PH502: Scientific Programming Concepts

Irish Centre for High End Computing (ICHEC)

September 23, 2020

Overview

- In this lecture we will continue with variables.
- We will examine operators and arithmetic.

Arithmetic Operators

- By arithmetic we mean adding, dividing etc.
- There are three subclasses integer, floating point and logical.
- Binary operators are those that operate on two variables *e.g.* $a + b$. The '+' (adding) is a binary operator.
- Unitary operators act on just one variable *e.g.* -10. The '-' operator makes 10 negative.
- When the compiler interprets an arithmetic expression each operator has a precedence. For example when we write $3a + b$ we mean the 3 multiplies the a first, the result is then added to b. Multiplication has a higher priority than addition. When constructing expressions it is important to bear this priority order in mind.

Assignment Operator

- In C and FORTRAN the $=$ operator assigns a variable an $r - value$. It has not quite the same meaning as $=$ in mathematical expressions.
- Below is valid C and FORTRAN code, x, y are integers.

```c
// C Code
x = 1;
y = x;
x = 2;
```

```fortran
! Fortran Code
x = 1
y = x
x = 2
```

- After the first line x has an $r - value$ of 1. The second line sets y to the same $r - value$ as x *i.e.* 1. After the third line x's $r - value$ is 2 but y's is still 1.
- The expression below adds one onto x's $r - value$ (mathematically it makes no sense).

```c
x = x + 1;
```

Integer Arithmetic

- They are used mostly for indices and counters in scientific computing.
- The result of applying the integer arithmetic operators to integers is another integer.
- Integer division: The resulting integer is obtained by discarding the fractional part.
- Modulus operator (%): It evaluates to the remainder obtained after dividing two integers.
- Increment/decrement operators (++/- -): It increases/decreases integer value by one.
    - Prefix form will increment/decrement the value and then return it.
    - Postfix form will return the value first and then increment/decrement it.

Integer Arithmetic

Examples

```
int i,j,k;  float z; // Declarations
i = 3%2;          // Remainder (=1)
j = 10/3;         // Division (=3)
++i; --j;         // Increment (=2) / Decrement (=2)
k = i*j;          // Mult. stay within range (=4)
z = 3/4;          // (=0.0)
z = 3.0/4.0;      // (=0.750000)
```

```
integer (kind=4) ::  i,j,k ! Declarations
real (kind=4) :: z
i = mod(3,2) ! Remainder (=1)
j = 10/3;    ! Division (=3)
i=i+1
j=j-1        ! Increment (=2) / Decrement (=2)
k = i*j;     ! Mult. stay within range (=4)
z = 3/4;     ! (=0.0000000E+00)
z = 3.0/4.0  ! (=0.7500000)
```

Increment/Decrement Operators

- The two forms of the increment and decrement operators, post and pre behave differently.
- The pre operator increments the variable first then applies the expression.
- The post version applies the expression then increments the variable,
- Example

```
int i=1,j;

j = ++i;        // i=2, j=2
j = i++;        // i=3, j=2
```