COMPUTING METHODS FOR PHYSICS

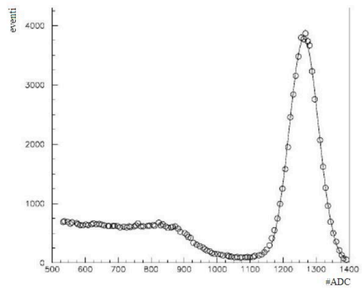
13 JULY 2021

You must submit your exam by **Tuesday Jul 13 at 13:00** following the instruction at http://www.roma1.infn.it/people/rahatlou/cmp/

Compton Scattering

Cesium-137 is a radioactive isotope which decays via beta emission (half life of 30.2 years) to a an excited metastable state of Barium ^{137m}Ba. This state decays with a half-life of 153 seconds to the ground state ^{137}Ba emitting a photon with energy E_0 = 662 keV. The goal is to simulate the impact of Compton scattering on the ^{137}Cs energy spectrum.

1. Generate 10^6 photons with initial energy $E_i = E_0$.
2. Assume that each photon has a 65% probability of undergoing Compton scattering in the crystal.
3. The energy E_f of the photon after the scattering is given by where $m_e = 511\,\mathrm{keV}$ is the electron mass and θ is the photon scattering angle with respect to the initial direction. $$E_f = \frac{E_i}{1 + (E_i/m_e)(1 - cos\theta)}$$
4. The value of angle θ for each photon must be generated according to the angular distribution $$1 + cos^2\theta$$
5. Plot the distribution of generated $cos\theta$ and store the plot as **`costheta.pdf`** .
6. Plot the distribution of E_f for the scattered photons and store the plot as **`compton-truth.pdf`** .
7. The energy of the photons is measured with an NaI calorimeter. Simulate the behaviour assuming an energy resolution of 2.5% to provide the spectrum of the energy spectrum of the observed photons
8. Plot the distribution of measured energy E_f for all photons (with and without Compton scattering) and store as **`compton-truth.pdf`** .
   Make sure reasonable binning are used for the histogram and labels and units are added.
   - You should see a peak around E_0 and a continuous distribution (a Fermi-Dirac shape) for $E_f < E_0$. (See the figure as an example)



- You can use C++, plain python, or jupyter notebook
- Provide the 3 pdf files together with your code
- You must implement at least 3 functions with proper arguments and return types for
    - Compton scattering and calculation of E_f
    - generation of theta
    - emulation of measured energy in the calorimeter
- If you choose python, NumPy must be used properly to emulate all photons with vector calculation
    - C-style loops and iteration are accepted but will result in -2 penalty
- Evaluation will also take into account
    - Choice of arguments, function interface, and return types
    - Correct physics calculations and consistent units
    - Proper labels, legends, and choice of histogram min and max