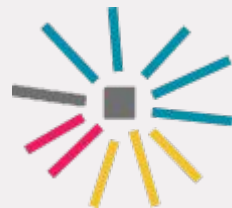


WIP

OPERATORS

PALIMPSEST



OPERATORS

What do these operators do or allow?

- Arithmetic
- Logical
- Relational
- Equality
- Conditional
- Conditional statements

OPERATORS



ARITHMETIC OPERATORS

do math

LOGICAL OPERATORS

evaluate
true/false



EQUALITY OPERATORS

test
equivalence

RELATIONAL OPERATORS

compare
value



ARITHMETIC OPERATORS



multiplication



addition



division



subtraction

**IN CODE
AMBIGUITY
IS BAD**

**SO THERE
ARE RULES
AND HELPERS**

PRECEDENCE

ORDER OF OPERATIONS



multiplication



addition



division



subtraction

Which takes precedence?

ORDER OF OPERATIONS



multiplication



addition



division



subtraction



ORDER OF OPERATIONS

$$x = 7 + 3 * 6$$

Which takes precedence?

ORDER OF OPERATIONS

$$X = \overset{7 + 3}{10} * 6$$

Addition?

ORDER OF OPERATIONS

$$x = 7 + 3 * 6$$

Multiplication?

ORDER OF OPERATIONS

$$x = 10 * 6$$

$$x = 7 + 18$$

What is the value of x?

PRECEDENCE

()

ORDER OF OPERATIONS

$$x = 7 + (3 * 6)$$

Which takes precedence?

ORDER OF OPERATIONS

$$a + b + c / 3$$

Where should the parentheses go?

ORDER OF OPERATIONS

$$a + b + c / 3$$

$$a + b + (c / 3)$$

Where should the parentheses go?

**PARENTHESES
ARE YOUR
FRIENDS**

**YOUR VERY
VERY GOOD
FRIENDS**

ORDER OF OPERATIONS

$$y = m * x + b;$$

Where should the parentheses go?

ORDER OF OPERATIONS

$$y = m * x + b;$$

$$y = (m * x) + b;$$

Where should the parentheses go?

IF IN DOUBT
WORK **INSIDE**
OUT AND **LEFT**
TO RIGHT

**GROUPING
PAIRS BY
PRECEDENCE**



EXERCISE

Place
parentheses

ORDER OF OPERATIONS

$$y = a * b * b + c * b - d;$$

Where should the parentheses go?

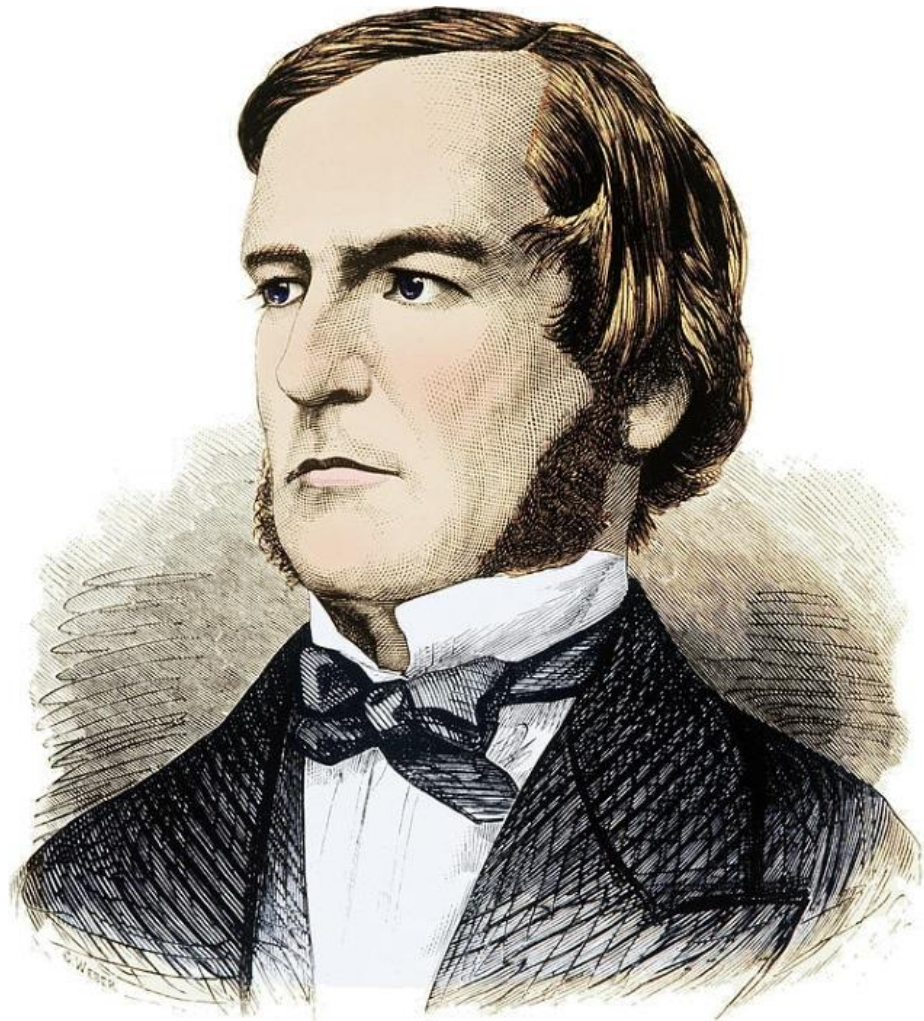
ORDER OF OPERATIONS

$$y = a * b * b + c * b - d;$$

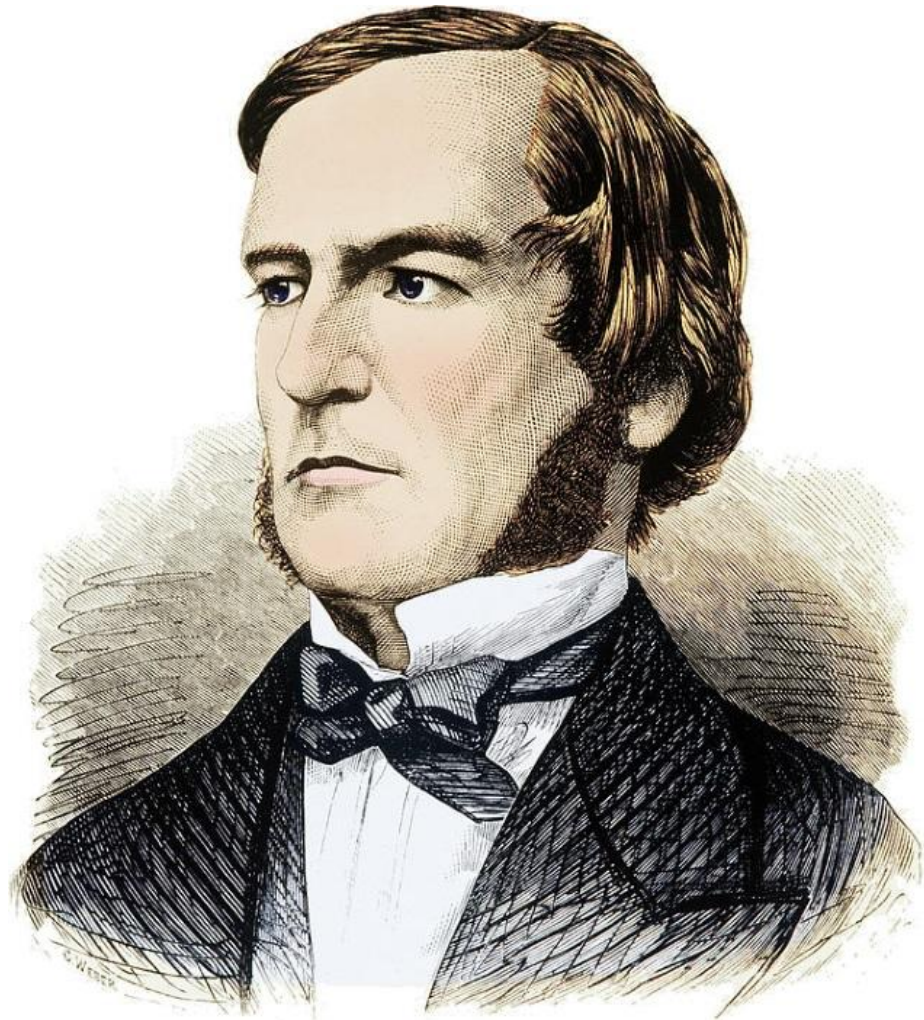
$$y = (((a * b) * b) + (c * b)) - d;$$

Where should the parentheses go?

BOOLEAN EXPRESSIONS

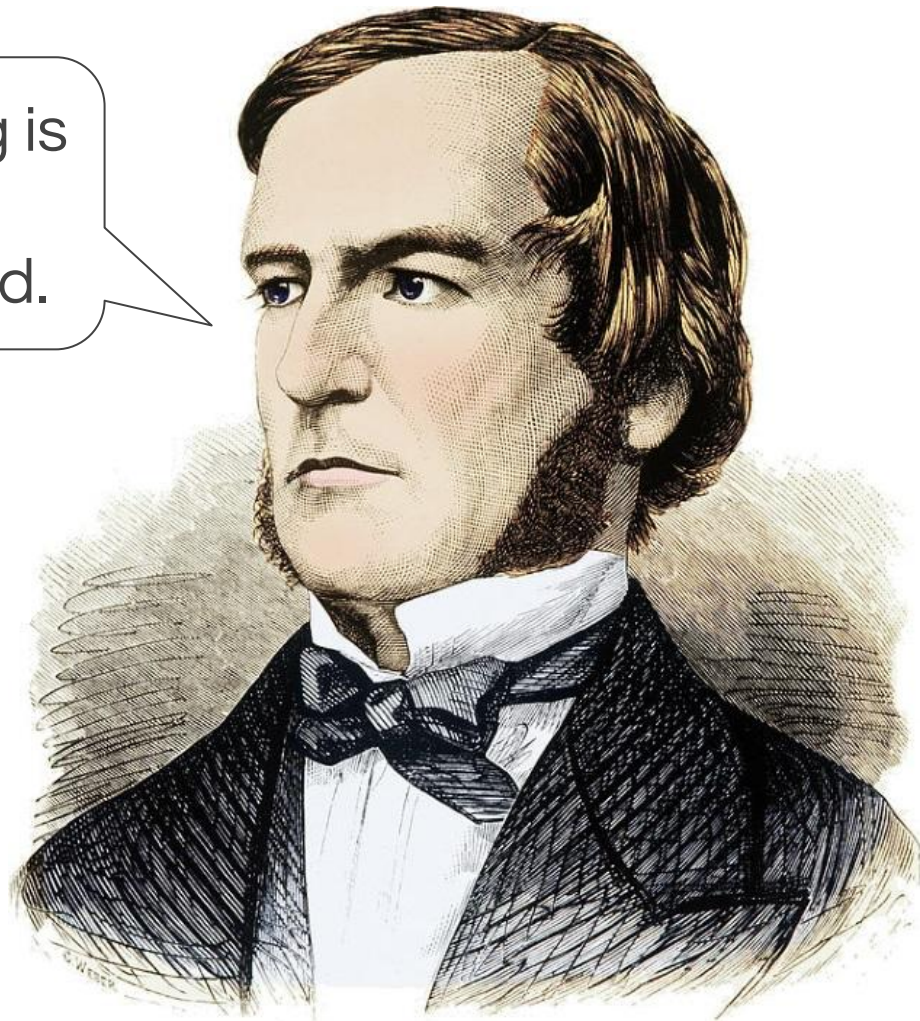


GEORGE BOOLEAN EXPRESSIONS



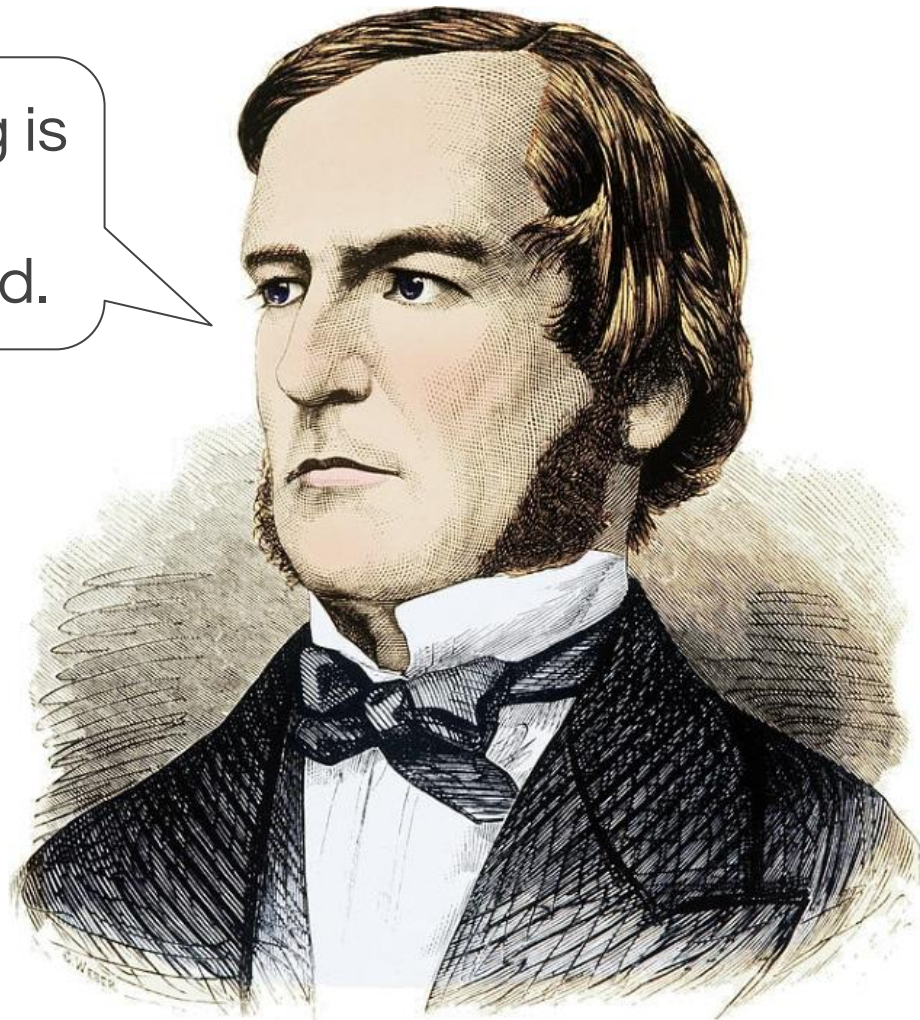
Programming is
positively
afternoonified.

BOOLEAN EXPRESSIONS



Programming is
positively
afternoonified.

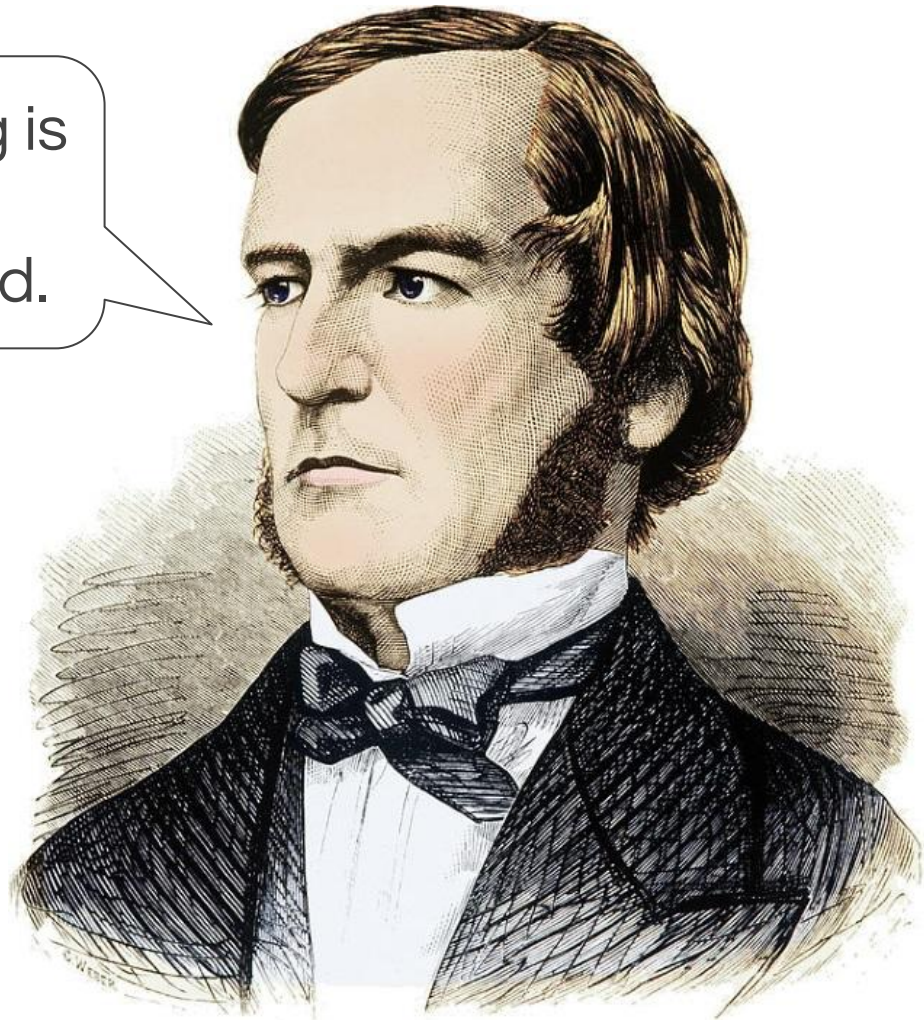
BOOLEAN EXPRESSIONS



Programming is
positively
afternoonified.

BOOLEAN EXPRESSIONS

True or False



RELATIONAL OPERATORS

> greater than

< less than

>= greater than
or equal to

<= less than or
equal to

== equal to

!= not equal to

```
01     public bool boolVar;
02     boolVar = false;
03
04     Debug.Log ("boolVar: " + boolVar);
05
06     public int a = 10;
07     public int b = 10;
08
09     boolVar = ( a == b );
10     Debug.Log ("boolVar: " + boolVar);
11     Debug.Log (a == b);
12     Debug.Log (a != b);
13     Debug.Log (a < b);
14     Debug.Log (a <= b);
15     Debug.Log (a > b);
16     Debug.Log (a >= b);
```

```
01     public bool boolVar;
02     boolVar = false;
03
04     Debug.Log ("boolVar: " + boolVar);      boolVar: false
05
06     public int a = 10;
07     public int b = 10;
08
09     boolVar = ( a == b );
10     Debug.Log ("boolVar: " + boolVar);      boolVar: true
11     Debug.Log (a == b);                     true
12     Debug.Log (a != b);                     false
13     Debug.Log (a < b);                       false
14     Debug.Log (a <= b);                      true
15     Debug.Log (a > b);                       false
16     Debug.Log (a >= b);                      true
```

EQUALITY OPERATORS

= is not the same as **==**

EQUALITY OPERATORS

= is not the same as **==**



assignment
operator



equality
operator

EQUALITY OPERATORS

= is not the same as **==**



assigns



evaluates

EQUALITY OPERATORS

= is not the same as **==**



“gets”

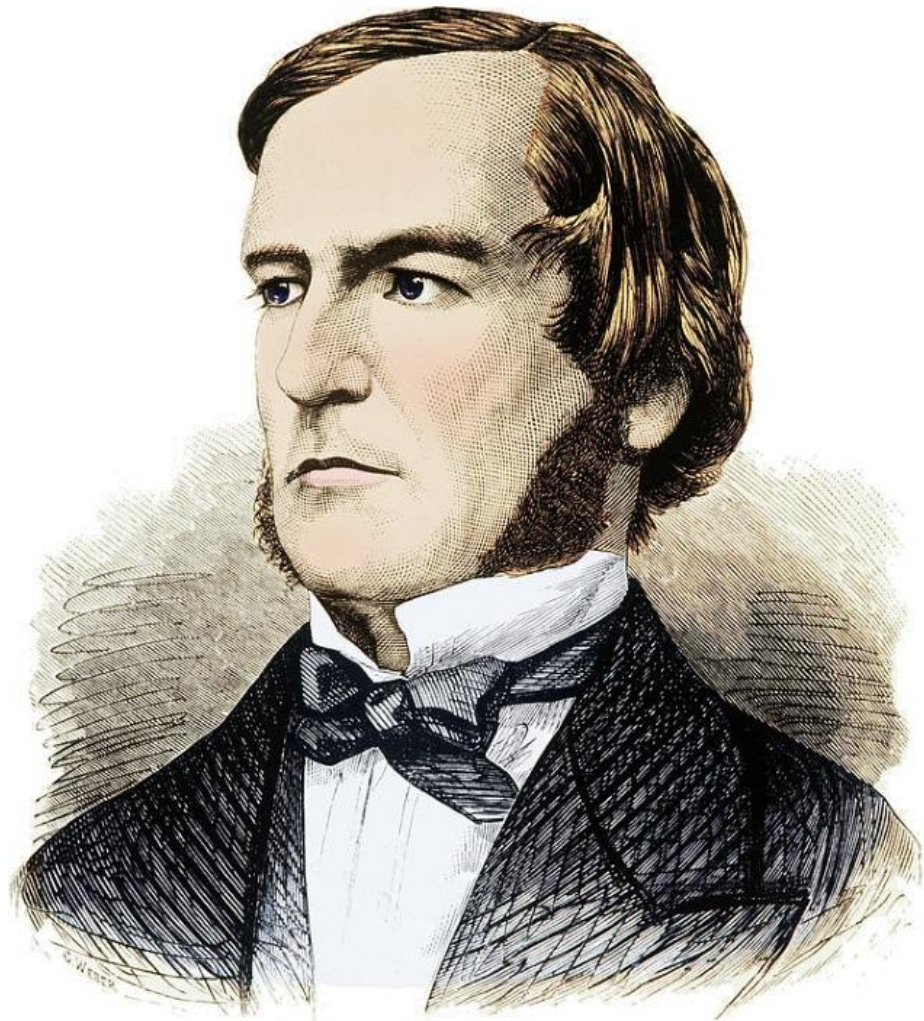
“is assigned the value”



“is **e**xactly **e**qual to”

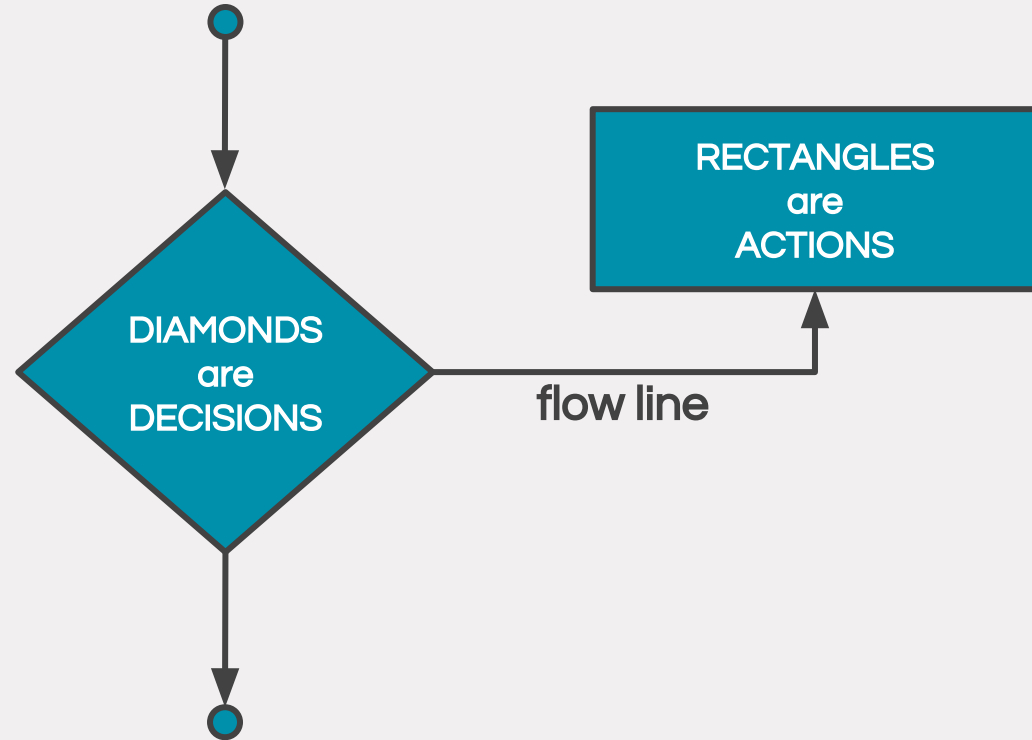
BOOLEAN EXPRESSIONS

What can we do
with these?



CONDITIONAL STATEMENTS

FLOW CHART BASICS



if BASICS

Pseudocode:

if some Boolean expression is true
do this

Example:

```
if ( x == y )  
{  
    Debug.Log("x equals y!");  
}
```

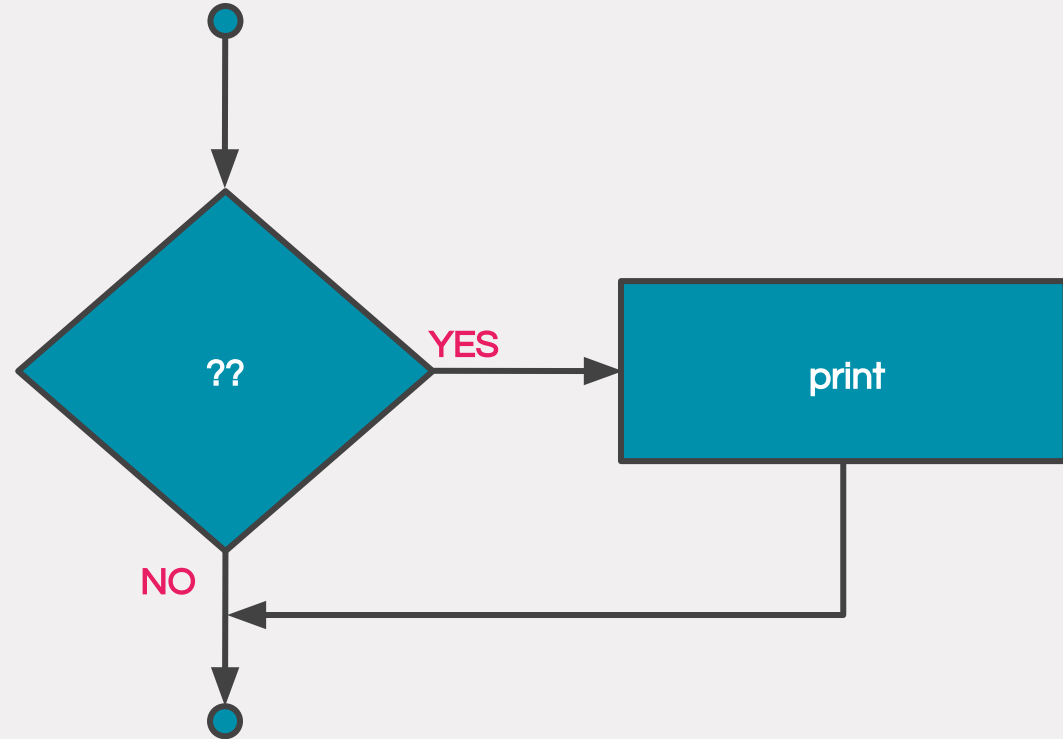
if MORE

```
if ( temperature >= 85 )  
{  
    Debug.Log("it's hot out!");  
}
```

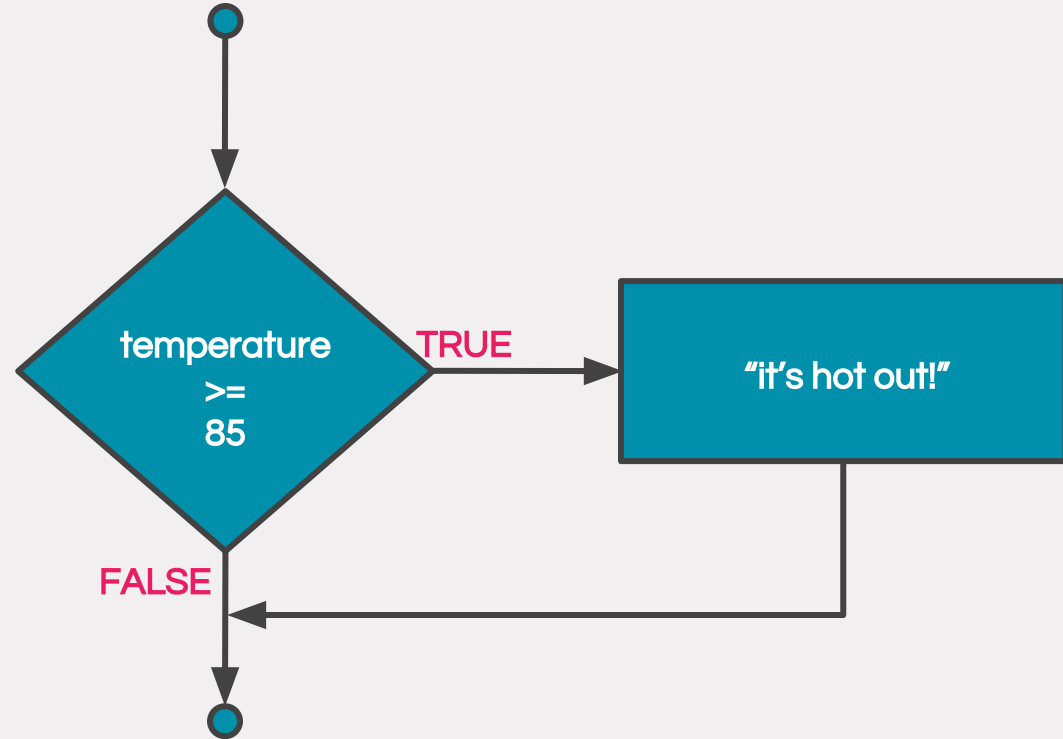
```
if ( temperature <= 65 )  
{  
    Debug.Log("wear a coat!");  
}
```

— — —

if FLOW CHART



if FLOW CHART



if else BASICS

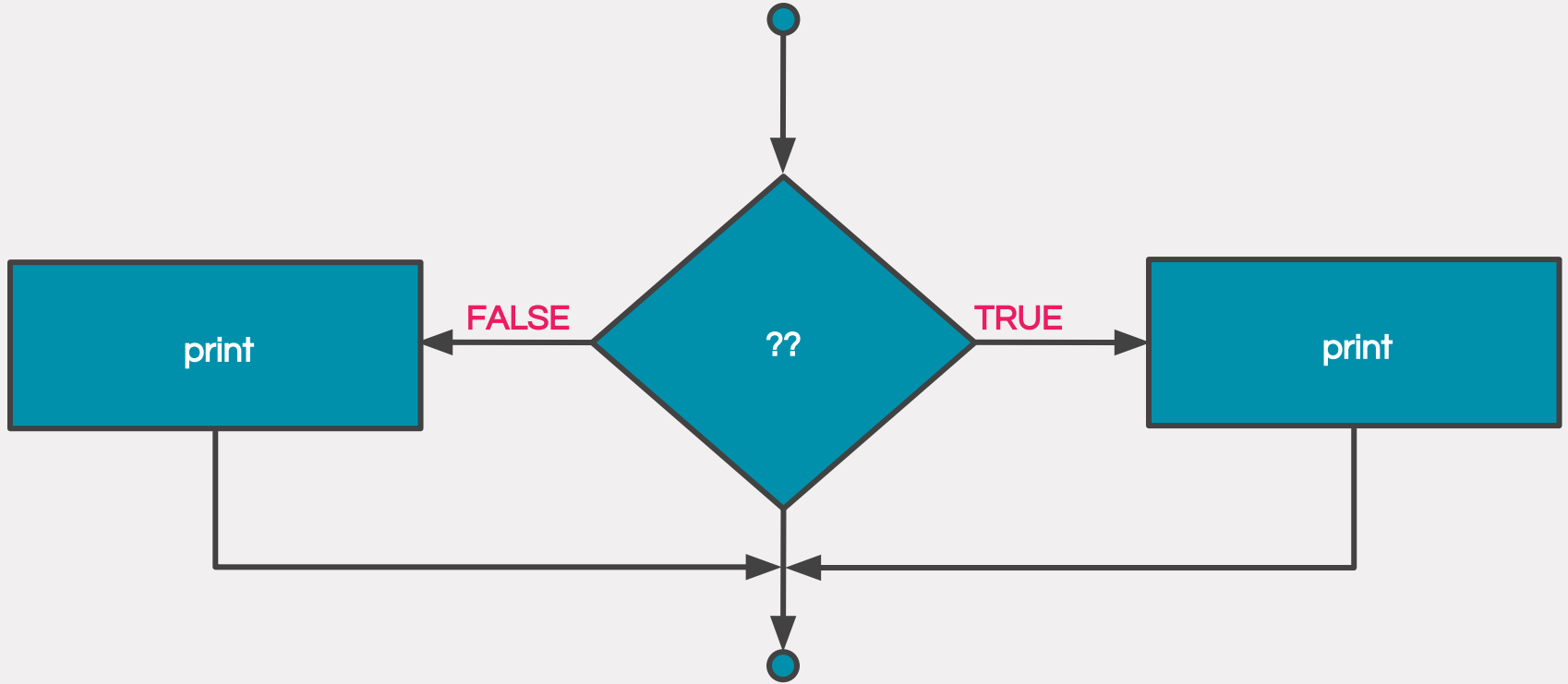
Pseudocode:

if some Boolean expression is true
do this OTHERWISE do this

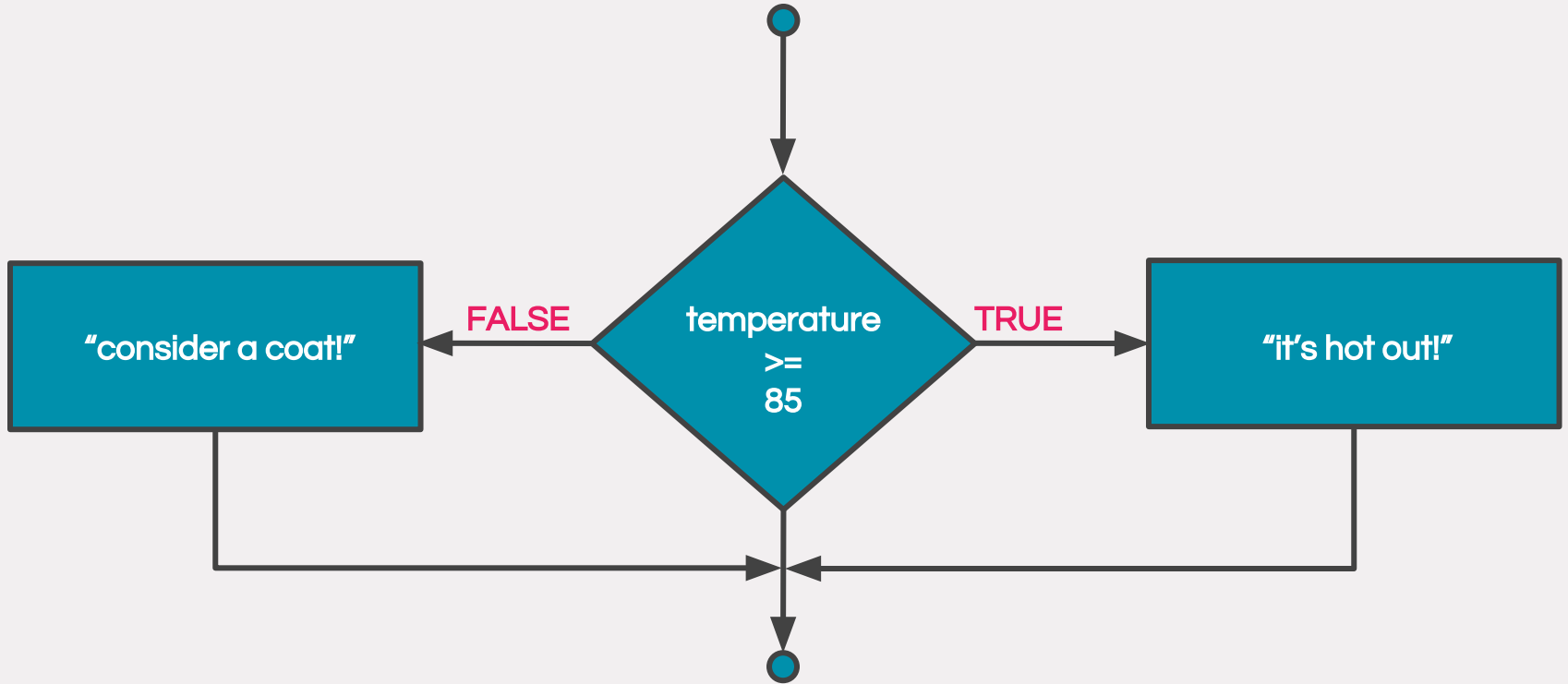
Example:

```
if ( temperature >= 85 )  
{  
    Debug.Log("x equals y!");  
}  
else  
{  
    Debug.Log("consider a coat!");  
}
```

if/else FLOW CHART



if/else FLOW CHART



LOGICAL OPERATORS

&&



AND

||



OR

&& BASICS

Pseudocode:

if this is true AND that is true, do this

Example:

```
if ( temp >= 85 && temp <= 65 )  
{  
    Debug.Log("beautiful day!");  
}
```

|| BASICS

Pseudocode:

if this is true OR this is true, do this

Example:

```
if ( temp >= 85 || precip == false
)
{
    Debug.Log("beautiful day!");
}
```

BONUS OPERATORS

+= condenses a statement

X += 2 is the same as **X = X + 2**

Shortcut Operator

BONUS OPERATORS

$x *= 5$ is equivalent to $x = x * 5$

$x -= 5$ is equivalent to $x = x - 5$

$x /= 5$ is equivalent to $x = x / 5$

BONUS OPERATORS

Post
X++ adds 1 to the value of x

Pre
++X adds 1 to the value of x but AFTER

Increment Operator

BONUS OPERATORS

Post
X++ adds 1 to the value of x

Pre
++X adds 1 to the value of x but AFTER

Sneak Preview of LOOPS - stay tuned!

OPERATORS REVIEW

Can you help me make that call?

- Definition
- Arithmetic
- Logical
- Relational
- Equality
- EXERCISE
- Conditional
- Conditional statements

NEXT UP

LAB

