NAME:
AME 20214
Introduction to Engineering Computing
Examination 1
Prof. J. M. Powers
17 October 2013

1. (10) Identify if the following statements are valid or invalid `Fortran` statements, by circling *valid* or *invalid*:

   (a) `y = 1,000.` *valid* *invalid*

   (b) `real ::  2by4` *valid* *invalid*

   (c) `integer ::  pi` *valid* *invalid*

   (d) `complex ::  Frequency` *valid* *invalid*

   (e) `i=i+1` *valid* *invalid*

2. (10) Convert the following mathematical expressions into `Fortran` code. Assume all variables are real.

   (a) $\sin^{-1}(2\pi x)$ *Answer:*

   (b) $\left(e^{|2x|}\right)^3$ *Answer:*

3. (10) If the statement is error free, evaluate the precise numerical value which would be returned by a `Fortran` program. Else, identify the error.

   (a) `0**0` *Answer:*

   (b) `0.**0.` *Answer:*

   (c) `1/2` *Answer:*

   (d) `sqrt(4)` *Answer:*

   (e) `log(1._8)` *Answer:*

4. (10) With an input of $1, 1, 1, 2, 3, 4$ give the output of the following code:

```
program test
read*,x1,y1,z1,x2,y2,z2
x3 = y1*z2-y2*z1
y3= z1*x2-z2*x1
z3 = x1*y2-x2*y1
print 100, 'The answer is ', x3,y3,z3
100 format(a20,f10.3,f10.3,f10.3)
end program test
```

5. (10) Locate syntax and run-time errors, if any, in the following:

   (a)
   ```
   do j=1,10.5,0.5
      print*,j
      end do
   ```

   Answer:

   (b)
   ```
   real :: a(10,1)
    do i=1,10
      a(i) = i**2
      end do
   ```

   Answer:

6. (5) In `UNIX`, identify which command copies a file to another file.

   (a) `cy` *filename1 filename2*

   (b) `dup` *filename1 filename2*

   (c) `cp` *filename1 filename2*

   (d) `mv` *filename1 filename2*

7. (5) In `UNIX`, which command moves you into your home directory?

   (a) `ls`

   (b) `hm`

   (c) `cd`

   (d) `mv`

8. (10) Identify, if any, *all* problems, grammatical, syntactical, and "TeXnical," in the following LaTeX code segment

   ```
   We all enjoy polynomials such as
   \begin{equation}
   $x**2 +x+1$
   \end
   where x is the unknown.
   ```

9. (30) Assume n and m are known integers known to the user at the beginning of any given run, but which can vary from run to run. Assume you have a data file named `data.txt` with nm random integers between -10 and 10 in a list in the following form:

```
-8
9
.
integer data
.
-2
-1
```

Write a short `Fortran` program which

(a) reads the data

(b) structures the data into a matrix of dimension $n \times m$ where the first column is populated by the first n numbers in the list, the second column is populated by the second n numbers in the list, and so on to the m^{th} column.