Instruction manual for using coil definitions

* Enter values for liftoff, coil radius, add any separation between any of the wires if necessary
* Define a width for the central coils which defines how wide the adjacent zeroes are to the central peak defined.
* Define n1, n2 which defines the coil array matrix. EG a 5 x 5 array of coils.
* Define the Wire position by using the Position\_Definer function. The function takes coil radius, n1 and n2 which setup a coil array grid centred around the origin with a list of the points in metres recorded.
* Setup an observer position list, defines our focussing points and define their location, around the origin with some of the points as a function of width. E.G. So coil centred on (0,0), but adjacent zeros have position (-width/2,0),(width/2,0),(0,-width/2),(0,width/2).
* Initialise an empty list which will be the predefined magnetic field.
* Set a focus value of the chosen magnitude, eg: 1E-7 Tesla
* Set a value for the index in the observer position list of which the target position of the focussing is. E.G I have 25 target points and I want to set the focus value on the 13th position which is the origin, the value is 13.
* Set a Target plane value so it is the Bx, By or Bz value where the focussing occurs. If Bx set to 0, if By set to 1 and if Bz set to 2.
* Fill the empty list of predefined magnetic field with 0, the length is 3 times the length of the observer position due to the individual components of x,y,z.
* Calculate the new field index of our focus value so it can be set. This is (Length of the observer field x Target Plane Value) + observer position target index – 1. Make sure this is an integer value.
* Set the predefined magnetic field to the focus value at the index value calculated in the previous point.

For a 1D scan

* Define a linespace, or magnetic field probing region for the field to be calculated on.

If the coil is circular:

* + Define an A,Ax,Ay and Az matrix using the Position\_Tensor function using the observer\_position, wire\_position, liftoff and coil radius arguments.

If the coil is square:

* + Define A,Ax,Ay and Az matrix using Position\_Tensor\_square function using the observer\_position,wire\_position,liftoff, and define L\_a and L\_b where L\_a and L\_b are set as the lengths of the sides of a square in metres.
* Use current\_finder function with A, B ( B is the predefined field from earlier) the Currents can be found for a defined set focused points and wire positions at a set liftoff.
* Define B,Bx,By,Bz using the Bfield function, for a circular coil use the arguments as the linespace, wire\_position, liftoff, coil\_radius and the Current from the previous plot. For a square plot use the arguments as linespace,wire\_position, liftoff (z position), set coil\_radius value as zero, then Current from previous function. Additionally set the shape to ‘square’ as the default setting is circular. Then set L\_a and L\_b to the values as used in the position tensor.
* Find the normal of the Bx,By and Bz and define it by squaring each element in the array add them element wise and square root them element wise.
* Plot the normal of the field and the linespace motion in 1D.
* Use the peak finder in python scipy.signal library to detect peaks. Plot the peaks to see whether it is working.
* Using the peak widths function from scipy.signal find the FWHM (Full width half maximum) values.
* Using plt.hlines functions plot the FWHM and store the real widths of the central peak and peak to peak ratio values.

For a 2D scan:

* Setup a 2D linescan as a list of points in a 2D plane at a set liftoff, use the number of points in each axis for how the 2D shape will be defined.
* Follow the same setup requirements as for square and circular coils as described above
* After finding the magnetic field and setting up individual component values. Use the Bfield plotter function to turn the values into a 2D plot in each dimension.