

PEP is written entirely in Fortran, but it has system dependencies nonetheless. These fall into several categories.

#### NAMELIST INPUT

PEP depends heavily on namelist input and on being able to scan an input file for the relevant markers. Most Fortran implementations recognize the use of "&" as the namelist special character, but some may insist on "\$". In either case, the input stream fed to PEP must always use "&", and PEP will make the conversion, if needed, internally. PEP also makes sure that each line of namelist input containing at least one equals sign ends with either an equals sign or a comma.

Dependency: nmlchrdt.inc

#### SYSTEM TIME INTERFACE

The details are highly system-dependent. PEP expects to get the current date and time occasionally during execution and also to monitor the amount of CPU time used.

Dependency: timdat.f

#### VARIOUS SYSTEM CALLS

PEP has hooks for a variety of systems. The unavailable entry points are included in a catchall dummy subroutine. When porting to a new system, such entries should be commented out if available, and new ones should be added as necessary.

Dependency: dumstf.f

#### FILE ACCESS

PEP does not explicitly open files for input or output. It expects to find the appropriate input files pre-allocated and ready to read with an implicit OPEN at the first read statement. It assumes that output files will similarly be opened and allocated as needed.

#### COMPILER VARIATION

PEP is coded using some extensions from standard Fortran 77. If it is to be ported to a system whose compiler does not support these extensions, a thorough revision should be undertaken to replace those structures. The non-standard features, as determined by the Sun Fortran compiler:

Easily "fixed"

1. lower case letters - the whole program could be upcased
2. IMPLICIT NONE statement - these could be removed

Practically universal

3. INCLUDE statement
4. specification statements after DATA statement
5. initializer in type statement
6. NAMELIST statement (see above)
7. length specification on non CHARACTER type

Very common extensions, but straightforward to "fix"

8. DO ... END DO
9. DO WHILE
10. symbol longer than 6 characters

Hard to do without

11. COMMON with both CHARACTER and nonCHARACTER variables
12. CHARACTER variable equivalenced to a noncharacter variable
13. quadruple-precision constant
14. quadruple-precision builtin functions

## MEMORY MANAGEMENT

On a system with limited memory, PEP can be linked as an overlay program. If so, the two logically separate commons for storing normal equations and other large arrays must be allocated as two \*different\* common blocks /MATZRO/ and /NRMMAT/, rather than just one.

Dependency: matzro.inc