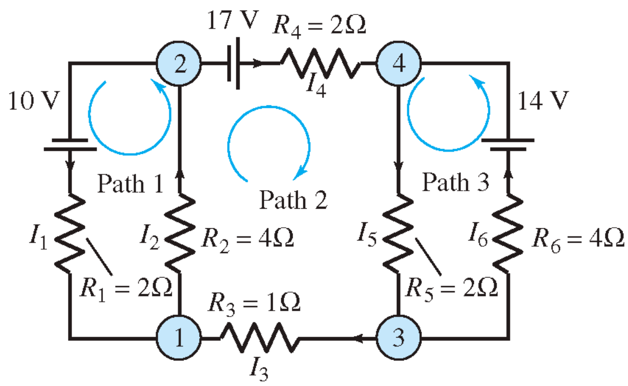
Assignment 1
Introduction and fundamental subspaces

To do

1. The figure below shows an electric circuit, with three power sources and six resistors. There are four nodes (numbered 1-4) and three paths in the circuit. The corresponding equation system to the right can be used to calculate the currents $I_1 - I_6$. The numbers 10, 17 and 14 in the right-hand-side is the voltage in the three power sources.



$$
\begin{cases}
2I_1 + 4I_2 & = 10 \\
4I_2 + I_3 + 2I_4 + 2I_5 & = 17 \\
2I_5 + 4I_6 & = 14 \\
I_1 - I_2 + I_3 & = 0 \\
I_1 - I_2 + I_4 & = 0 \\
I_3 - I_5 + I_6 & = 0 \\
I_4 - I_5 + I_6 & = 0
\end{cases}
$$

(Where the equation system come from is not necessary to know, but a derivation can be found last in this assignment).

a. Write a Python (or Matlab or other language) program that computes the currents I_1-I_6 and plot them in a histogram. You can use built-in tools in Python (you don't need implement a solver for the equation system). It must be easy to change the voltages in the program. Run the program for a few different voltage choices.

b. Here are a few theoretical questions to answer linked to the equation system:
   i. What is the rank of the matrix?
   ii. What is the dimension of the four fundamental subspaces $\boldsymbol{C}(A), \boldsymbol{N}(A), \boldsymbol{C}(A^T)$ and $\boldsymbol{N}(A^T)$ in this case?
   iii. Which subspace does the solution vector belong to?
   iv. Is the solution to the equation system unique given a certain right-hand-side? Motivate you answer.
   v. Which subspace does the right-hand-side vector belong to?
   vi. Is it possible to find a right-hand-side where no solution exists? If so, exemplify and figure out which subspace that right-hand-side vector belong to.

2. The matrix-matrix multiplication $C = AB$, where A is $m \times p$ and B is $p \times n$, can be organized in different ways. The usual way to do it when you do hand calculations is based on repeated dot-products, but it can be structured in other ways. Below are the basic algorithms for the dot-product way and one alternative way:

`for i=1 to m`
` for j = 1 to n`
`  for k = 1 to p`
`   C(i,j)=C(i,j)+A(i,k)*B(k,j)`
	`for k=1 to p`
	` for j = 1 to n`
	`  for i = 1 to m`
	`   C(i,j)=C(i,j)+A(i,k)*B(k,j)`

Note, the only difference is the permuted loops.

a. Implement the two versions in Python (or Matlab) and compare the cpu-time for each version. Are there any differences in the time it takes to do the calculations?
   You can create the matrices A and B as random matrices. Implement the two algorithms in two different functions and calculate the time it takes for the function to run. In Python, use **`time.process_time`** (you might have to import **`time`**). In order to calculate cpu-time that you can trust, you'll need choose matrix sizes that takes at least order of seconds to calculate (let's say around 30 seconds).

   Note 1: It's not necessarily a significant time difference between the two implementations (it might be dependent on programming language)
   Note 2: It's perfectly fine to use other languages than Python here, like Julia or Matlab. It's also perfectly fine to compare several languages.

b. The leftmost algorithm is based on repeated dot-products, but what basic operation is the 2nd (rightmost) algorithm based on?

Where does the equation system in Q1 come from?
You don't need this to solve the assignment, but it might be interesting to understand where the equation system in task 3) comes from anyway. To derive the equation system, we use
1) *Ohm's law*, *U=RI* (in the resistors), where *U* is the voltage drop in the resistor
2) *Kirchoffs 1st law*, the sum of the currents arriving at a node is equal to the sum of the currents leaving the node.
3) *Kirchoffs 2nd law*, the voltages in a closed-circuit loop are equal to 0.

In the three circuits, path 1-3, the law 3) together with law 1) gives equation (1) to (3)

$$\begin{cases} 2I_1 + 4I_2 = 10 & (1) \\ 4I_2 + I_3 + 2I_4 + 2I_5 = 17 & (2) \\ 2I_5 + 4I_6 = 14 & (3) \end{cases}$$

In the nodes 1- 4, law 2) yields equation (4) to (7)

$$\begin{cases} I_1 + I_3 = I_2 & (4) \\ I_1 + I_4 = I_2 & (5) \\ I_3 + I_6 = I_5 & (6) \\ I_4 + I_6 = I_5 & (7) \end{cases}$$

Combine these two and you get an equation system with 7 equations and 6 unknowns.