Department of Physics and Mathematics

Assignment for Computational Physics Project

a component of Module 04307 - Computational Physics Project & Library Research Skills

(This assignment is worth 15% of the total marks for this module)

Word Count –

This assignment should be submitted via Canvas
Monday 6th March 2016 at 2pm *at the latest,*
using the specific coversheet for this assignment, which can be found on Canvas
For the assessment, the required documents are the following: 1) the figure produced during the assignment; 2) the data file produced during the assignment; 3) the python notebook produced, saved and submitted as *.ipynb file.

The goal of this assignment is to verify the following basic skills in python language: 1) read data from a file, 2) manipulate the data, 3) show data in a meaningful plot and 4) write data in an external file.

About half of the elements heavier than iron present in the solar system today were made in stellar explosions by the nuclear r-process. In a time-scale of less than a second, elements like gold or platinum can be created from neutrons, protons and alpha particles. It is still matter of debate in the scientific community about the exact conditions that allow the r-process to happen.

Description of the data files and Tasks

In the files wind_expand.001.dat, wind_expand.003.dat and wind_expand.005.dat results of r-process simulations are presented.
The three files have the same structure:

Column 1: N – number of neutrons
Column 2: Z – number of protons (atomic number)
Column 3: A – number of neutrons and protons (mass number)
Column 4: X – isotopic abundance in mass fraction
Column 5: Y – isotopic abundance in number (e.g., $Y_i = X_i/A_i$)

Here follows the list of the tasks for this assignment:

- Task 1 :

1a) Write a function that open files in the wind_expand*dat format.
You **cannot** change the format of the original files, if this happens, no marks will be given for this task, even if the data are loaded correctly.
1b) By using the function written for task 1a, load the data from the three different files wind_expand.001.dat, wind_expand.003.dat and wind_expand.005.dat into your python notebook.

The assignment with coversheet should be submitted via eBridge (TurnitIn) no later than Monday 6th March 2016 at 2pm. Submissions received after the deadline will be subject to deductions as outlined in the Course Guides and Student Handbooks.

Please note that submissions via eBridge are date and time stamped for this purpose.

You should use the same function three times, once for each file.

- Task 2:

Starting from the three sets of data, calculate the elemental abundances in mass fraction: the elemental abundance is given by the sum of the isotopic abundances with the same number of protons, Z, given in column 2). For this task, use the data provided in column 4 (X) in the files. For each set of data, generate two new lists of the same size: z_element (number of protons which identifying the element) and x_element (elemental mass fraction).

- Task 3:

Make a plot of the three abundance sets, in the same figure:
- plot the abundances x_element in semilog scale with respect to the proton number z_element.
- use different symbols for the three sets
- add meaningful labels for the x and y axes, and in the figure legend. The figure legend should not overlap with the data plotted.
- For the x-axis, limit the plot range between 25 and 83; for the y-axis, limit the plot range between 1.0e-6 and 1.5.
- Highlight the position of the elements Sr (Z=38), Ba (Z=56), W (Z=74), Re (Z=75) and Pb (Z=82), by adding vertical lines at the corresponding positions and the element names in the figure. These should be clearly visible, and not overlap with the data.

- Task 4:

Write in an external file called 'rprocess_data.txt' the data used in previous tasks.
Use the following format for the file:
- first column: z_element
- second column: x_element from wind_expand.001.dat
- third column: x_element from wind_expand.003.dat
- forth column: x_element from wind_expand.005.dat

Add a meaningful header to the file. The file should be human readable, meaning that the data should be written with a defined format for integers (data in z_element) and floats (data in x_element).

Write meaningful comments in your notebook, explaining what you are doing step by step. Up to 5% will be added to the final marks for providing good documentation.

Marking scheme:

- Task 1 (creating a function and reading data): 30%

The score is divided as following:

- 20% task 1a

- 10% task 1b

- Task 2 (manipulating the data and create new arrays/lists): 20%

- Task 3 (plot): 15%

The assignment with coversheet should be submitted via eBridge (TurnitIn) no later than Monday 6th March 2016 at 2pm. Submissions received after the deadline will be subject to deductions as outlined in the Course Guides and Student Handbooks.

Please note that submissions via eBridge are date and time stamped for this purpose.

The score is divided as following:

- make the plot of the right data (mark = 8%);
- add labels and legend (mark = 3%);
- use different symbols for different data (mark=3%)

- Task 4 (writing data): 15%

The score is divided as following:

- 2% write a meaningful header

- 13% write the data required.

- Additional marks (comments, style in plotting and data production): 20%

The score is divided as following:

- 5% if good documentation is given in the notebook.
- 10% good plotting style in Task 3 (e.g., no overlap of legend or text in the plot with data, labels big enough to read).
- 5% data written with a specified format in the file rprocess_data.txt, easy to read.

The marking will be based on:
- the python notebook or the python file (tasks 1-4)
- the figure with the data plotted (task 3)
- the data file (task 4).

The assignment with coversheet should be submitted via eBridge (TurnitIn) no later than Monday 6th March 2016 at 2pm. Submissions received after the deadline will be subject to deductions as outlined in the Course Guides and Student Handbooks.

Please note that submissions via eBridge are date and time stamped for this purpose.