FORTRAN 90: Repetitive Execution

Meteorology 2270

Loops

- One of the original strengths of computer programming.

- Two types of loops
  - Loops controlled by a counter.
  - Loops controlled by a logical expression.

Counter Controlled DO loops

- DO control-variable = initial-value, limit, step-size
  statement-sequence
  END DO
- Initial-value, limit, step-size are integers
- Step-size must be non-zero and may be omitted.
  - If omitted, step-size will be 1.
- Execution of loop

Execution of loop

1. Control variable is initialized.

2. Control variable is compared to limit
   - Control-variable <= limit for step-size > 0
   - Control-variable >= limit for step-size < 0

3. If above statements are true
   1. Execute body of the loop
   2. Add step-size to control-variable
   3. Repeat Step 2

4. If above statements are false, terminate the loop.

Example

- DO number = 1, 9

  Print *, number, number**2

  END DO

  Print *, “Number = “, number

- What should you expect?

- Example: Nested DO loops

General DO loops

- Counter Controlled DO's: number of executions is known before execution of loop begins.
- What if you don't have this information?
  - DO-EXIT construct
  - DO-CYCLE construct
- DO-EXIT construct
  - DO

    statement sequence 1

    IF (logical expression) EXIT

    statement sequence 2

    END DO

General DO's cont.

- If the logical expression never becomes true, an **infinite** loop results.

- Depending on where you put your IF statement, you have either a:
  - Pre-test loop
  - Test-in-the-middle loop
  - Post-test loop

Pre-test/Middle/Post-test

- Pre-test loops
  - DO
    IF (logical-expression) EXIT
    statement sequence
    END DO
  - Statements that follow decision may never be executed.
- Test-in-middle
  - See previous example.
- Post-test loop
  - DO
    Statement sequence
    IF (logical expression) EXIT
    END DO
  - Loop will always execute at least once.
  - Temperature-conversion program

DO-CYCLE construct

- Exit statement causes repetition of loop to terminate by transferring control to the statement following the END DO.
  - What if we want to terminate only the current repetition and jump to the next one?
  - Use DO-CYCLE construct

- Modify temperature conversion program to only process temperatures of 0 degrees Celsius or above.
  - IF (Celsius < 0.0) THEN
    Print *, “*** Temperature must be 0 or above ***”
    CYCLE
    ENDIF