Computer Methods in Engineerings
Exercise on gradient descent

Problem 1
Consider the one dimensional function

$$f(x) = 2x^3 + 2x^2 - 4x \quad \text{for } x \in [-1, 1]. \tag{1}$$

a) Find the roots of $f(x)$ and check your result using the numpy `roots()` function.
Hint: The `roots()` function take the polynomial coefficients as a vector input, e.g. for $h(x) = 2x^2 - 1$ you would write: `np.roots([2.0,0.0,-1])`.

b) Plot $f(x)$ in python using `plot()`. Use an appropriate sampling interval Δx on the x-axis. Give your plot a title and label the axes. Indicate the roots with a red dot and blue dot. Plot also the line $y = 0$

c) Use Netwon's method to find a root of the function f. Suggested starting-points are 0.5 and 0.75. As there are several roots, explain why you get the root you get when using the Netwon method.

d) Use Netwon's method to find a maximum or minimum value of the function f. Suggested starting-points are somewhere in the range $[-0.5, 0.0]$. Explain why you get the solution you get when using the Netwon method and your chosen starting-point.

Problem 2
We will now continue with the exercise from the bracket search, where we assumed we were going to build a pipeline to transport CO2 from a capture site to an offshore storage site. A code for calculating the profit from the pipeline as a function of the radius of the pipe was given in the previous exercise.

a) Create a code to find the gradient of the profit with respect to radius.

b) Use the derivate to make a more efficent bracket search. Compare the number of iterations with the barcket search conducted in the previous exercise.

c) Create a code where you do a gradient descent to find the optimum.