Working with Numbers
Integers vs Reals

Data Types:

An integer is a positive or negative whole number (or zero). In Turing, an integer value can be any whole number between -2,147,483,647 to 2,147,483,647.

A real number includes all integers, but also allows for numbers with a fractional or decimal component. For example, 3.141592654 is the mathematical constant π.

Converting: Integer to Real

Since the real numbers include the integers, this conversion is automatic.

```
var intNum : int
var realNum : real

intNum = 4

realNum = 4          % saves 4.0
realNum = intNum     % saves 4.0
realNum = intNum/3   % saves 1.33333...
```

Converting: Real to Integer

Most real numbers cannot be stored as an integer. Trying to do this without understanding the limitations can result in some unusual behaviour.

```
var intNum : int
var realNum : real

realNum := 4/3    % stores 1.33333
intNum := 4/3     % produces an error
```

Conversion Operations - Truncate

Most programming languages provide special commands to convert between real and integer.

The most common conversion is the truncate command, which literally “cuts off” the decimal. This is also the same as always rounding down.

In some languages, this happens automatically after a mathematical operation.

e.g., 3.1415 becomes 3
or 2.718 becomes 2

Conversion Operations - Rounding

In math courses, we learn to round a decimal value up or down depending on the digit.

For 0, 1, 2, 3, and 4, round down
For 5, 6, 7, 8, and 9, round up

Examples - To round to an integer value:

3.14 becomes 3 (the 1 rounds down)
2.718 becomes 3 (the 7 rounds up)

Conversion Operations – Turing

```
var num : int

% declare a constant value for pi
const pi : real := 3.14

num := floor(pi)    % truncated to 3
num := round(pi)    % rounded to 3
num := ceil(pi)     % rounded up to 4

num := floor(2.71) % truncated to 2
num := round(2.71) % rounded to 3
num := ceil(2.71)  % rounded up to 3
```

Integer Division

When first learning division, we generally learn integer division. That is, we work only with whole numbers and a remainder.

This type of division has some programming applications, so most languages include a way to perform integer division.

The most common name for this operation is “modulo”, or the “modulo” operator, and it provides the remainder.

Integer Division – Modulo (“mod”)

What is 7 ÷ 3?

2.333333 or 2 and 1/3 or 2 remainder 1

The modulo operator would allow us to find that remainder of 1.

e.g., 7 modulo 3 gives the value (remainder) 1

Integer Division - Turing

In Turing, there are operators for both the remainder and the whole number (quotient).

```
var quotient : int
var remainder : int

quotient := 8 div 3    % is 2
remainder := 8 mod 3   % is 1

put "8/3 = ", quotient ..
put " remainder ", remainder
```