Spring 2006 Fortran f90/f95 Course Homework I

Email the solution programs to vesa.apaja@gmail.com. You can find me in the quest room of the physics department, FL210.

A function $f(x)$ is tabulated from a to b ($a < b$) with constant steps dx and n values x_i. Write a f90/f95 subroutine that computes the numerical derivative from the given table of function values $f(x_i)$ at points x_i. The derivative is computed using the center difference formula

$$f'(x_i) \approx \frac{f(x_{i+1}) - f(x_{i-1})}{2dx} ,$$

if $x_i \in (a, b)$, using

$$f'(x_i) \approx \frac{f(x_{i+1}) - f(x_i)}{dx} ,$$

if $x_i = a$ and finally using

$$f'(x_i) \approx \frac{f(x_i) - f(x_{i-1})}{dx} ,$$

if $x_i = b$. The call to the subroutine should be `call derive(ftab,n,a,b,derftab)`, where `ftab` is the table of function values, `n` is the number of tabulated values, `a` and `b` are the x limits, and `derftab` is output derivative values at the same points as `ftab` is given.

Write also a short main program that tests the subroutine: compare the numerical derivative to a known exact derivative of a few functions.

Following criteria has to be met:

- Use f90/f95 free form syntax, not the old FORTRAN 77 fixed form syntax. If you don't know what this means you are probably writing using a free form already.
- Use `implicit none`
- Use `intent(in)` attribute in the subroutine to make sure you don't accidentally change `ftab`, `n`, `a` or `b`.
- Real variables should be double precision.
- The subroutine call should be exactly of the form given above.
- The subroutine should be in a separate file called `derive.f90`

CONTINUES ON THE NEXT PAGE

- Always use indentation. For example do loops should look like

  ```
  do i = ...
     program lines
     do j = ...
        program lines
        do k = ...
           program lines
        end do
        program lines
     end do
     program lines
  end do
  ```

  Gnu emacs does this for you, just open the file with the .f90 suffix and press the tabulate key on each line (from top to bottom). There is also a toolbar indent that can indent the whole buffer. If you use some other editor you may have to indent manually. Please check out the program examples on the web page.

- Comment the program. You can use your own judgement to deside what to comment: avoid commenting trivial actions that can be directly seen from the code itself. Comments of the style

  ```
  ...
  ! this inserts a to b
  b = a
  ! this calls the derivative subroutine
  call derive(ftab,n,a,b,derftab)
  ```

  are rather uninformative and just make the program longer and less readable.

  CONTINUES ON THE NEXT PAGE

- Keep the formulas in the program lines as close to their mathematical outlook as possible, so that anyone can easily see by just looking at the program what the formulas are. For example, `fxi` can mean $f(x_i)$. Avoid very long names: the variable `number_of_hydrogen_atoms` is certainly easy to associate with number of H atoms, but may result in formulas like

  ```
  total_energy_stored_in_hydrogen_atoms = number_of_hydrogen_atoms* &
         mass_of_hydrogen*speed_of_light**2
  ```

  which, to my opinion, is harder to read than

  ```
  Etot_H = N_H*m_H*c**2
  ```

  or even

  ```
  m = m_H
  E = m*c**2
  Etot_H = N_H*E
  ```

  Fortran stands for "FORmula TRANslation", let's keep it like that.