# Flow Chart 8

## Exercise 6.4

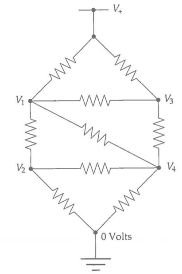
- Use function solve from numpy.linalg to solve exercise 6.1

$$2w + x + 4y + z = -4$$

$$x = -4 - 2w - 4y - z$$

**Exercise 6.1: A circuit of resistors**

Consider the following circuit of resistors:




All the resistors have the same resistance $R$. The power rail at the top is at voltage $V_+ = 5$ V. What are the other four voltages, $V_1$ to $V_4$?

To answer this question we use Ohm's law and the Kirchhoff current law, which says that the total net current flow out of (or into) any junction in a circuit must be zero. Thus for the junction at voltage $V_1$, for instance, we have

$$\frac{V_1 - V_2}{R} + \frac{V_1 - V_3}{R} + \frac{V_1 - V_4}{R} + \frac{V_1 - V_+}{R} = 0,$$

or equivalently

$$4V_1 - V_2 - V_3 - V_4 = V_+.$$

a) Write similar equations for the other three junctions with unknown voltages.

b) Write a program to solve the four resulting equations using Gaussian elimination and hence find the four voltages (or you can modify a program you already have, such as the program `gausselim.py` in Example 6.1).

$V_1$

$$4V_1 - V_2 - V_3 - V_4 = 5 \qquad V_+ = 5$$

$V_2$

$$\frac{V_2 - V_1}{R} + \frac{V_2 - V_4}{2} + \frac{V_2 - 0}{R} = 0$$

$$-V_1 + 3V_2 - V_4 = 0$$

$V_3$

$$\frac{V_3 - V_1}{R} + \frac{V_3 - V_4}{R} + \frac{V_3 - V_+}{R} = 0$$

$$-V_1 + 0 + 3V_3 - 1 = 5$$

$V_4$

$$\frac{V_4 - V_1}{R} + \frac{V_4 - V_2}{R} + \frac{V_4 - V_3}{R} + \frac{V_4 - 0}{R} = 0$$

$$-V_1 - V_2 - V_3 + 4V_4 = 0$$

# Flow Chart

Define the resistance & Vt

↓

Create the matrix using the equations for the junctions

↓

Define the solution for $Ax = V$ (so $V$)

↓

Use function solve from numpy.linalg to solve it.

↓

Print the four voltages

# Exercise 6.9

$\hat{H} = \frac{-\hbar^2}{2M} \frac{d^2}{dx^2} + V(x)$ , zero outside the wall
go to 0 at $x=0$ and $x=L$

Fourier sine series $\psi(x) = \sum_{n=1}^{\infty} \psi_n \sin \frac{\pi n x}{L}$

| Define Constants & create the matrix |
|---|

↓

| Use nested for loops & if statements to see what equation will be use depending on whether m=n |
|---|

↓

| Calculate the eigenvalue using np.linalg.eigvalsh |
|---|

↓

| Repeat the same for c and d with 10×10 matrix & 100×100 matrix corresponding to each. |
|---|

↓

| Extract the eigenvectors corresponding to groud, first excited & second excited states. |
|---|

↓

| Generate the x-values & evaluate the wavefunctions for each state. |
|---|

↓

| Plot the probability densities as a function of x in each state |
|---|