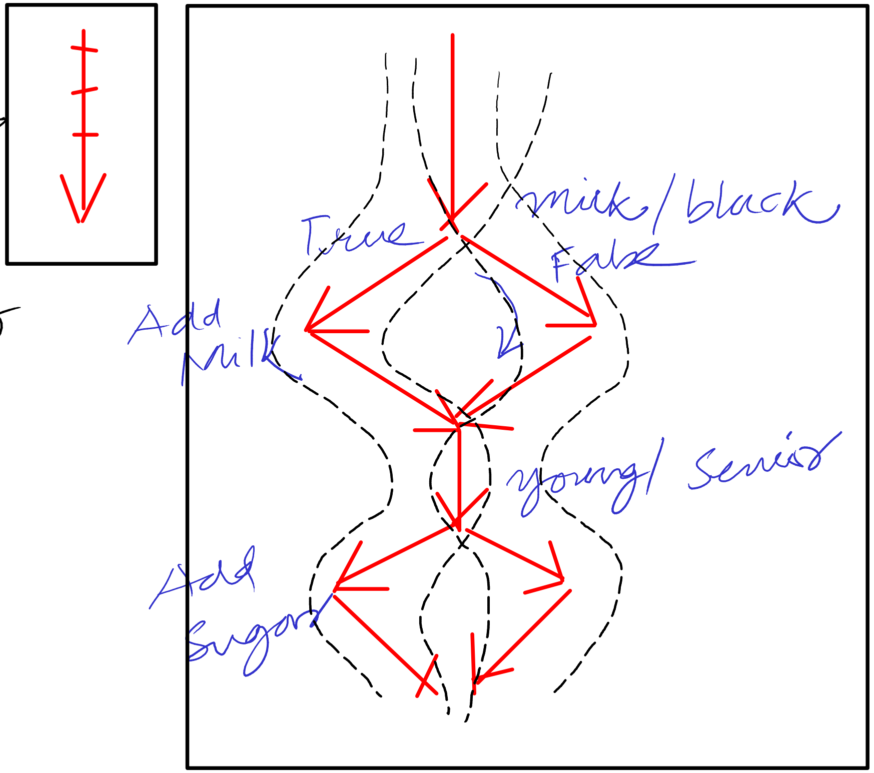


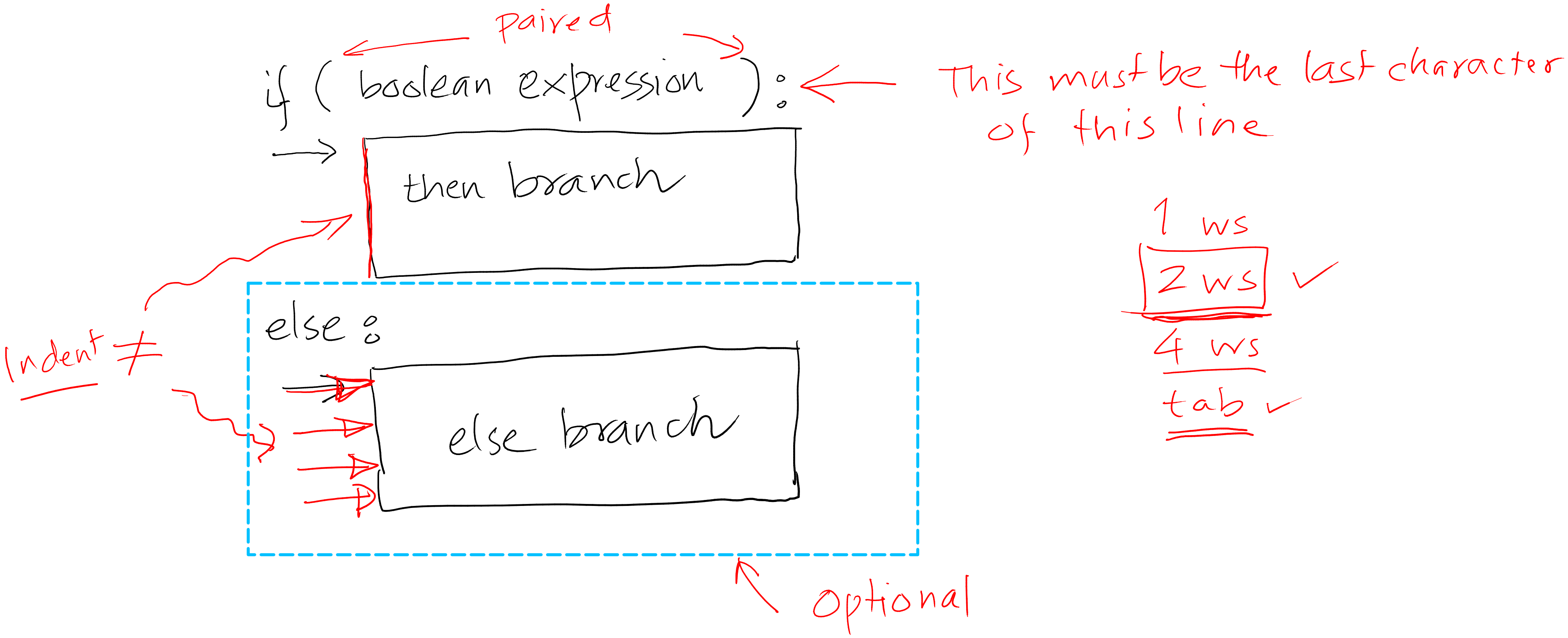
Without branching

With branching.

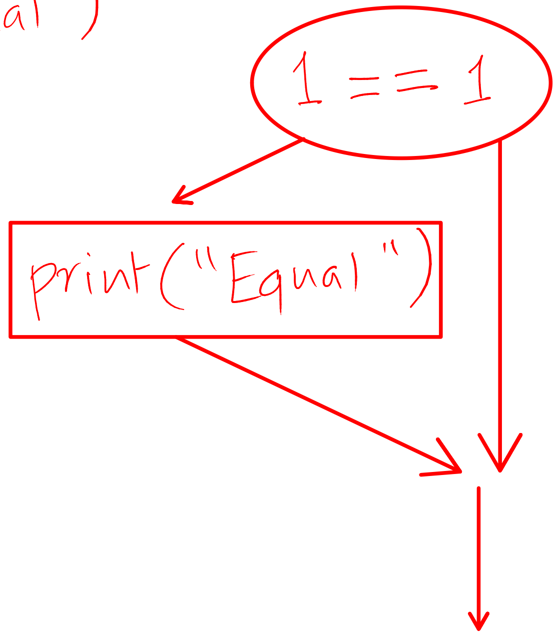
Flowchart

Flow graph

Control flow graph.

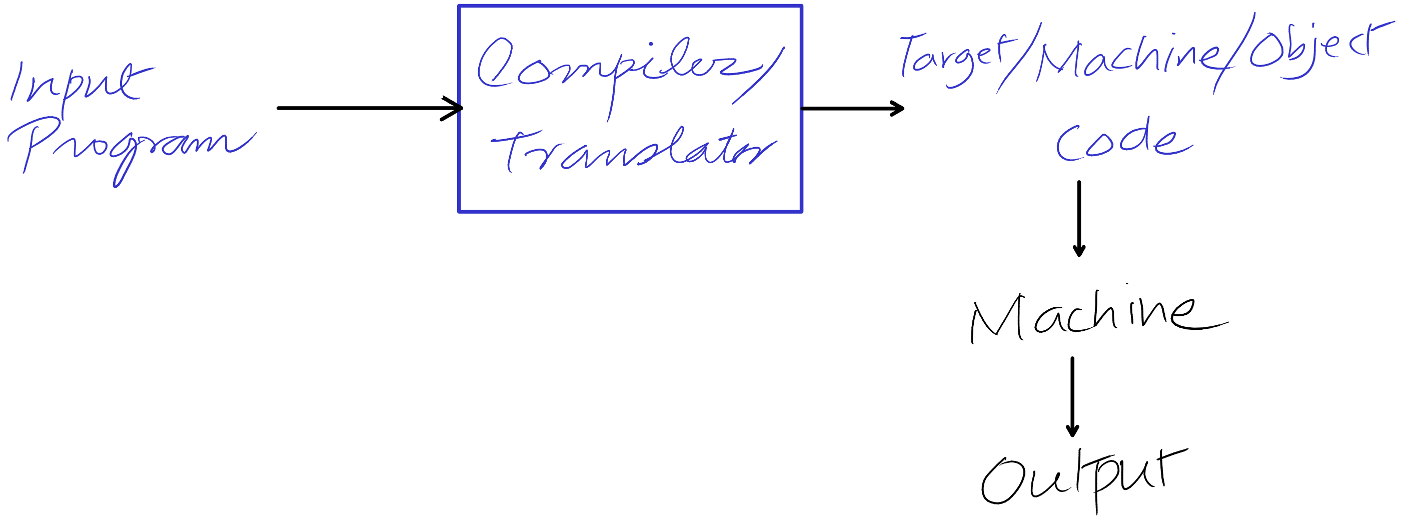

paired
if (boolean expression):
This must be the last character of this line
then branch
else :
else branch
Indent #
optional
1 ws
2 ws ✓
4 ws
tab ✓

```
if (1==1):
    print("Equal")
```

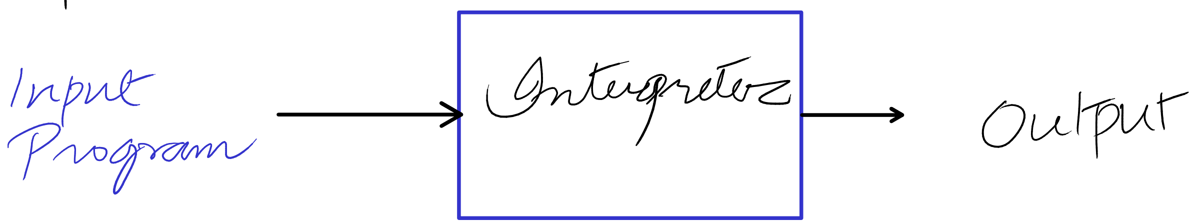


Compilation v/s Interpretation

Compiled PL :



Interpreted PL :



Repetition

Problem decomposition

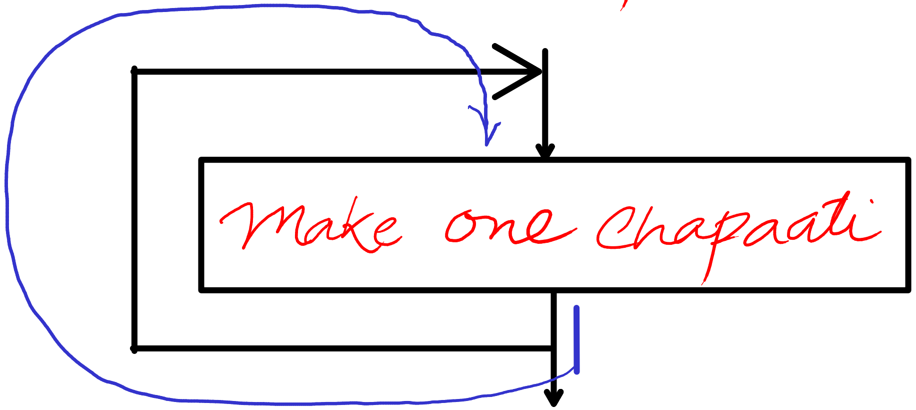
A journey of a thousand miles begin with ...

— R.F.

Make Chapaati

make 10 Chapaatis

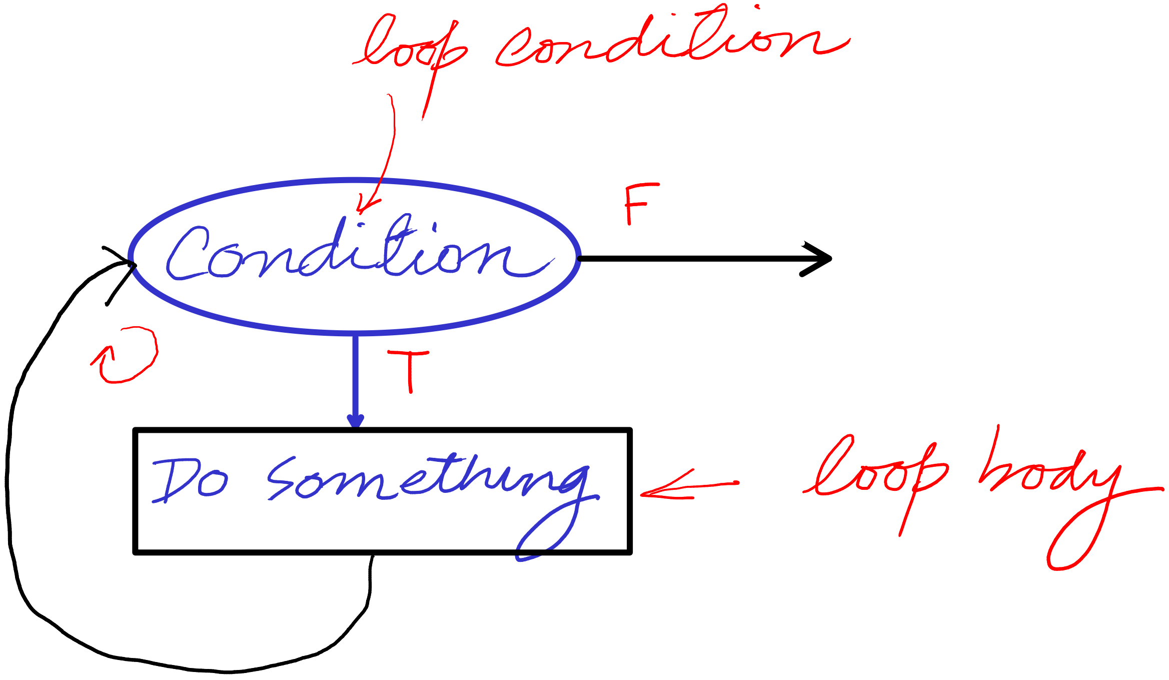
```
Repeat 10 times
    Make one Chapaati
```



RAMU.

LOOPS

1. WHILE



1. Iteration : Execution of loop body ONCE.

IL is bad?

✓ For pure Computation — YES

✓ Real life programs — NOT ALWAYS

Compiler, search

Algorithms — Inputs → Output

Run to completion programs. RTC

Examples of program which run forever.

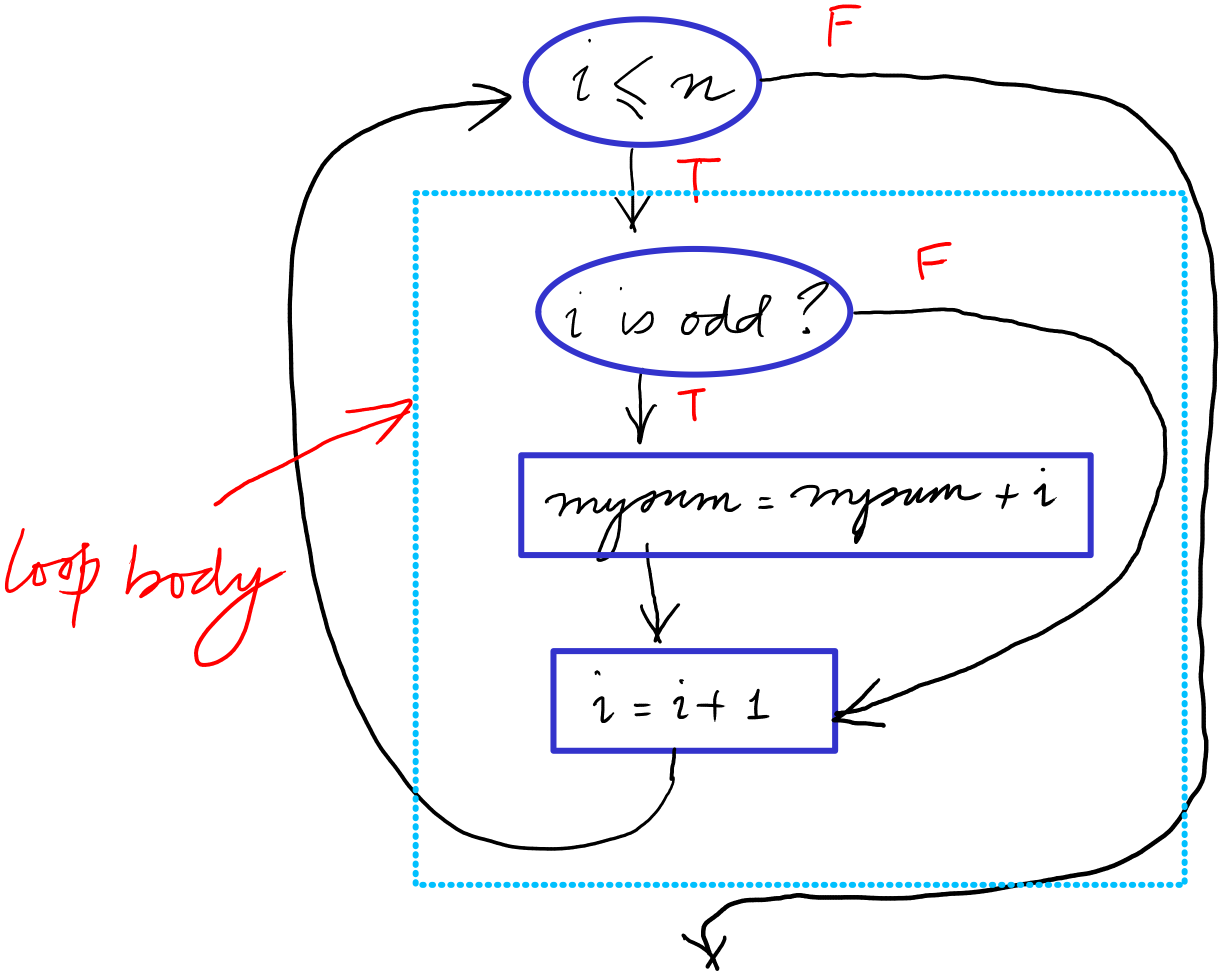
- Servers – CT server
- GUI – Chrome, Xournal++
- Embedded software

REACTIVE SYSTEMS

Graphical User Interface

while(True):

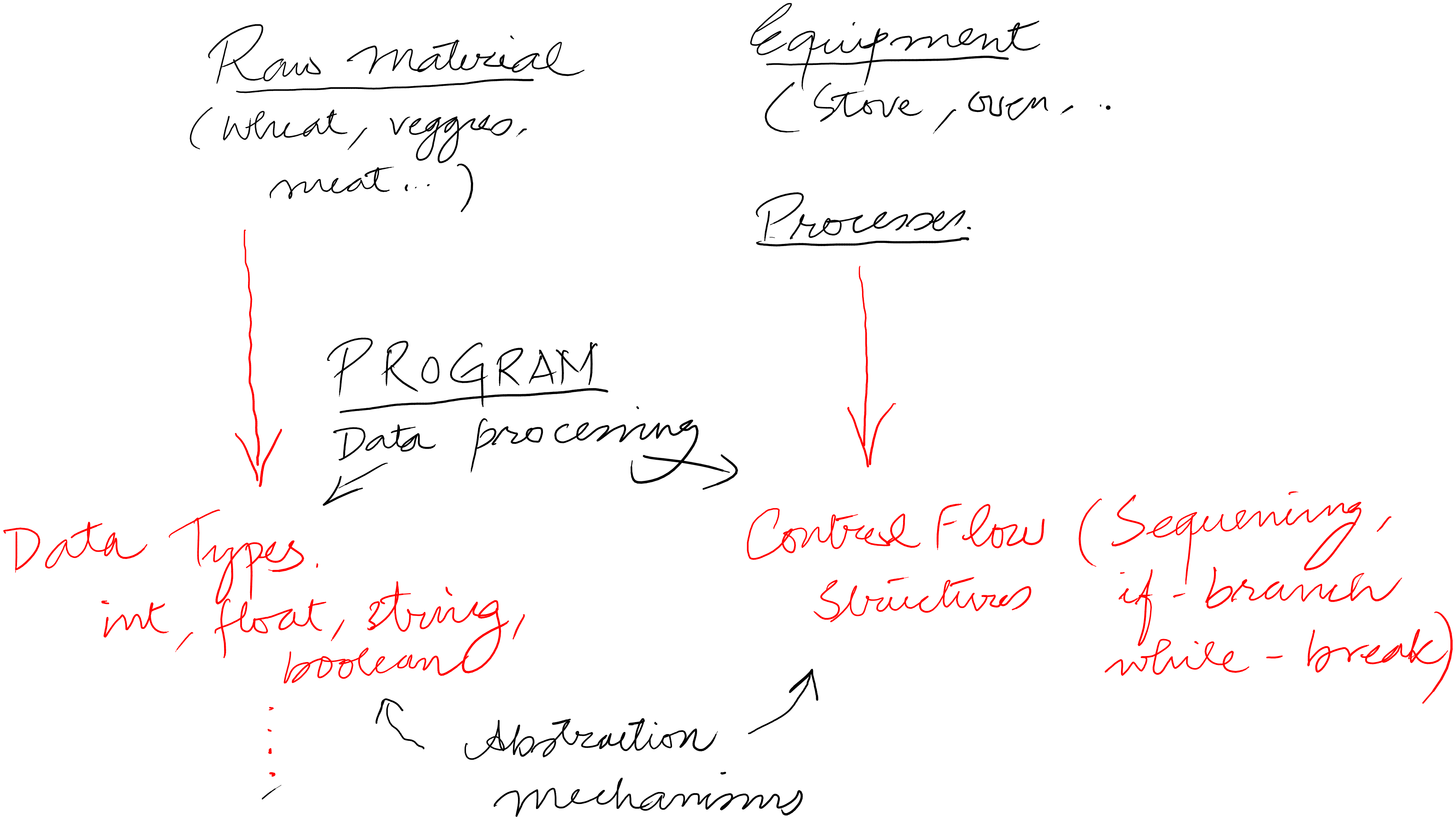
≡


i ≤ n
F
T
i is odd ?
F
T
mysum = mysum + i
i = i + 1
loop body

Program begin to become complex

- Not because there are lots of features in a prog. lang.
- But because these few features can interact with one another in arbitrary ways.

FOOD RECIPES



Raw material

(wheat, veggies, meat...)

Equipment

(Stove, oven, ..

Processes.

PROGRAM

Data processing

Data Types.

int, float, string, boolean,

⋮

Control Flow (Sequencing, if-branch while-break)
Structures

Abstraction mechanisms

COLLECTION

Lists

Python — Dynamically typed

C, C++, Java — Statically typed

Heterogeneous lists — Elements can be of diff. types.

Mutable — list elements can change

Type Checking — DTPL — Runtime
STPL — Compile time

Functional

len([1,2,3])

3

l[1]

= 2

non-functional

l.append(3)

l

[1,2,3,3]