Formatted output

- Lots of I/O related features in Fortran – we don’t need to go into all of them
- Fortran PRINT statments do formatting by preceding the variables to be printed with a “format specification”.
- Format specifier in parentheses.
- “i” for INTEGER; “a” for character; “f” for float (i.e. REAL); “e” for exponential format
- each of these letters must be followed by a field width
- for “f” and “e” the field width must be followed by a dot and the number of characters to be used for the fraction part
- The format specifier may be preceded by a repeat specifier (see example).

Formatted output: example

```
PROGRAM format_spec1
IMPLICIT NONE
  INTEGER :: j = 12345
  REAL :: x = 1.414
  REAL, DIMENSION(5) :: v = (/ 1.1, 1.2, 1.3, 1.4, 1.5 /)

  PRINT '(i5)', j          ! print integer in field width of 10
  PRINT '(f10.3)', x       ! print real in field width of 10 wi
                           ! 3 chars reserved for fractional pa

  PRINT '(5f10.3)', v      ! print 5 reals using above format
  PRINT '(f8.4 i10)', x, j ! print x and then j
END PROGRAM format_spec1
```

Formatted input

- Formatting commands that can be used for PRINT can also be used with “READ”.
- To read formatted input from the keyboard, the syntax is, for example

  ```
  READ ( *, ’(f10.3, i5)’) x, j
  ```
- Usually “READ*, <var_list>” does what is required.

File I/O

- File I/O in Fortran works via “unit numbers”.
- Unit numbers enumerate connections between the program and e.g. files on disk.
- Unit numbers for files are issued by the system using the OPEN keyword.
- I/O is done using WRITE (very much like PRINT) and READ
- WRITE (<unit_num>, <fmt>) vars
- READ (<unit_num>, <fmt>) vars
- Unit numbers are released using the CLOSE keyword.

Output to a file

```fortran
PROGRAM file_output
IMPLICIT NONE
  INTEGER :: u, ios ! unit number, IO status==0 => ok
  REAL, DIMENSION(5) :: v = (/ 1.1, 1.2, 1.3, 1.4, 1.5 /)

  ! ampersand does line continuation
  OPEN(UNIT=u, IOSTAT=ios, FILE='myfile.txt', &
    STATUS='replace', ACTION='write')

  IF (ios == 0) THEN
    WRITE(u, '(5f10.5)') v
    CLOSE(u)
  ELSE
    PRINT '(a25)', 'Error: file not opened.'
  END IF
END PROGRAM file_output
```

Input from a file

```
PROGRAM file_input
IMPLICIT NONE
  INTEGER :: i, u, ios ! counter, unit number, IO status==0
  INTEGER(8), DIMENSION(50) :: xv ! int vector
  REAL(16), DIMENSION(50) :: pv   ! real vector
  REAL(16) :: phi ! var to store golden ratio
  phi = 0.5 * (SQRT(5.0_16) + 1) ! golden ratio
  OPEN(UNIT=u, IOSTAT=ios, FILE='fibdata.txt', &
        STATUS='old', ACTION='read') ! open file
  IF (ios == 0) THEN
    DO i = 3, 50
      READ (u, '(i22, f34.32)') xv(i), pv(i)
    END DO
    CLOSE(u)
    DO i = 3, 50
      PRINT '(i4, e15.5)', i, pv(i) - phi
    END DO
  ELSE
    PRINT '(a25)', 'Error: file not opened.'
  END IF
END PROGRAM file_input
```