The C++ Language

Complex Mathematical Expressions

Simple Mathematical Expressions

- We've seen simple mathematical expressions
  - Only two *operands*, data being operated on
  - Only one *operator*, denoting operation to perform
- Special rules
  - Dividing integers
  - Mixing floating point and integer operands
- What about more operands and operators?
- How does mixing data types work?

Complex Mathematical Expressions

- Suppose you have the statement

```
avgTemp = FREEZE_PT + BOIL_PT / 2.0;
```

- Is `FREEZE_PT + BOIL_PT` calculated first?
- Or, is `BOIL_PT / 2.0` calculated first?
- In order to answer the question, *precedence* must be defined for operations.

Complex Mathematical Expressions

- Precedence rules for C++

Order of evaluation	Operations (from left to right)
1)	Expressions in parentheses
2)	Unary +, Unary -
3)	* / %
4)	+ -

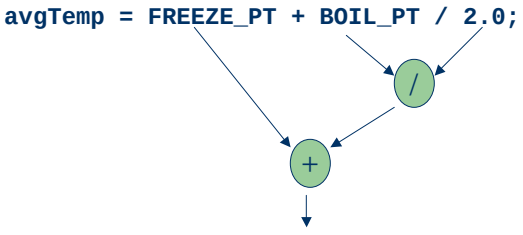
Complex Mathematical Expressions

- According to precedence rules, `BOIL_PT / 2.0` is evaluated first.
  - This probably is not what was intended. Instead, write

```
avgTemp = (FREEZE_PT + BOIL_PT) / 2.0;
```

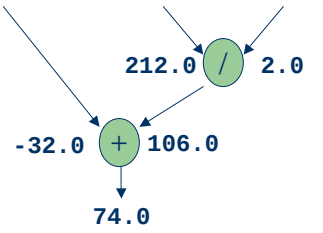

Complex Mathematical Expression

- Using precedence, a complex expression can be evaluated as many simple expressions.
- A useful tool for evaluating complex expressions is an *evaluation tree*.



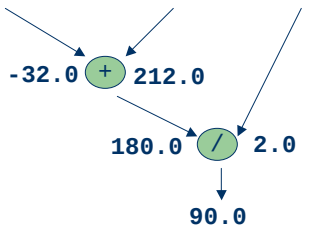
Evaluation Tree (Example)

```
avgTemp = FREEZE_PT + BOIL_PT / 2.0;
```



Evaluation Tree (Example)

```
avgTemp = (FREEZE_PT + BOIL_PT) / 2.0;
```

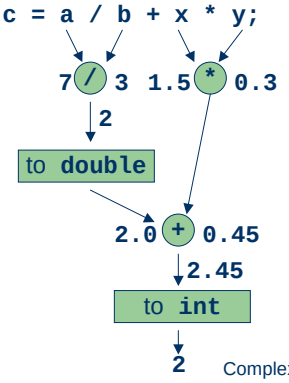


Evaluation Tree (Example and Exercise)

```
int a, b, c;
double x, y;
```

a	b	x	y
7	3	1.5	0.3

```
c = a / b + x * y;
```




```
c = a / (b + x) * y;
```


Complex Expressions (Exercises)

- Write C++ expressions for

$$b^2 - 4ac$$

$$a + b - c$$

$$a \cdot -(b + c)$$

$$\frac{a+b}{c+d}$$

$$\frac{1}{1+y^2}$$

Complex Expressions (Exercises)

- Draw evaluation trees for the following

```
int a, b, c;
double x, y, z;
a = 3; b = 5; x = 1.3; y = 2.7;

z = a * a + b;
c = a - x * b + y;
c = x + a % b;
z = (x + y) / a - b * -y;
```