NUMERICAL OPTIMISATION
TUTORIAL 2

MARTA BETCKE
KIKO RUL·LAN

EXERCISE 1

(a) Code backtracking line search, steepest descent and Newton's algorithms. See Cody Courseworks for more guidance.
Submit your implementation via `Cody Coursework`. **[30pt]**

(b) Apply steepest descent and Newton's algorithms (with backtracking line search) to minimise the Rosenbrock function

$$f(x) = 100(y - x^2)^2 + (1 - x)^2.$$

Set the initial point $x_0 = (1.2, 1.2)^T$ and the initial step length $\alpha_0 = 1$. Plot the step sizes used by each method over the iterations as well as the trajectories traced by the iterates in $\mathbb{R}^2$. Try explaining the trajectories.
Submit solution via `TurnitIn`. **[40pt]**

(c) Redo the calculations in b) with the more difficult starting point $x_0 = (-1.2, 1)^T$ and explain the trajectories.
Submit solution via `TurnitIn`. **[10pt]**

(d) Repeat the calculations in b) and c) using the line search in Algorithm 3.5 from Nocedal, Wright. This line search produces step lengths which satisfy the strong Wolfe conditions. Use the implementation provided in Moodle: `lineSearch.m`, `zoomInt.m`. Compare the new step lengths with those obtained with backtracking.
Submit solution via `TurnitIn`. **[20pt]**

Assignment Project Exam Help
https://powcoder.com
Add WeChat powcoder

Remark. The submission to `TurnitIn` should not be longer than 4 pages. Avoid submitting more code than needed (if any) and focus on explaining your results.