

## Task 1c, calibrations

```
In [ ]: import numpy as np
import matplotlib.pyplot as plt

from scipy.optimize import curve_fit
```

```
In [ ]: stepValues = [10, 15, 20, 25, 100, 500, 1000]
correctionValues = [900, 400, 250, 150, 30, 25, 25]

correctionValuesBySteps = [a / b for a, b in zip(stepValues, correctionValues)]
```

```
In [ ]: def ExpFunc(x, a, b, c):
    return a * np.exp(-b * x) + c

pars, cov = curve_fit(ExpFunc, stepValues, correctionValues)
print(pars)
```

```
[8106.82039178 -58.0198913 ]
```

```
In [ ]: plt.scatter(stepValues, correctionValues, color="indianred", zorder=5)

xRange = np.linspace(10, 1000, 1000)

plt.plot(xRange, ExpFunc(xRange, pars[0], pars[1], pars[2]), color="cornflowerblue")

plt.xlabel("Number Of Steps")
plt.ylabel("Delay Correction [us step-1]")
#plt.xlim(0, 200)
```

```
Out[ ]: Text(0, 0.5, 'Delay Correction [us step-1']')
```

