032CM Fall 2025 Gianluca Levi (gianluca.levi@units.it)

Assignment 2

Problem 1

(a) The following list contains valid and invalid Fortran constants. State whether or not each constant is valid. If the constant is valid, specify its type. If it is invalid, briefly state why it is invalid.

- `'Distance =`
- `0.5e-3`
- `0.5d-3`
- `.true.`
- `.maybe.`
- `2^3`
- `'Don't panic'`
- `"Don't panic"`
- `19,5`
- `"3.14159"`
- `3.14159`
- `10.`
- `"Pass / Fail"`

(b) For each expression below, determine whether it is valid or invalid in Fortran. If valid, predict the value and type of the result. Verify by writing, compiling, and running a Fortran program that executes the expression.

- `5 ** 2 / 3 + 1`
- `3.(-4. / 2)`
- `(2.) ** real(nint(-2.2))`
- `int(-1.5) ** 2. / floor(2.2)`

(c) Assume a Fortran program with implicit none and these declarations:

```
integer  :: x, y
real     :: r
```

For each variable assignment below, determine if it is valid or invalid. If valid, say what it evaluates to. If invalid, provide a brief explanation.

- `x*y = 2 + 5`
- `xy = 2 + 5`
- `int(2.) = x`
- `r = 7 / 2`
- `r = 7. / 2`

(d) Given the atomic masses m_1 and m_2, the *reduced mass* of a diatomic molecule is

$$\mu = \frac{m_1 m_2}{m_1 + m_2}.$$

Write a Fortran program that reads the masses of the two atoms in atomic mass units, computes the reduced mass μ, and writes out μ in grams. Use meaningful variable names and include comments. What is the reduced mass of the CO molecule?

Problem 2

In binary, many fractions cannot be represented exactly. They are represented as approximations in floating-point format. Fortran can use *single precision* (`real*4`, default) and *double precision* (`real*8`), which can affect numerical results.

(a) Write a Fortran program that evaluates the sum of $\frac{1}{3} + \frac{1}{3}$ in the following three ways:

   (a) `1.d0/3.  + 1/3`
   (b) `1./3.  + 1.d0/3.`
   (c) `1.d0/3.  + 1./3.d0`

   Print the results with at least 15 digits of precision.

   Hint: To print with such precision in Fortran, you can use

```
write(*,'(F25.15)')
```

(b) Compute and print the value of $\frac{2}{3}$ evaluated in double precision.

(c) Compare the three sums from question (a) with the result form question (b). Which results are equal? Can you explain the observed differences?

Problem 3

The *factorial* of a positive integer n is defined as

$$n! = n \times (n-1) \times (n-2) \times \cdots \times 2 \times 1,$$

with the special case

$$0! = 1.$$

Write a Fortran program that asks the user to input an integer n. If $n > 0$, the program should compute $n!$ using a loop. If $n = 0$, the program should return the result $0! = 1$. If $n < 0$, the program should raise an error message and stop execution. Print the result with a clear message, e.g. `Factorial of n is ....`