

情報基礎B

Lecture 5: Complex formulas

Matias Korman

Tohoku University Graduate School of Information Sciences
System Information Sciences
Design and Analysis of Information Systems

PLAYING WITH SPREADSHEETS

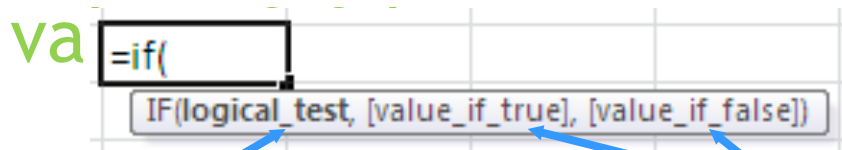
Comparing data

- Comparison
 - A15 = $10 > 3$
 - B15 = $10 < 3$
 - C15 = $(10 * 2) < 3$
 - D15 = $C1 > C2$
 - E16 = $C7 > E2$
- Result is “TRUE” or “FALSE”

Interesting programs?

- Branching
 - Action depends on something else
- Implemented with IF function
 - IF(criterion, action1, action2)
 - Proceed action1 when the criterion is true, otherwise proceed action2
 - =IF(logical_test, value_if_true,

va =if(



```
=IF(logical_test, [value_if_true], [value_if_false])
```

Logical test or cell number

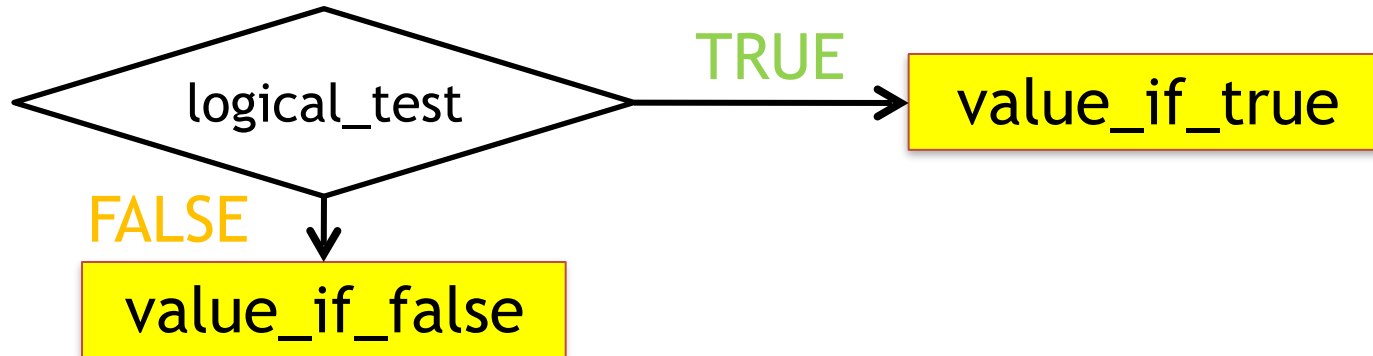
Characters with “” or numbers

Simple Program

- Operators used in logical tests
 - $A=B$ A is equal to B
 - $A>B$ A is larger than B
 - $A<B$ A is smaller than B
 - $A\leq B$ $A=B$ or $A<B$
 - $A\geq B$ $A=B$ or $A>B$
 - $A\neq B$ A is not equal to B

Simple Program

=IF(logical_test, value_if_true, value_if_false)



=IF(logical_test, 1, 0)

•A16 =IF(A15, 1, 0)

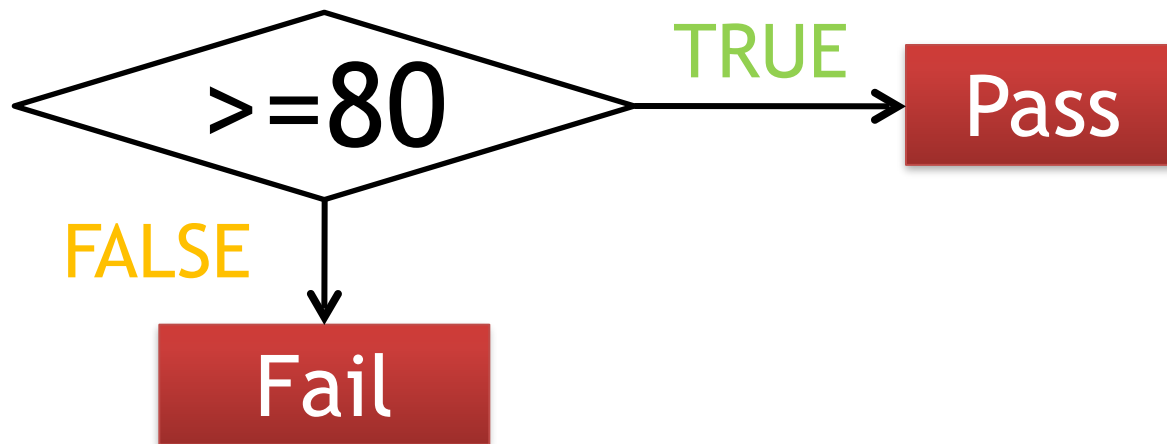
=IF(logical_test, "string", "string")

•B16 =IF(A15, "True", "False")

•C16 =IF(C1>C2, "Correct", "Wrong")

Grading a student

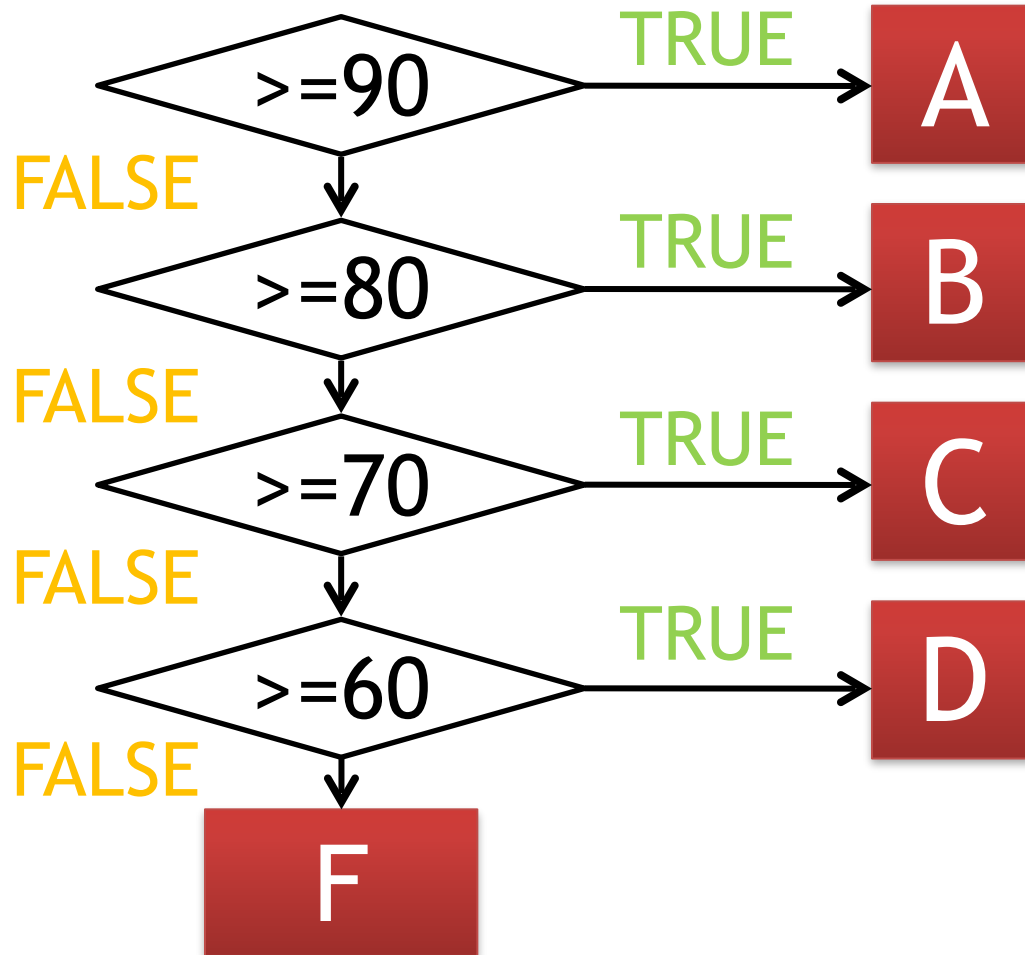
- D16 =IF(A1>=80, “Pass”, “Fail”)



Exercise 1: Coarse grading

- Use table from last week
 - download from <https://goo.gl/Kks4Bh>
- Program a grading system on excel which outputs “Pass” or “Fail”
 - Pass: if Score of Japanese, English, and Math is more than 80
 - Fail: otherwise
 - Add “Result” on G1
 - Same entry for positions G2 to G21

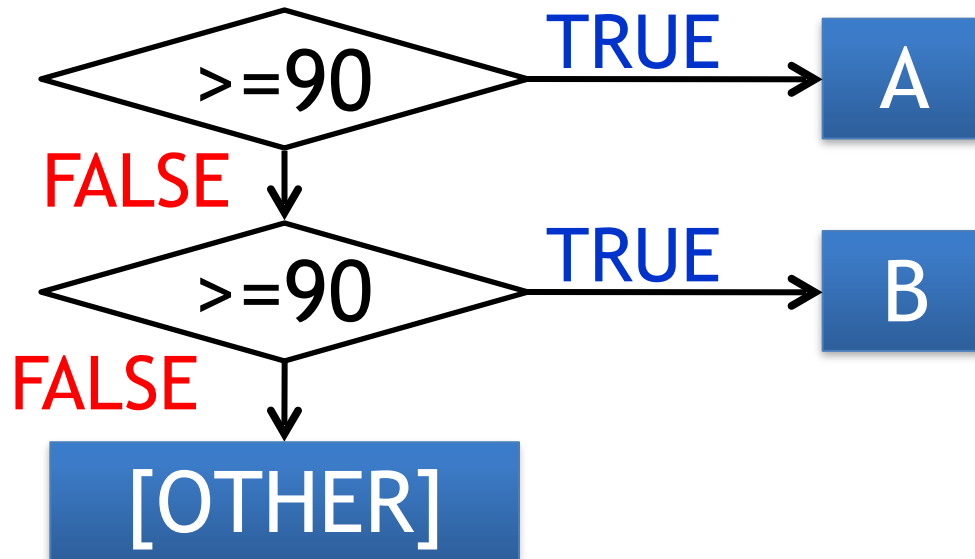
Branching



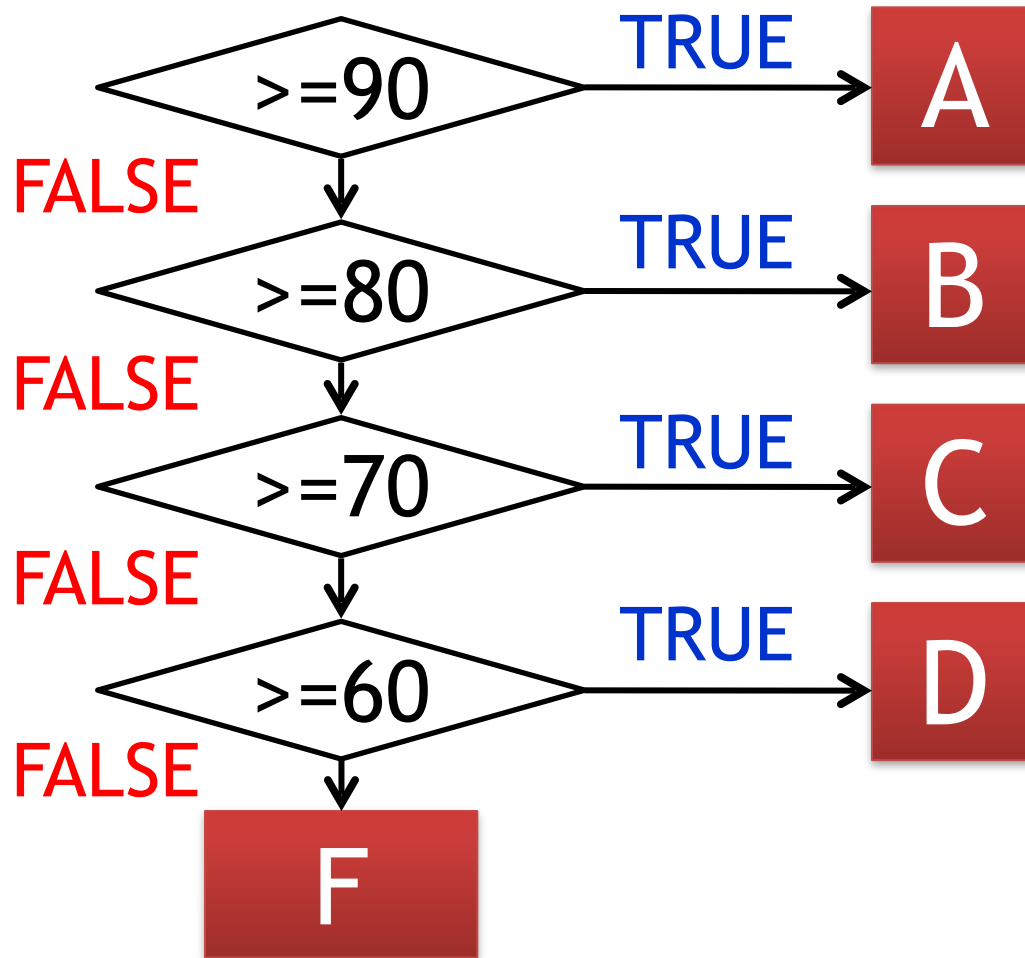
Branching

- Three scores by nesting “IF” operations

```
=IF(B2>=90, “A”, IF(B2>=80, “B”, ”OTHER”))
```



Do we need to stop at 3?



Writing in Excel

- Grade
 - A 100 > Score >= 90
 - B 90 > Score >= 80
 - C 80 > Score >= 70
 - D 70 > Score >= 60
 - F 60 > Score

```
=IF(B2>=90, "A",  
IF(B2>=80, "B",  
IF(B2>=70, "C",  
IF(B2>=60, "D", "F"))))
```

Exercise 2: fine grading

- Make a grading system which outputs “A”, “B”, “C”, “D” or “F” for each Subject
 - A $100 > \text{Score} \geq 90$
 - B $90 > \text{Score} \geq 80$
 - C $80 > \text{Score} \geq 70$
 - D $70 > \text{Score} \geq 60$
 - F $60 > \text{Score}$
- Apply same method to all students
- Add “Japanese”, “English” and “Math” header each on G1, H1 and L1

Counting

- How many students got A on Japanese?

```
=COUNTIF(range, criteria)
```

```
=COUNTIF(G2:G21, "A")
```

Exercise 3: counting

- Count numbers of students that got each grade in each subject

	Japanese	English	Math
A	6	4	5
B	4	7	6
C	3	4	2
D	3	1	2
F	4	4	5

Exercise 3: counting

- Count numbers of students that got each grade in each subject

	Japanese	English	Math
A	6	4	5
B	4	7	6
C	3	4	2
D	3	1	2
F	4	4	5
SUM	20	20	20

- Sanity check: totals in each course

CHARTS

Let's make pretty drawings!

- Bar chart

The screenshot shows the Microsoft Excel interface with a data table. A red circle labeled '1' highlights the selected data range (rows 2-6, columns L-M). A red circle labeled '2' highlights the 'Chart Wizard' button in the 'Insert' tab of the ribbon.

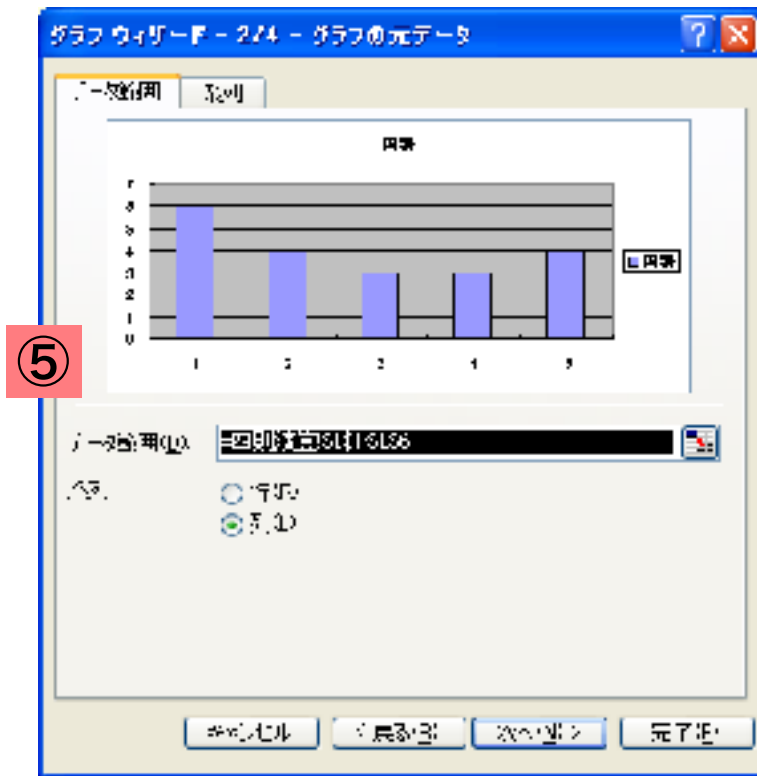
	J	K	L	V	W	U
1			国語	英語	数学	
2		秀	6	4	5	
3		優	4	7	6	
4		長	3	4	2	
5		可	3		2	
6		不可	4	4	5	
7						
8						
9						
10						
11						

1. Select data range
2. Click “Chart Wizard”
3. Select a chart type

The screenshot shows the 'Chart Wizard' dialog box. A red circle labeled '3' highlights the 'Chart type' list on the left, where '柱状図' (Bar chart) is selected. The right side of the dialog shows various chart style options.

Chart

- Bar chart



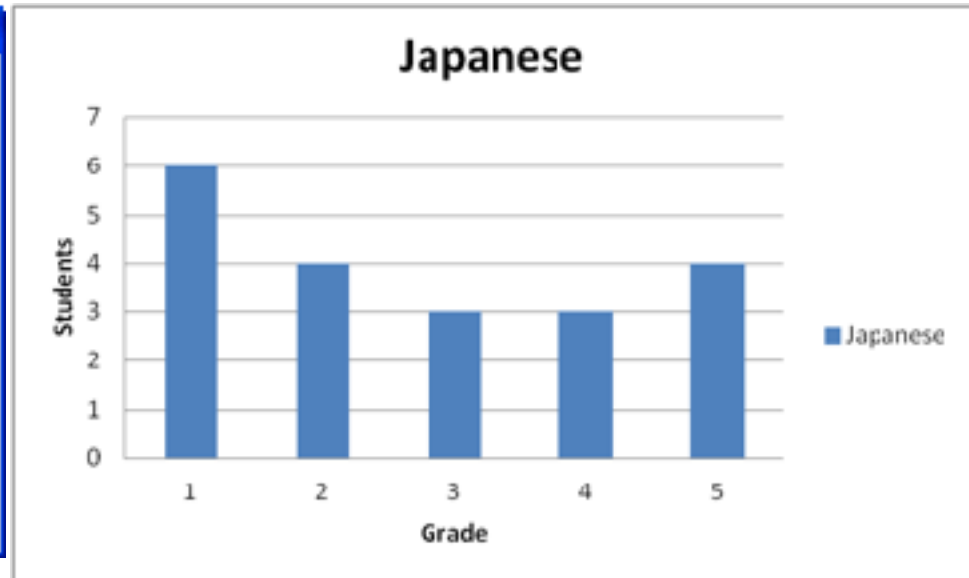
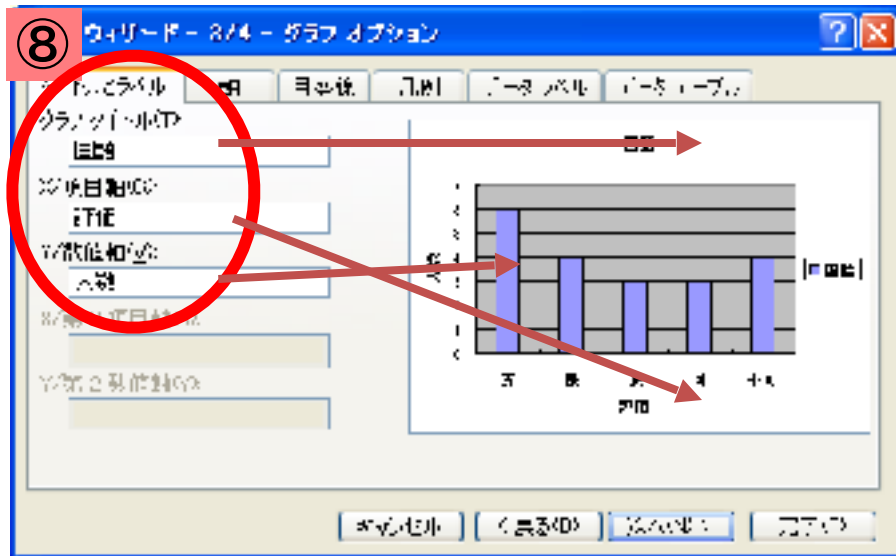
5. Data range is shown. Or enter a range directly



6. Select "Series"
7. Set Category (X) axis labels

Chart

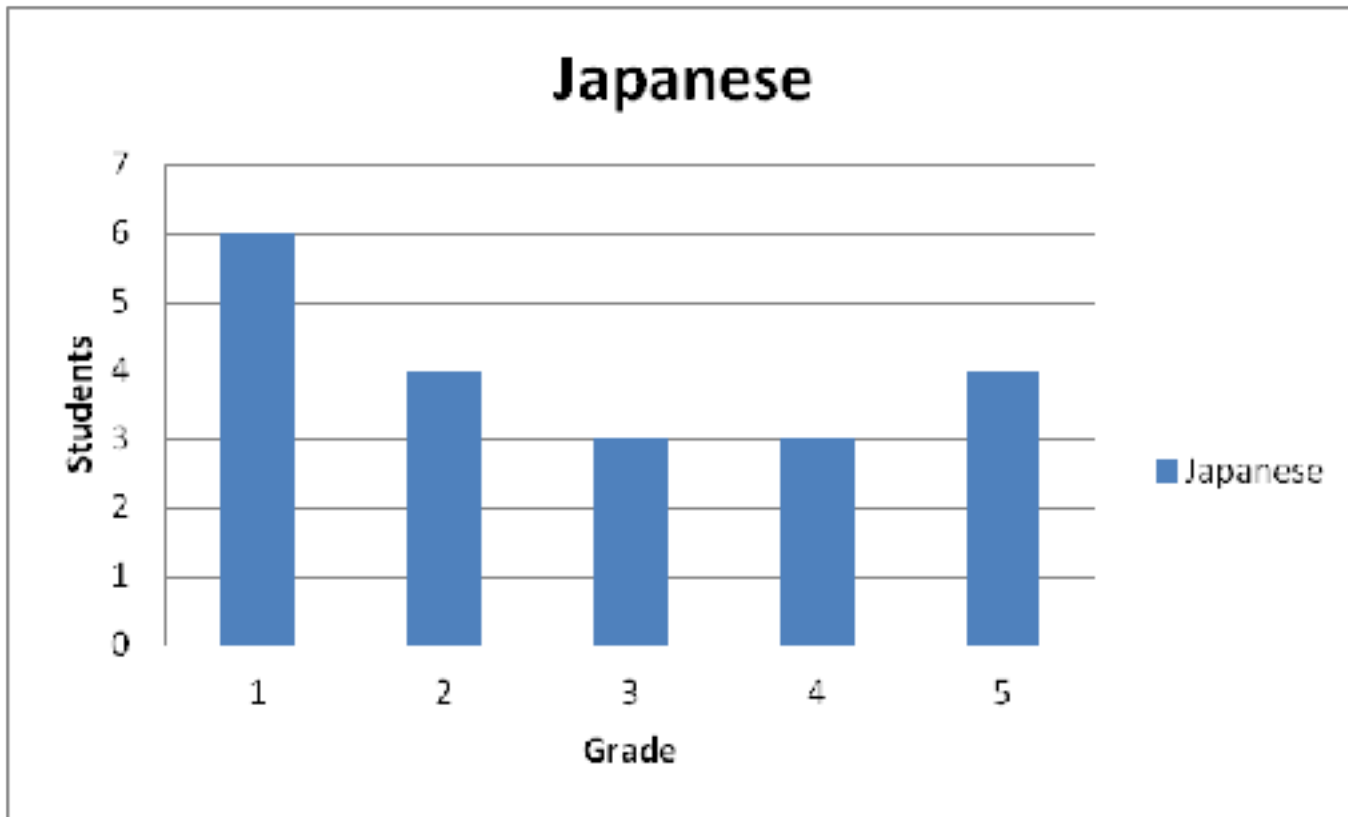
- Bar chart



8. Enter chart title and label for category and value

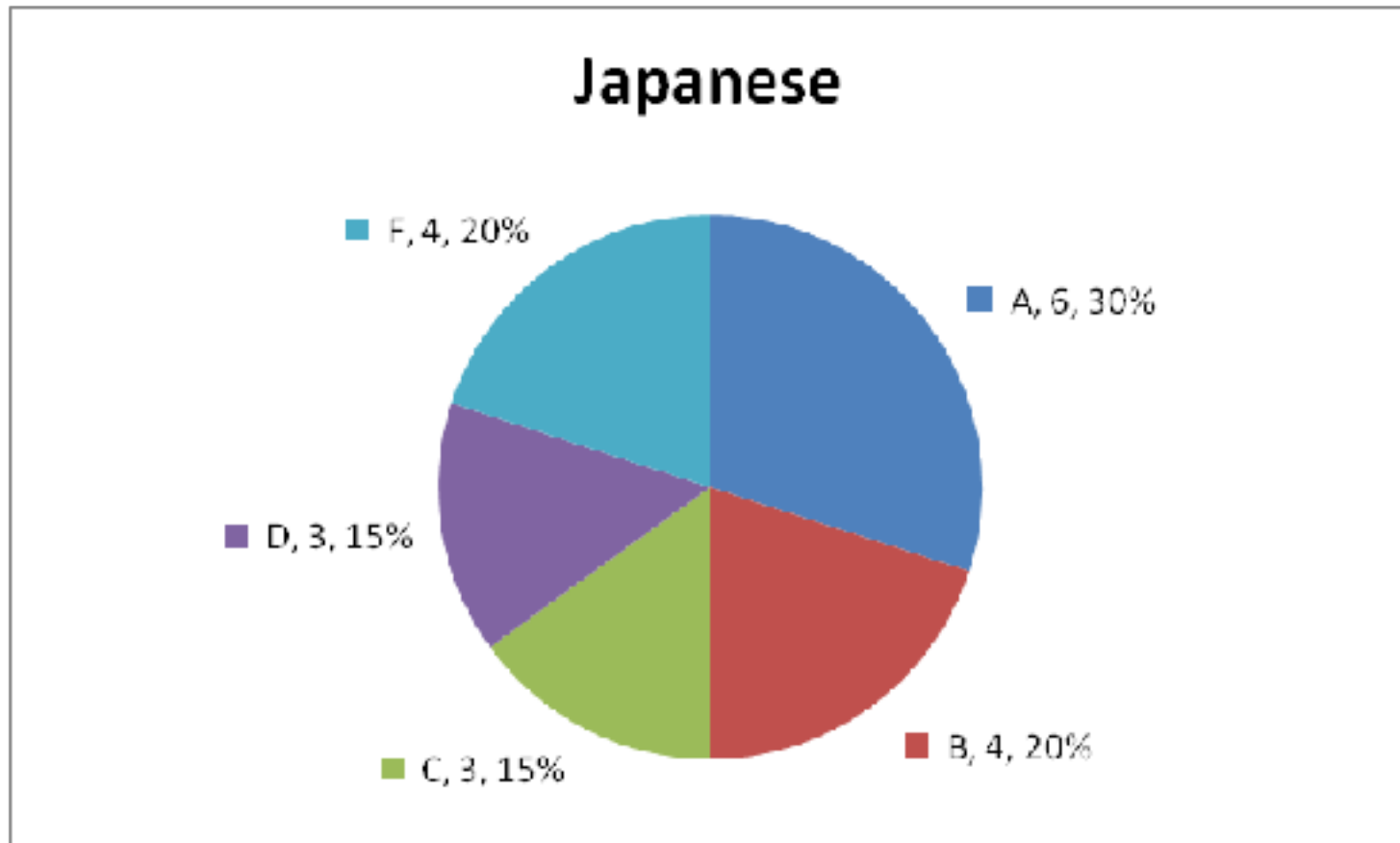
Many different charts

- Bar Chart



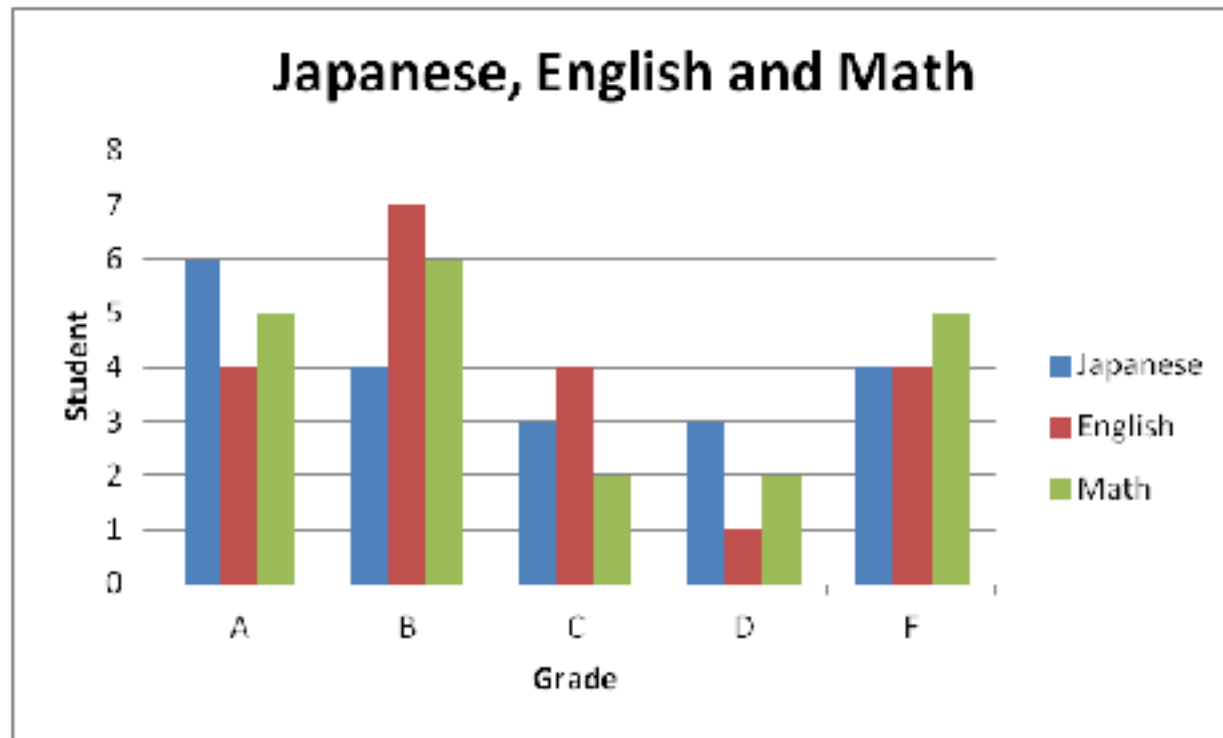
Other Charts

- Pie chart



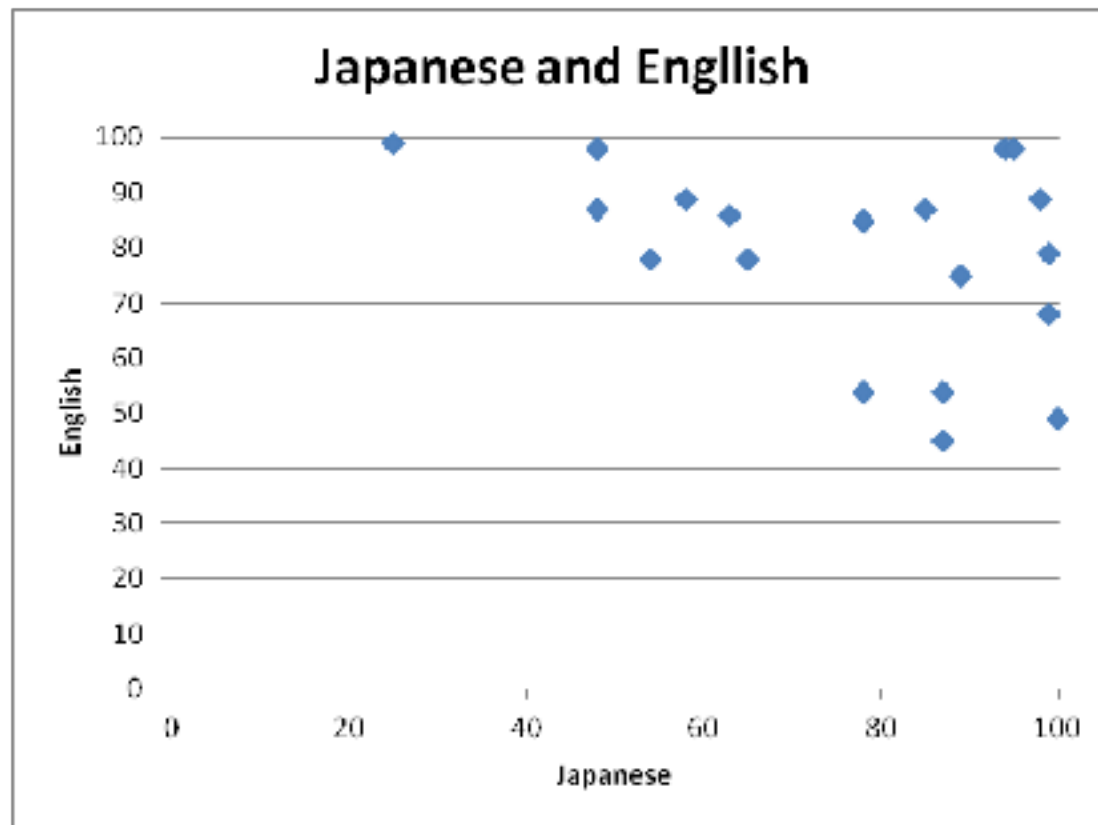
Even more

- Column chart



How many drawings can we have?

- XY (scatter)



情報基礎B (Computer Literacy)

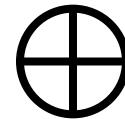
Lecture 6: computer language

Matias Korman

Tohoku University Graduate School of Information Sciences
System Information Sciences
Design and Analysis of Information Systems

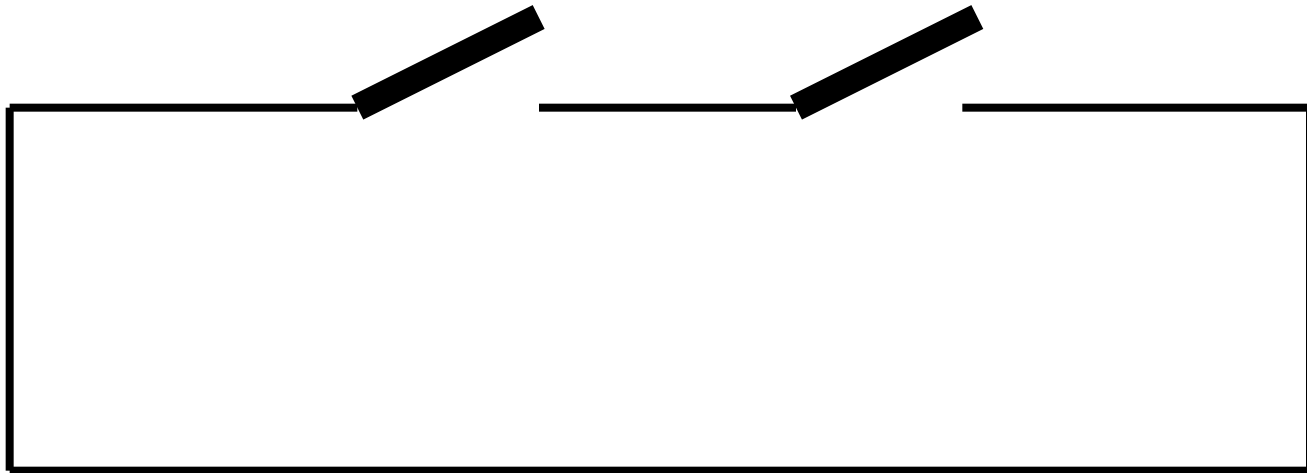
Computation

- The world of $\{0, 1\}$
 - Boolean algebra
 - Logical operation
- AND operation
 - $0 \times 0 = 0 \times 1 = 1 \times 0 = 0, 1 \times 1 = 1$
 - Series circuit
- OR operation
 - $0 + 0 = 0, 0 + 1 = 1 + 0 = 1, 1 + 1 = 1$
 - Parallel circuit
- NOT operation
 - $\text{NOT}(0) = 1, \text{NOT}(1) = 0$



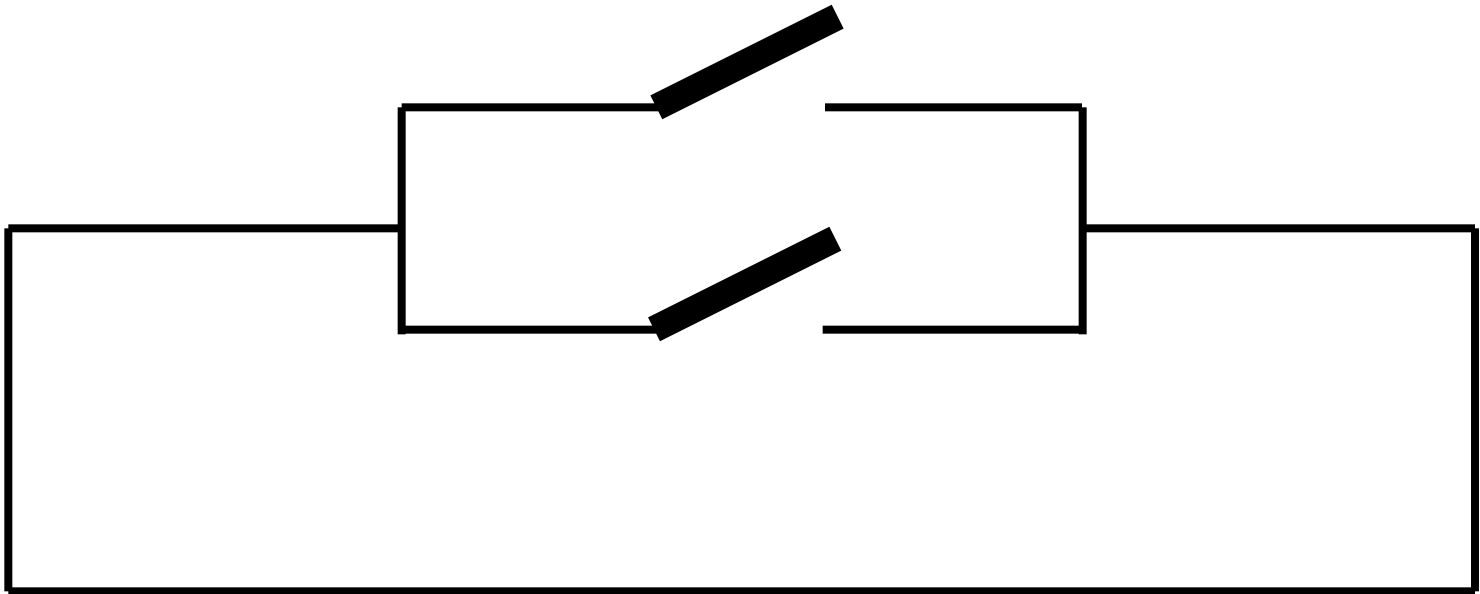
Computation

- AND operation
 - $0 \times 0 = 0 \times 1 = 1 \times 0 = 0, 1 \times 1 = 1$
 - Series circuit
 - Switching : relay, transistor, diode



Computation

- OR operation
 - $0 + 0 = 0$, $0 + 1 = 1 + 0 = 1$, $1 + 1 = 1$
 - Parallel circuit



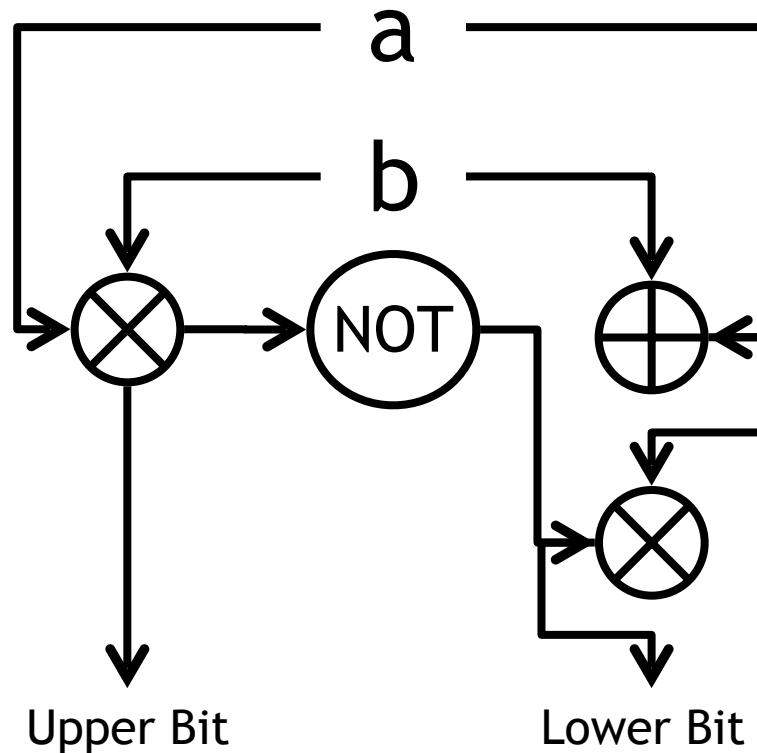
Computation

- NOT operation
 - $\text{NOT}(0) = 1$, $\text{NOT}(1) = 0$



Adding two bits

- Circuit to calculate $a + b$ ($a, b = 0$ or 1)



$$0 + 0 = 00$$

$$1 + 0 = 01$$

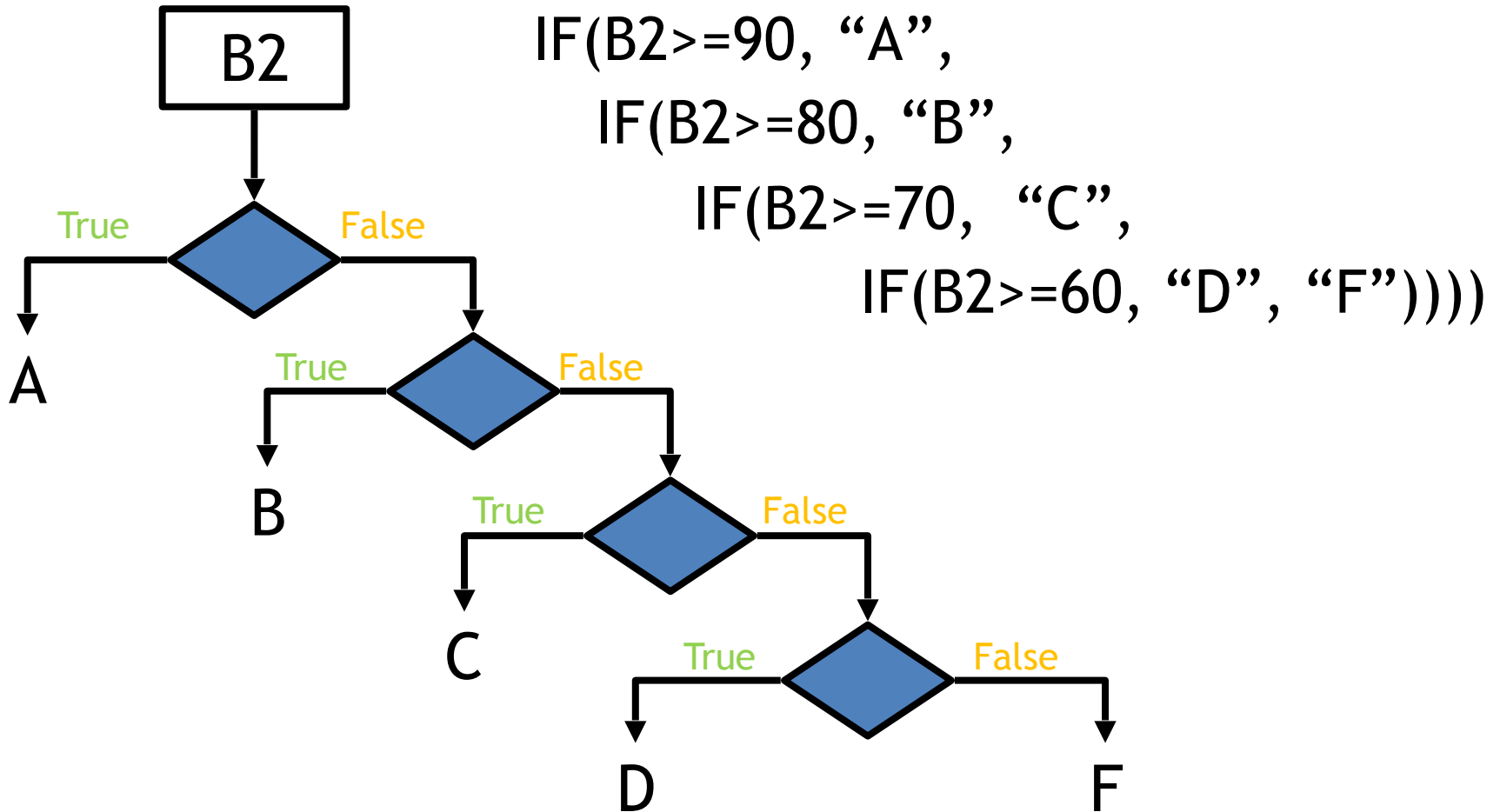
$$0 + 1 = 01$$

$$1 + 1 = 10$$

Computation Model

- AND, OR, NOT
 - Logic operations
 - Addition, subtraction, ...
 - IF (via comparison)
- Programming model (decision tree model)
 - Basic math operations
 - Load data from storage
 - Move data to storage
 - IF operations to branch

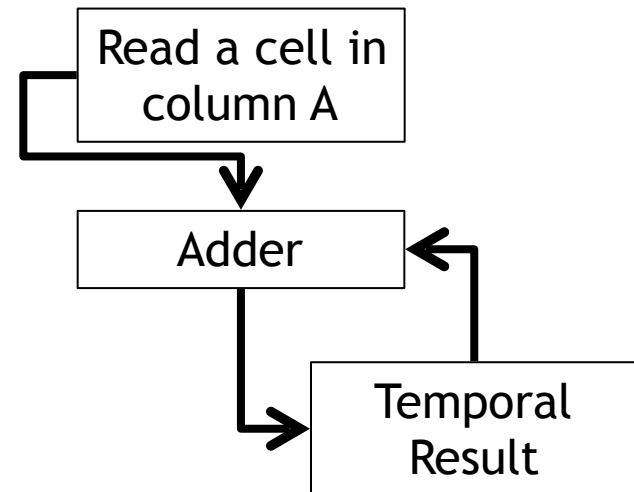
Program and Decision Tree



Computation proceeds through the tree

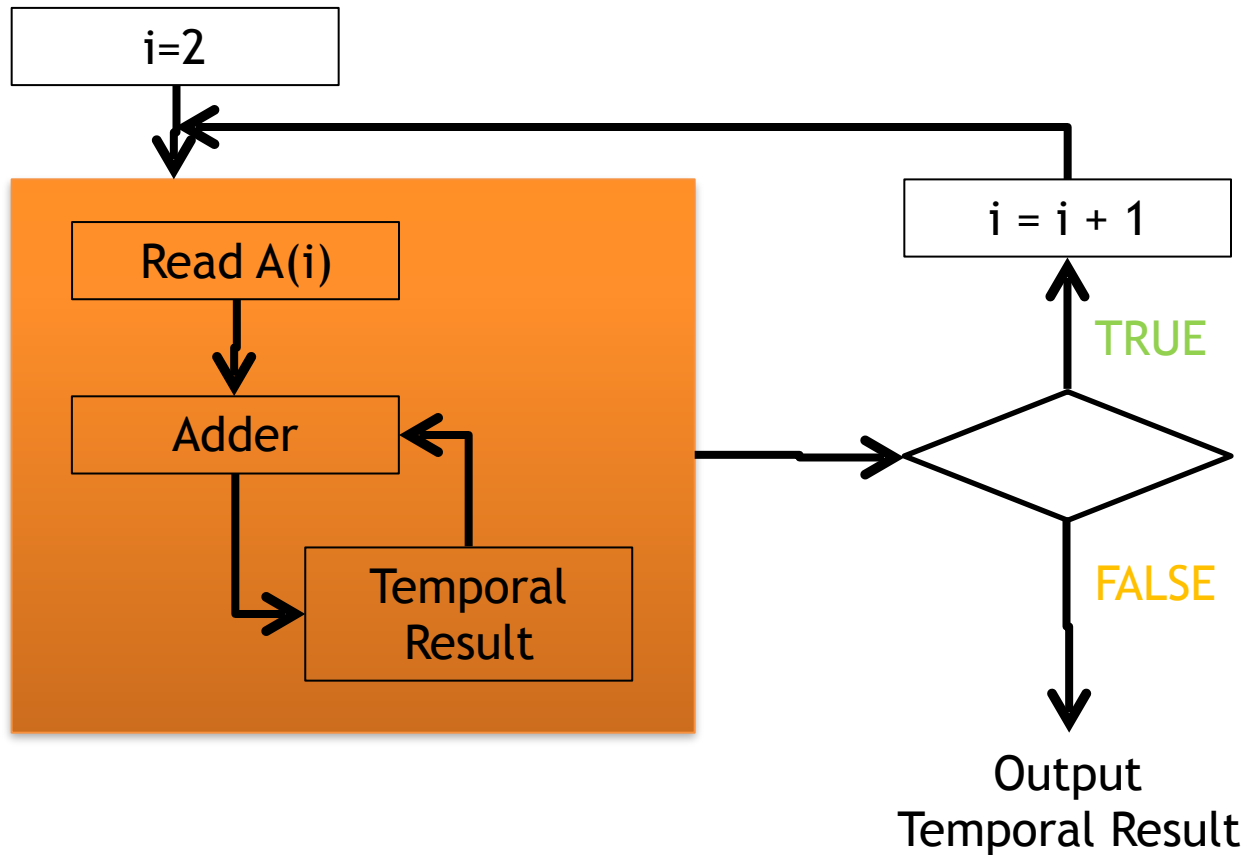
New tool: loop

- SUM(A2:A100)
 - Decision tree becomes deep
 - Loop structure
 - Loop management
 - When to stop?
 - In Excel...
 - By setting range A2:A100



Loop

- Example: $SUM(A2:A100)$



Programing

1. Think about an algorithm
 - e.g. Grading procedure
2. Show a flowchart with decision tree and loop structure
3. Write the algorithm in programming language
 - Simple easy vocabulary
 - No ambiguities

Programing

- Make operations for computers
 - Algorithm design
 - Logical thinking needed
 - Coding
 - Transform idea into “computer language”
 - Learning a programming language
 - Read programs
 - Change/make your own
 - Debug (error correction) is crucial

Programing hints

- Learning programming language
 - Practice makes perfect
- Follow examples
 - This is a pen > This is a dog
 - Follow good examples
- Learn from the mistakes
 - This is a apple ??
 - I is a man ??
 - 99.9999% of computer errors are our fault
- System guides in the type of error
 - Syntax error (i.e., #VALUE)
- Guess what is wrong/mistakes
 - Logical thinking helps