Review

ICS3U0 – Unit 1

1. Handy info to memorize

Input

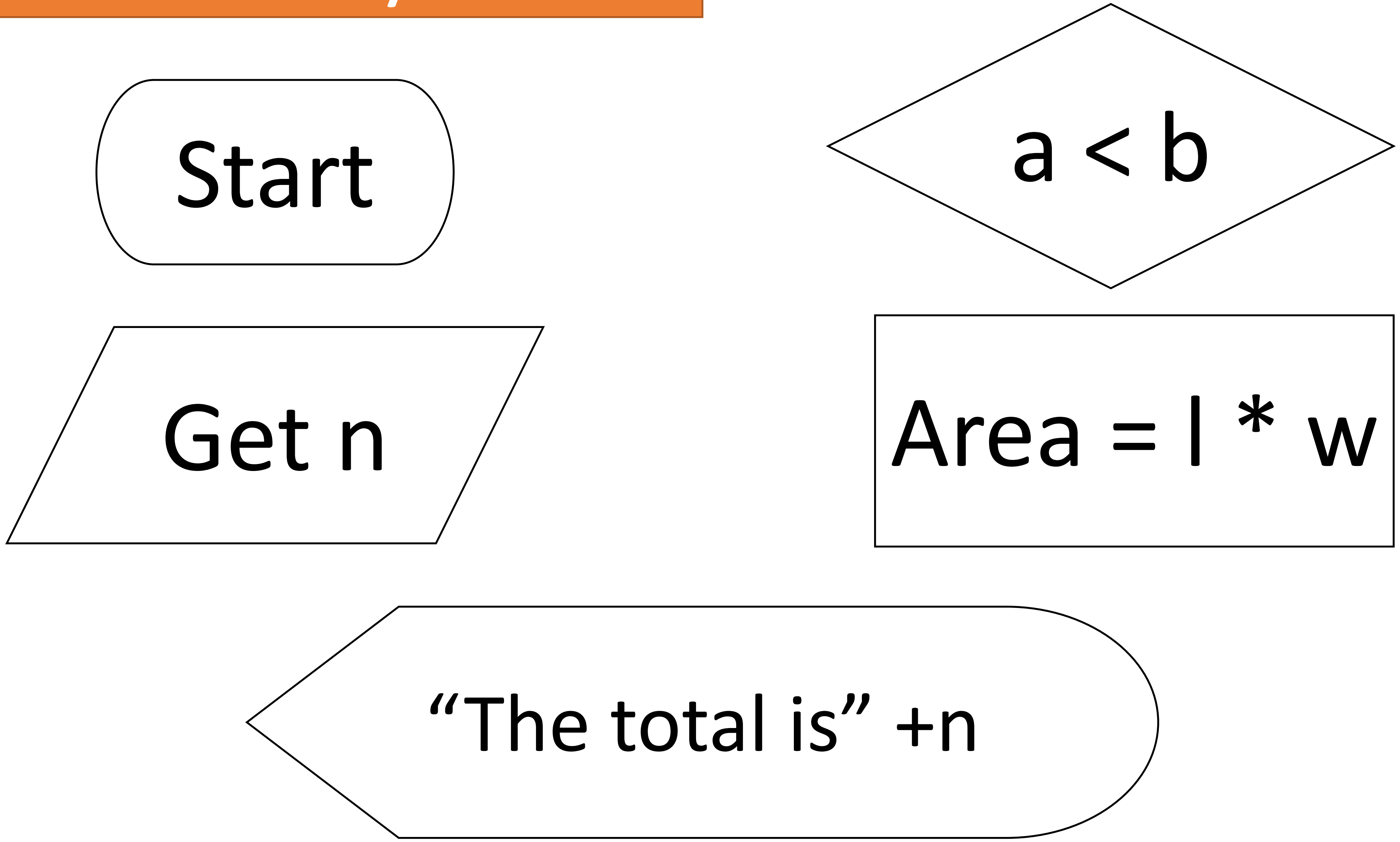
```
int n = IO.inputInt("Number? ");
```

```
String name = IO.inputString(“Name? ");
```

Output

```
System.out.println("The total is "+n);
```

Flowchart Symbols



Mod (%, Remainder) Facts

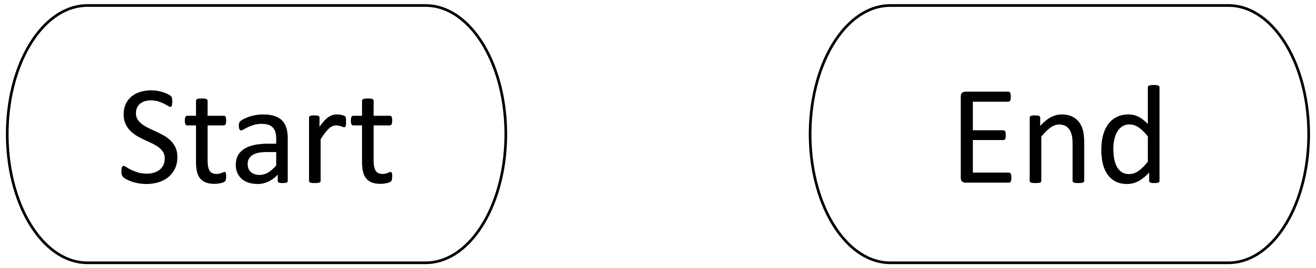
1. x % 0 = Error
2. x % 1 = 0
3. x % 2 = 1 (x is odd)
4. x % 2 = 0 (x is even)
5. x % y = x (if y>x)
6. x % x = 0

1. 8 % 0 = Error
2. 5 % 1 = 0
3. 10 % 2 = 1 (x is odd)
4. 9 % 2 = 0 (x is even)
5. 10 % 24 = 10 (24 bigger)
6. 3 % 3 = 0

Boolean Expression Errors

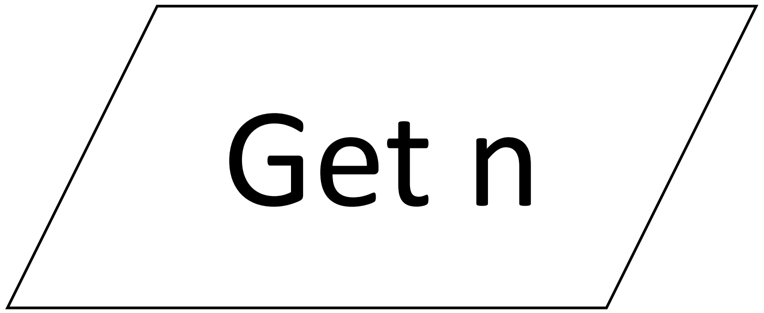
`y=< 4`	`y<= 4`
`1<x<5`	`x>1 && x<5`
`x>1 && <5`	`x>1 && x<5`
`x=y`	`x==y`

2. Flowchart Rules



The ONLY thing it can hold.

One of each per flow chart.



```java
int n = IO.inputInt("Number? ");
```

From IO,input lines

ONLY write “Get” + variable name

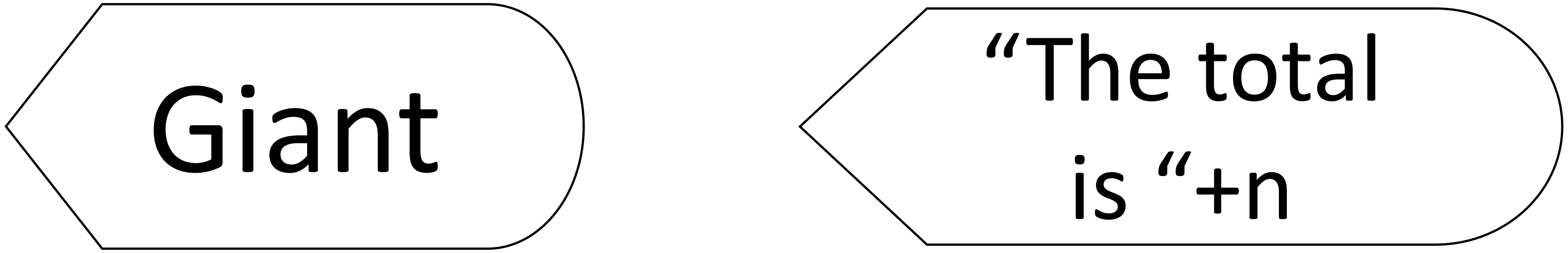
SA = 6 * a * a

Calculate surface area

```
int SA = 6 * a * a;
```

Math calculations

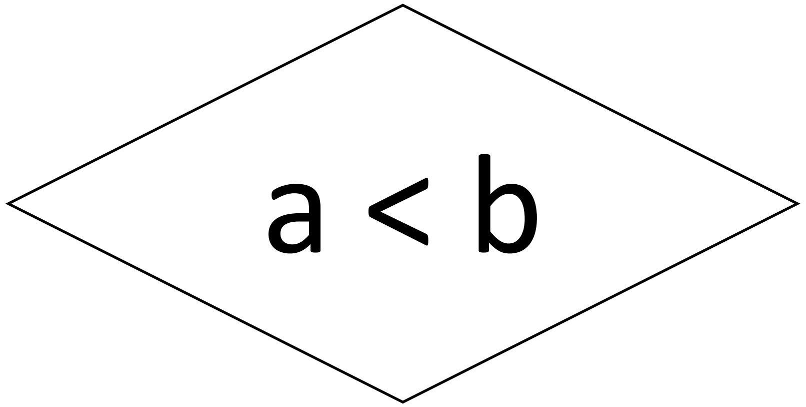
Leave the variable type off, put in the calculation

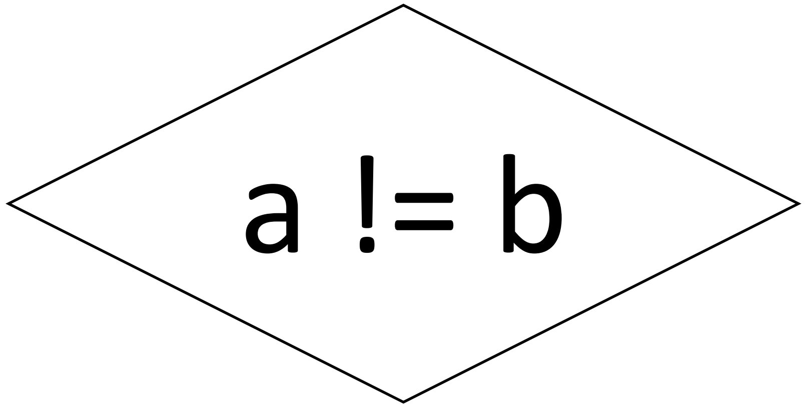


```java
System.out.println("Giant");
System.out.println("The total is "+n);
```

System.out.println statements

Just include what is in the brackets.





```
if (a < b)
else if (a != b)
```

Boolean Expressions

Just include what is in the brackets.

3. Why is X useful?

Binary

- The number system used by computers.
- Everything on a computer is translated to binary, is stored in binary, and is transmitted over the internet in binary.
- It is used instead of regular Base-10 numbers because it is easy to store on hardware. Numbers can be reduced to a series of on/off commands.

Boolean Algebra

- Mathematical expressions that results in either true or false.
- They can be joined to make more complex Boolean expression using Boolean Operators like AND, OR, NOT.
- They are used to control if statements and loops. This means that Boolean Algebra is one of the cornerstones of computer science.

ASCII

- Stands for American Standard Code for Information Interchange (you don’t have to know that)
- It is used to translate letters to a number. The number is then translated to binary.
- It is how simple text is stored in binary.
- It only works for English characters. It is too short to do other languages.
- With ASCII, you can’t discard the leading zeros: 0000001 can not be written as 1.

If Statements

- Decision statements
- Useful to do something like determine a winner: if one score is larger than the other, the first player wins.
- Controlled by Boolean Expressions.
- The first Boolean Expression that is true is the clause that runs.
- Once one clause is run, it skips to the end of the if. The others aren’t even tested.

If Statements Syntax

- First clause must be an if
- The last clause can be an else. This means it is a the default. It does not need a Boolean expression.
- The middle clauses are else ifs
- There is no semi-colon after the Boolean expression
- When the clause has more than one line of code inside it, then it needs to have { } around it.

Flowcharts

- Diagrams that trace how a computer FLOWS through a piece of code.
- Used to plan out a program before you code.
- Also used to understand a program because it represents it visually.

Hexadecimal

- Base 16 numbers.
- They use the columns based on 16: 4096, 256, 16, 1
- They use the digits 0-9 and also A=10, B=11, C=12, D=13, E=14, F=15
- Computers do NOT use them. Their only purpose is as a short form of binary code for HUMANS.
- Because $2^4=16^1$, four binary digits can be condensed into one hexadecimal digit.

Input

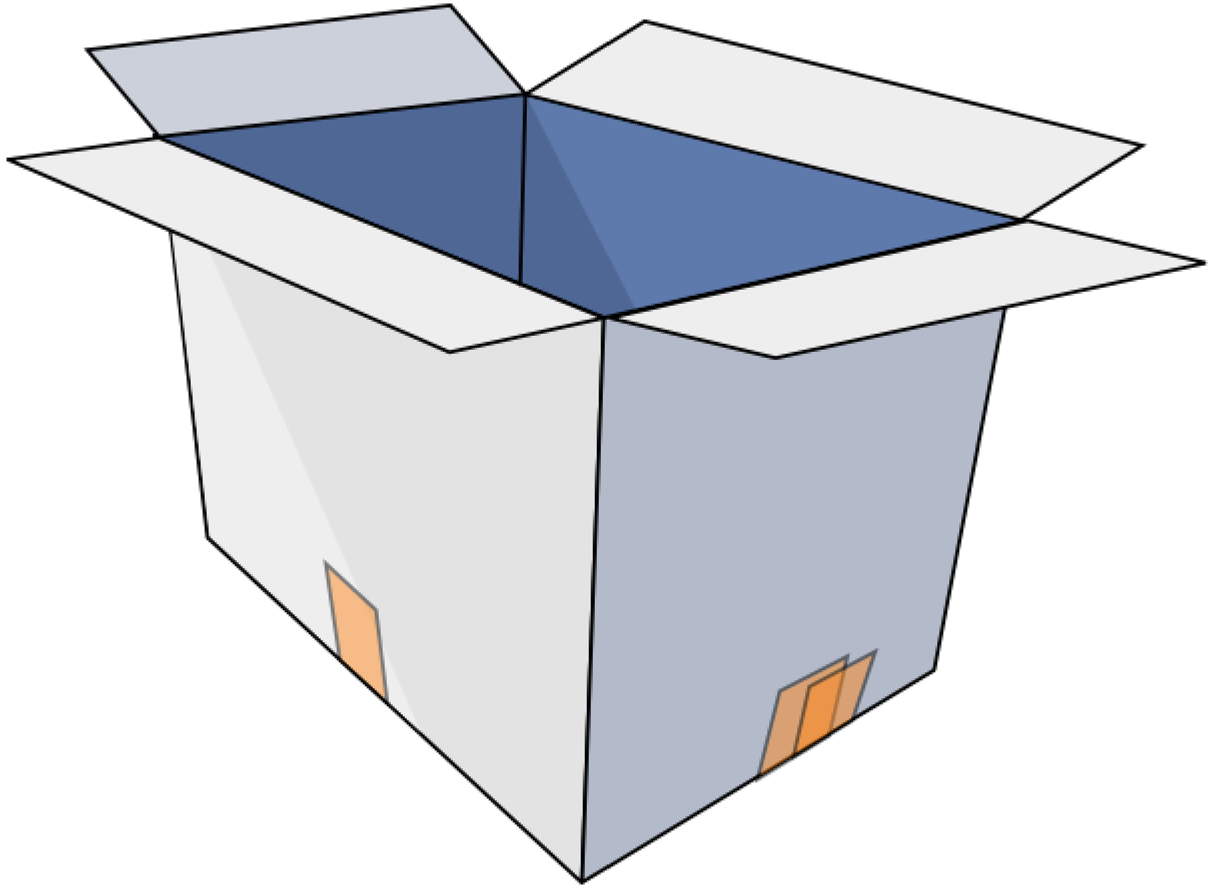
- Input in Command Line Interfaces (CLIs) comes from the keyboard.
- We write a prompt, or question, and the computer waits for an answer.
- Input is stored in variables so that we can use it again.

- Input is useful because we can customize our output. With input, our program can calculate a specific answer OR it can have customized responses for the user.
- Input makes programs responsive.

4. Variables and Special Output

Back slash

```
\n   new line
\t   tab
\”   make a quote
        needed because quotes start
        and end the text in the println
\\   make a back slash
        needed in case we don’t want to
         pause the output, and we actually
         want a back slash.
```



A variable is used in programming to store a value until we need it again.

It is sort of like a box.

Types

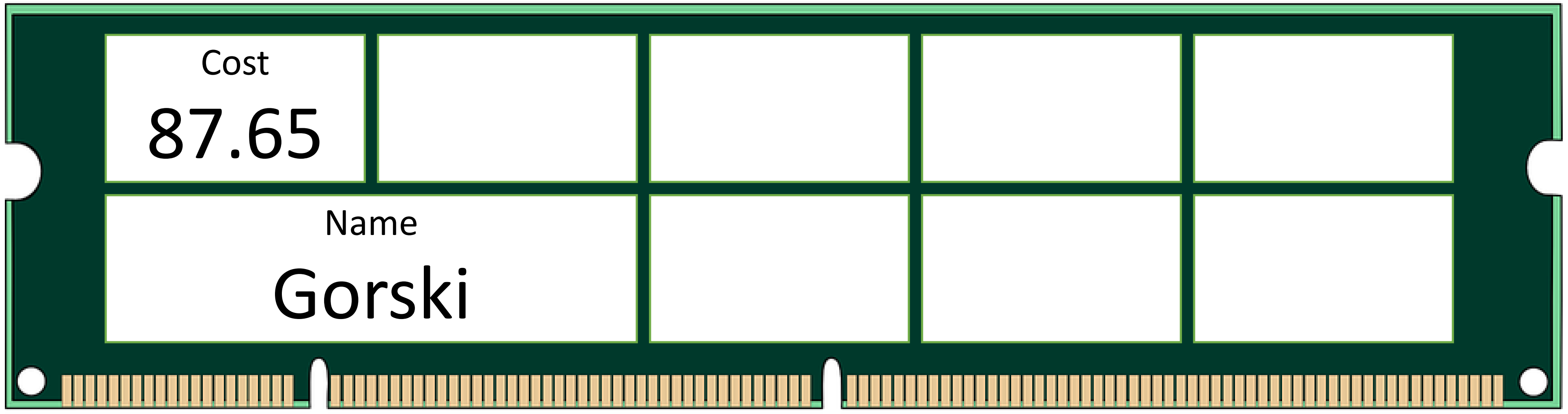
Java variables have types:

- A type is a kind of data – for example, words are text.
- A type is also an amount of RAM allocated to a variable.
- A type is also the kinds of operations that can be done with a variable.

The Types

- in order of size. (Smallest to largest)

```
boolean - true or false
char - single characters, for example: 'a'
int  - integers, for example: -2, 0, 67
double - decimals, for example: 2.0, 3.123
String - text, for example: "Hello"
```



```
double cost = 87.65;

String name = “Gorski”;
```

A variable is a space in RAM.
It has a name, a value, and a type.

Its value can change whenever you want. Its name and type cannot.

Declaring

Declaring is creating a new variable

```
int num = 4;
String name = “Gorski”;
double amt = 5.6778;
char ans = ‘y’;
double total = amt * 3;

type name = starting value;
```

Variable Naming Rules:

Because variables are used in the code, they have be named carefully so the computer can understand.

1. No spaces. Camel case or underscores are fine.
2. Meaningful is easier in the long run.
3. Cannot start with a number. Otherwise, numbers are fine.
4. Cannot contain odd characters.
5. Cannot contain reserved words. (eg. public or main). These already have a purpose in java.