PH502: Scientific Programming Concepts

Irish Centre for High End Computing (ICHEC)

September 23, 2020

Overview

- This lecture we will discuss casting.
- Casting is a mechanism to change one variable type to another.
- Doing this will ensure that the new variable has the correct representation.
- However there are some considerations.

Changing Variable Type

Casting

- Sometimes it is necessary to change the variable type.
- Lets consider the expression $x = i/j$ where i, j are integers and x is real.

```
i = 2; j = 3;
x = i/j; /* Result 0.0 */
x = (float) i/(float) j;
/* Result 0.66666 */
```

```
i = 2; j = 3;
x = i/j ! Result 0.0
x = real(i)/real(j)
! Result 0.66666
```

- In changing the integer variables to floats the expression is calculated as a real expression hence the correct answer is returned.

Casting in C

- In C you can change one variable into any other variable type, *e.g.* $(int)x$, $(double)x$ or $(char)x$.
- However the limitations of each variable type need to be taken into account.
- Also when converting from a character to integer (say '1') you will get the ASCII table value and not one.
- When converting from double or float to integer, representation errors should be considered. The second line of code returns the number excluding digits after the dot. The third line returns the nearest integer value.

```
int i; double x; char c='1';
i = (int) x;
i = (int) rint(x);
i = (int) c - 48;
```

Casting in Fortran

- There are intrinsic functions that perform the conversions, $int(x)$, $real(x)$ and $dble(x)$ to convert x to integer (kind=4), real (kind=4) and real (kind=8). Like C, numeric ranges need to be considered.

```
integer (kind=4) :: i
real (kind=4) :: x
real (kind=8) :: dx
i = int(dx); dx = dble(x); i = nint(x);
dx = dble(1.0)
```

- The function $nint(x)$ returns the closest integer to x.
- To convert to and from character variables internal read and write statements are used.

Internal Read and Write

Fortran

- Previously we have used "write(6,*)" to print information to the screen.
- Internal write statements allow conversion of strings to numeric variables.

```
character (len=24) :: line
character (len=9) :: s
integer (kind=4) :: i
real (kind=4) :: x
line = ' 1234 1234.456 etc etc.'
!       1   5   10   15   20  24
read(line,'(i5,f9.3,a9)') i,x,s
! now i=1234, x=1234.456, s=" etc etc."
write(line,'(i5,f9.3,a9)') i,x,s
```

- The read statement allows conversion of a character string into other variable types and write is the opposite.