Excises 1

Applications of Accelerators

WS 2022-2023

Dr. Shahab Sanjari

Name of student: Shymaa Ali Fathi Ali

Date of assignment: 27/09/2022

Date of submission:

Deadline: 05/09/2022

Solve the following questions:

|  |  |
| --- | --- |
| 1. How much voltage is needed to accelerate an electron to energy of 4e-4 [J]? | [1 point] |

Electron mass = 9.1 e-31 kg

Electron charge =1.6 e-19 C

Electron wavelength = 6.4 e-15 m

Convert 4e-4 [J] to [eV] by dividing by the electron charge = 4e-4 / 1.6 e-19 = 2.5 e15 [eV]

Voltage = 2.5 e15 [eV] / 1 = 2.5 e15 [V]

|  |  |
| --- | --- |
| 1. Write in your own words, which force is most suitable for particle acceleration and why? | [1 point] |

There are four fundamental forces in the universe with different ranges and different strengths, the strong and weak forces of them have a short range, therefore, cannot be used to accelerate particles. The gravitational force also cannot be used since it is not strong enough due to the small mass of charged particles. The final force, the electromagnetic force, can be used to accelerate particles because it has a long range of effects and acts on the charged particles.

|  |  |
| --- | --- |
| 1. Warmup computational exercise:  * Create a GitHub account. * Write a function in Python code that multiples 2 (two) 3x3 Matrices (array) and returns the result. * Call this function using two randomly generated 3 by 3 matrices and print the results. * Place your code in your GitHub repository and submit the link in your exercise document. The code can also be in the form of a Jupiter Notebook on your GitHub account. | [2 points] |

GitHub repository link: https://github.com/ShymaaAli/Applications-of-Accelerators-Exercises

|  |
| --- |
| #!/usr/bin/env python  #Shymaa Ali Fathi.  #Applications of accelerators exercise solution.  #EX1.  #Write a function in Python code that multiples two 3x3 Matrices (ndarray) and returns the result.  #Call this function using two randomly generated 3 by 3 matrices and print the results.  import numpy as np  #creating a function to multiply two 3x3 arrays, function with arguments  def multiples\_arrays(arr1,arr2):  task=np.multiply(arr1, arr2)  return task  #Define the generator needed to randomly generate numbers  rng = np.random.default\_rng()  #Generating a 3x3 array of random numbers  arr1=rng.random((3,3))  print ("array1",arr1)  arr2=rng.random((3,3))  print ("array2",arr2)  #call the multiples\_arrays function  result=multiples\_arrays(arr1,arr2)  print ("result of multiplication",result)  print ("End of EX1") |