**Question 1:** (100 points) You are to design and write a Fortran program to perform numerical integration of an arbitrary, real-valued, analytic function. As part of the input to the program, you should ask the user the allowable error in the integration and your algorithm should achieve at least this accuracy. The program you write will have an external subroutine that will return the value of the function for a given value of its independent variable. The function routine is given below and would be used as

```fortran
Real*8 val, afunc
Val = afunc(x)
```

Where all arguments are real*8, x is the independent variable, val is the value of the function.

Your program should take as input the allowable error in the integration, the range of x over which the integration is to be performed and should return

\[
\text{Int} = \int_{x_S}^{x_E} f(x) \, dx
\]

where \(x_S\) and \(x_E\) are the start and end of the integral and \(f(x)\) is the function coded in the afunc subroutine. Optionally, you may also return an estimate of the error in the integral.

You are to:
(a) Design the program addressing issues about implementation and user interface  
(b) Write the program with self-documentation to the user on how to use the program.  
(c) Discuss and show examples of how you tested the operation of your program including trial functions that you tried.

Your submission should the text associated with the designing the program and the Fortran program. The latter you can e-mailed (as a text attachment) to tah@mit.edu. Your program will be compiled, linked and tested. You should use the `afunc` routine below and test your program with different forms of `afunc` uncommented.

```fortran
*----------------------------------------------------------
real*8 function afunc(x)
*
* User supplied routine with function to integrate
*
* PASSED VARIABLES
  real*8 x    ! X coordinate at which function is to be  
              ! evaluated

**** Test program with different lines below uncommented.
C   afunc = x**3
C   afunc = x**4
C   afunc = sin(x**3)
C   afunc = exp(-x**2)

**** That's all
return
```
end