Assignment #3 (*REVISED*)

Due Wednesday, 28 September 2022, at the start of class

From the textbook[1] please read sections 2.4, 2.6, and 3.1, plus Appendices B.4 through B.8. We will get back to sections 2.5 ("rates of convergence") and 2.7 ("Newton's method for nonlinear equations") soon enough, but we will be pushing toward the simplex method for linear programming in the shorter term (Chapters 3,4,5).

DO THE FOLLOWING EXERCISE from section 2.4, page 58:

- Exercise 4.3

DO THE FOLLOWING EXERCISES from section 2.6, page 66:

- Exercise 6.5

DO THE FOLLOWING EXERCISES from section 3.1, page 82:

- Exercise 1.1
- Exercise 1.3

Problem P8. *This problem considers a simple,* **and totally inadequate**, *algorithm to illustrate "General Optimization Algorithm II" on page 55. We will do much better soon, but the reason I am asking this weird question is that some of the difficulties seen here will remain for all time.*[2] *This exercise asks you to analyze the deficiencies of this particular algorithm,* not *propose a better one!*[3]

Consider one-dimensional unconstrained optimization problems

$$\min_{x\in\mathbb{R}} f(x)$$

for objective functions f which have one continuous derivative f'. I propose the following algorithm:

> **Algorithm P8A.** Assume functions $f(x)$ and $f'(x)$ are supplied.
> 1. Set $x_0 = 1$.
> 2. For $k = 0, 1, 2, \dots$
>   - (i) If $|f'(x_k)| < 10^{-3}$ then stop.
>   - (ii) If $f'(x_k) > 0$ then let $p_k = -1$. Otherwise let $p_k = +1$.
>   - (iii) Let $\alpha_k = 0.01$. Let $x_{k+1} = x_k + \alpha_k p_k$.

[1] Griva, Nash, and Sofer, *Linear and Nonlinear Optimization*, 2nd ed., SIAM Press 2009.

[2] Optimization algorithms always require the user to know *some* properties of their problem.

[3] I would, in fact, expect that you can propose a better one.

a) Implement Algorithm P8A. In MATLAB it will be a function

```
function z = p8a(f,df)
```

where the inputs are functions f = `f` and f' = `df`. The output `z` is the proposed x-coordinate of the solution, i.e. it is the minimizer.

b) Run the algorithm, and state briefly what happens, in the following cases:
(i) $f(x) = x^2 - 3x + 2$
(ii) $f(x) = \cos(x/50)$
(iii) $f(x) = e^{\sin(10x)}$
(iv) $f(x) = \operatorname{sech}(x)$

c) In several complete sentences, describe the main deficiencies of Algorithm P8A. It is a good idea to use the results of part **b)** to illustrate some of your points.

d) Next, observe that there are magic, fixed numbers inside Algorithm P8A which we can instead put under the control of the user. This gives a new algorithm:

> **Algorithm P8B.** Assume functions $f(x)$ and $f'(x)$ and numbers $x_0 \in \mathbb{R}$, $\epsilon > 0$, and $\delta > 0$ are supplied.
> 1. (*The user has supplied* x_0.)
> 2. For $k = 0, 1, 2, \dots$
>   (i) If $|f'(x_k)| < \epsilon$ then stop.
>   (ii) If $f'(x_k) > 0$ then let $p_k = -1$. Otherwise let $p_k = +1$.
>   (iii) Let $\alpha_k = \delta$. Let $x_{k+1} = x_k + \alpha_k p_k$.

There is no request to implement P8B. However, in MATLAB it would be a function like

```
function z = p8b(f,df,x0,eps,delta)
```

In complete sentences, address whether Algorithm P8B is significantly better than P8A. Specifically, consider what the user needs to know about $f(x)$ in order to use P8B effectively.