A - Open = "0", Closed = "1"

A	B	A AND B
0	0	0
0	1	0
1	0	0
1	1	1

A	B	A OR B
0	0	0
0	1	1
1	0	1
1	1	1

A digital computer – Binary arithmetic

Decimal 6.625 equals to 110.101 in binary.

Converting the integer part 6 to binary:

Integer	Operation	Quotient (Integer)	Remainder
6	6 / 2	3	0
3	3 / 2	1	1
1	1 / 2	0 [STOP]	1

↑

Converting the fractional part 0.625 to binary:

Fraction	Operation	Fraction	Integer
0.625	0.625 * 2	0.25	1
0.25	0.25 * 2	0.5	0
0.5	0.5 * 2	0 [STOP]	1

↓

Finding the maximum

Given three distinct integers as input, find out the maximum of them and show the same as output.

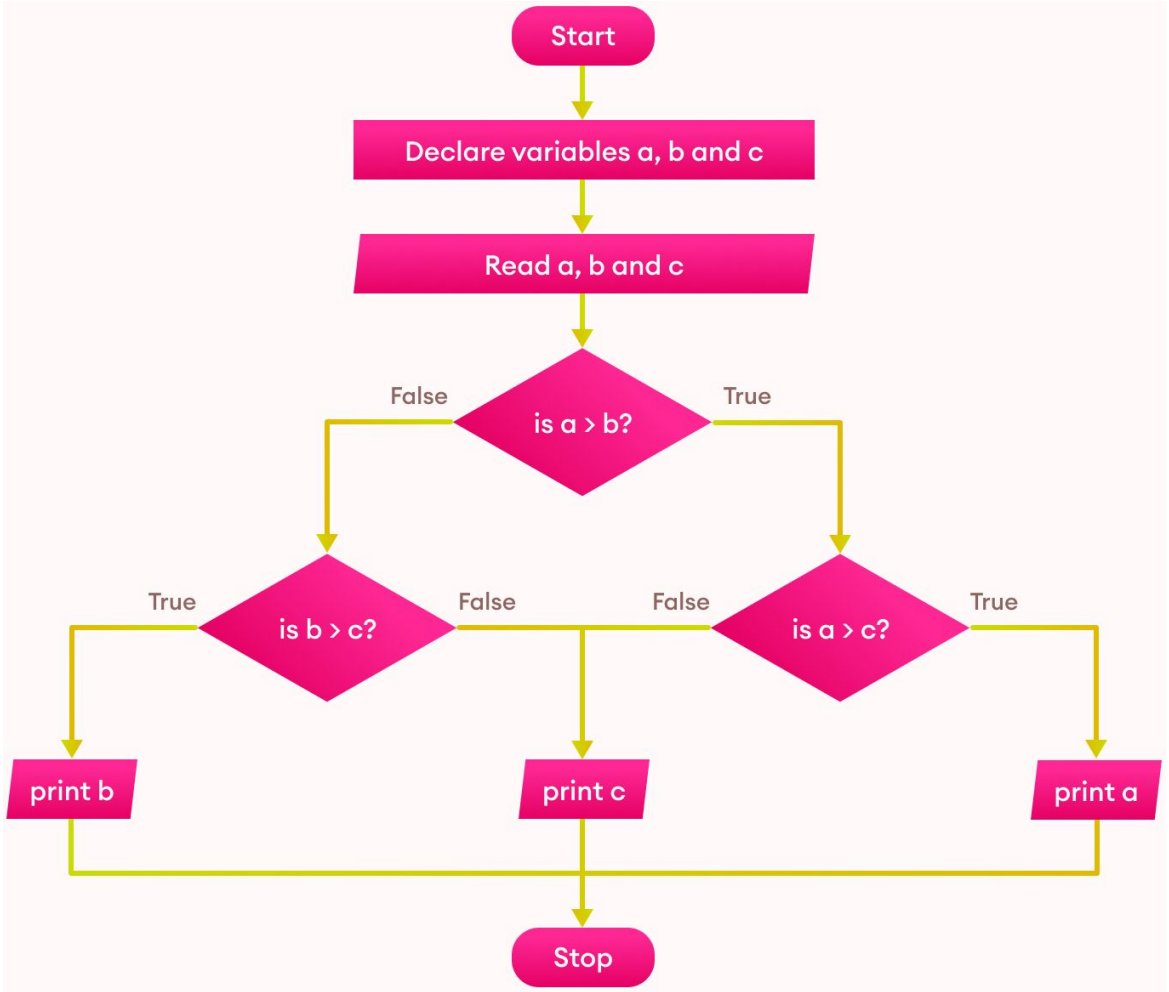
Choosing the maximum – A naive approach

```
Inputs: a, b, c  // All are distinct values
if a > b and a > c then do // 2 comparisons
        Output a
end if
if b > a and b > c then do // 2 comparisons
        Output b
end if
if c > a and c > b then do // 2 comparisons
        Output c
end if
```

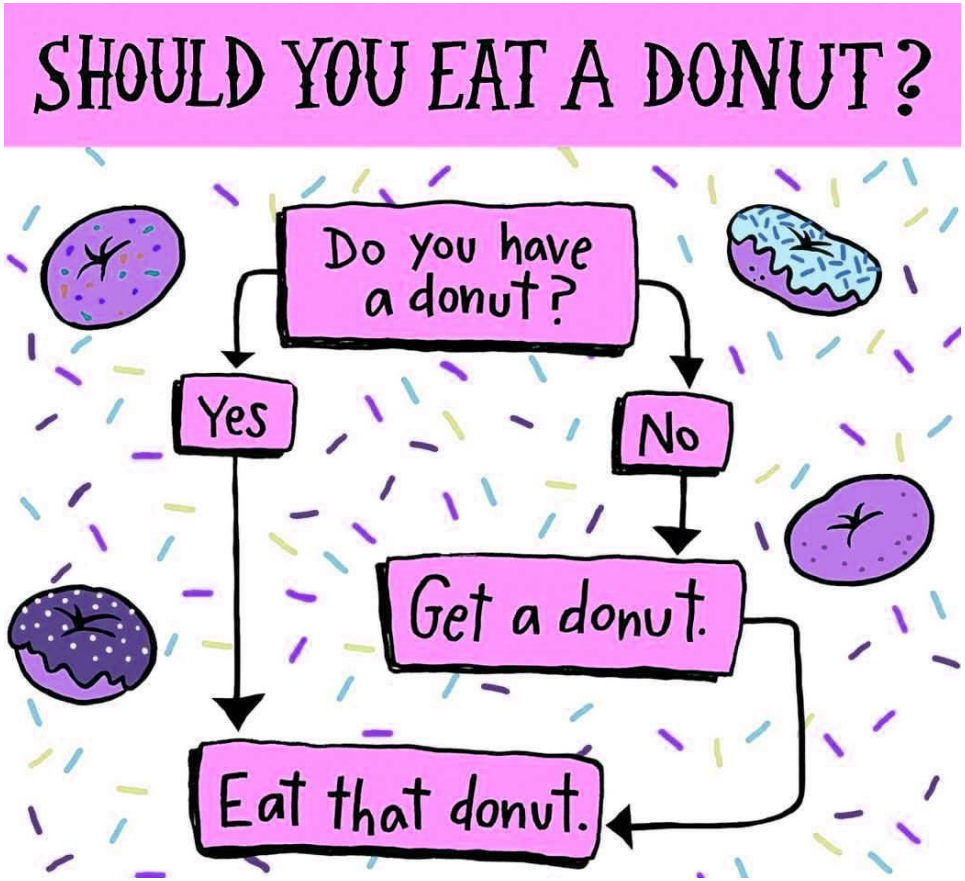
Finding the maximum – A better approach

```
Inputs: a, b, c // All are distinct values
if a > b then do // 1 comparison
    if a > c then do // 1 comparison
        Output a
    end if
    else do
        Output c
    end else
end if
else do
    if b > c then do // 1 comparison
        Output b
    end if
    else do
        Output c
    end else
end if
```

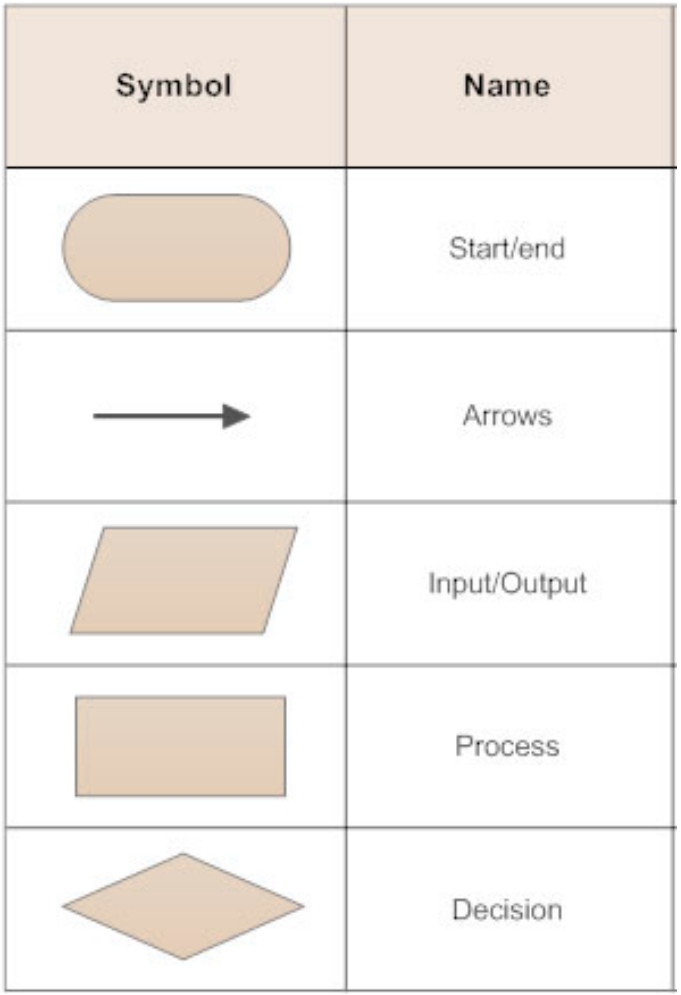
Finding the maximum – Using flowcharts



Decision making is everywhere

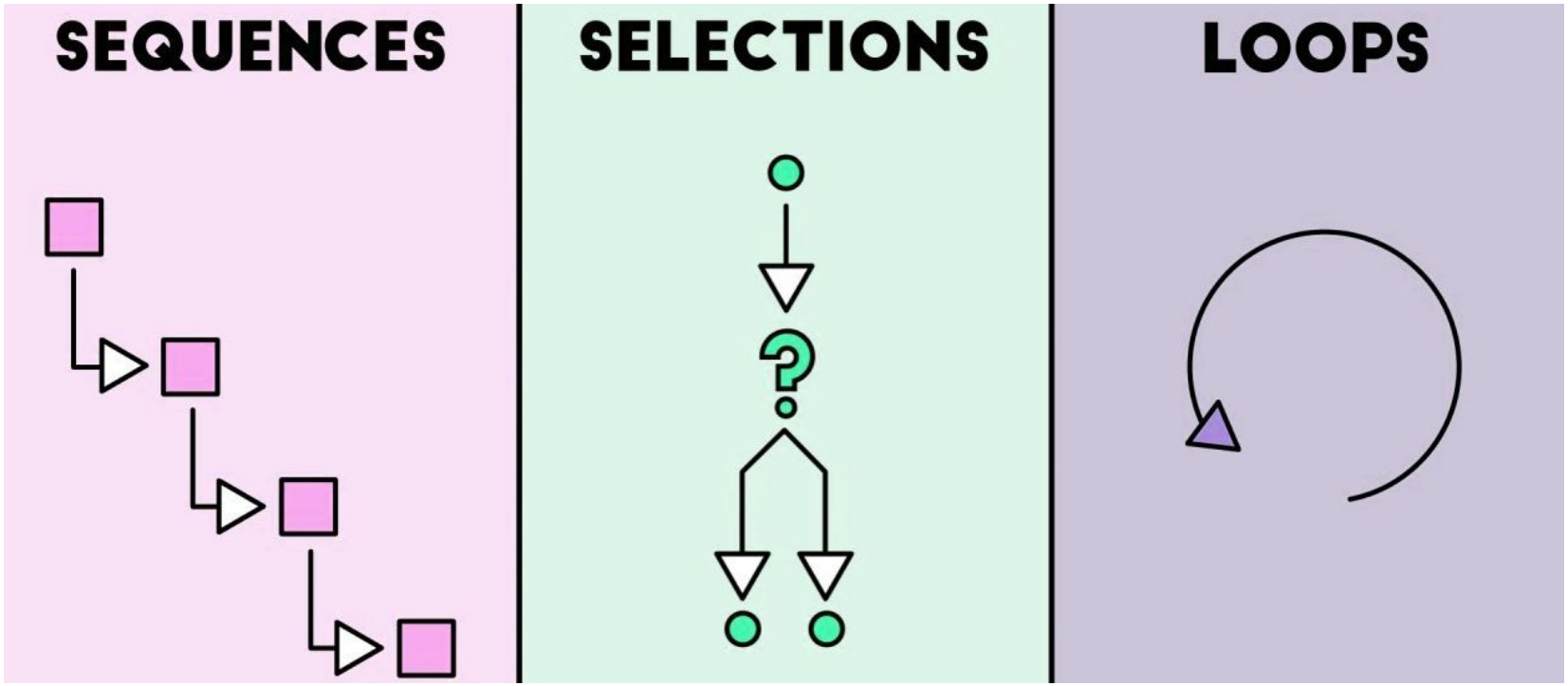


Basic symbols in a flowchart

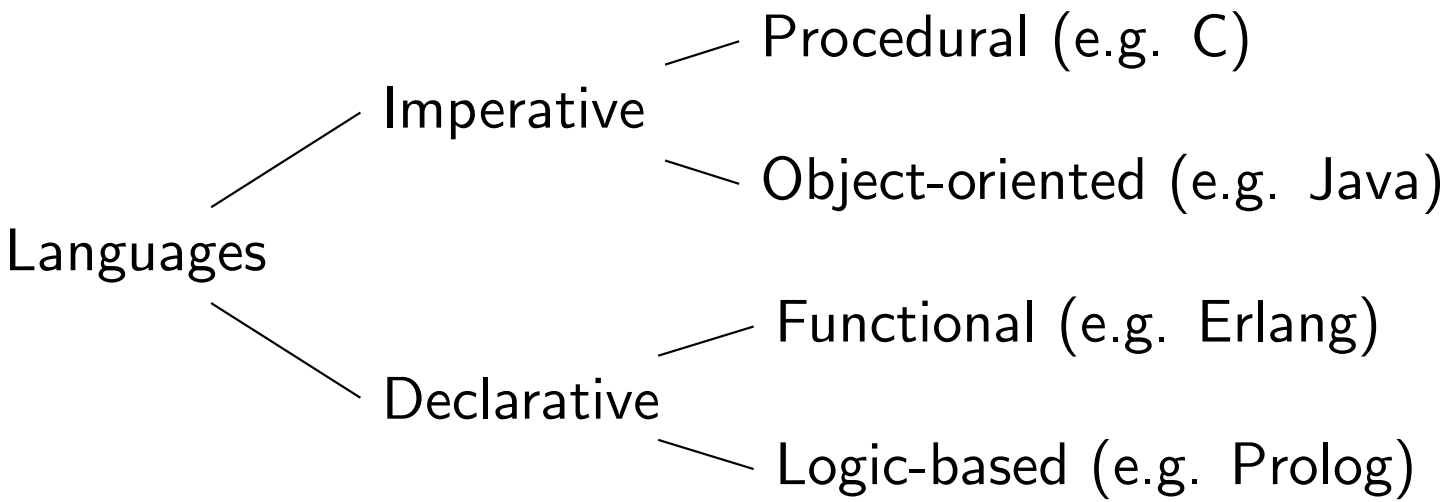


Symbol	Name
	Start/end
	Arrows
	Input/Output
	Process
	Decision

Types of control flows



Types of programming languages



Imperative vs declarative programming language

Imperative	**Declarative**
1. It tells the computer what steps to obtain a result 2. It uses statements that change a program's state	1. It tells the computer what result they want 2. It expresses the logic of a computation without describing its control flow

Procedural vs object-oriented programming language

Procedural	**Object-oriented**
1. It takes a top-down approach to divide a program into smaller parts (known as functions)	1. It uses objects to represent everything in a program
2. It treats data and methods separately	2. It encapsulates data and methods together
3. It is less secure	3. It is more secure

Functional vs logic programming language

Functional	**Logic**
1. Program evaluation is one-way	1. Program evaluation is two-way
2. It uses a virtual machine on which functions operate	2. It performs applies query on a special domain
3. It avoids state and mutable data	3. It extracts knowledge from basic facts and relations

Compiler vs interpreter

Compiler	**Interpreter**
1. It processes the entire program at a time	1. It processes a single instruction at a time
2. It transforms the source code into machine readable instructions	2. It interprets the source code into executables
3. It generates intermediate object code, hance memory requirement is more	3. It does not generate intermediate object code, hence memory requirement is less
4. It is relatively faster	4. It is relatively slower