Brief Answers to Quiz 02

1:20-2:10PM, Oct 16, 2020.

1. Find the smallest N so that $\left|\sum_{i=0}^{N}\frac{3^i}{i!}-\exp(1)^3\right|<10^{-5}$. Let your code print the answer N and $\left|\sum_{i=0}^{N}\frac{3^i}{i!}-\exp(1)^3\right|$ on screen, and also write them down on the answer sheet. Type the error bound 10^{-5} directly in your code, and do not use it as an input. Extra credits for more efficient method(s).

   Name your code using your student ID number, quiz number and problem number, such as m107000001_q2p1.m or m108000002_q2p1.c. Extra credits using C, C++, etc.

   Ans:

   $N = 15$ **(5 pts)**
   absolute error = 2.492E-06 **(5 pts)**
   Code **(10 pts)**
   Extra points by using C **(3 pts)**
   Extra points by nested summation **(5 pts)**

2. Consider the following recursive formula $p_1 = 1, \quad p_2 = a, \quad p_n = \frac{10}{3}p_{n-1} - p_{n-2}$, used as an algorithm to compute p_N for a given N. For what values of a is this algorithm stable in relative error? Explain.

   Ans:

   (a): Exact solution is given by

   $$p_n^{\text{exact}} = c_1\left(\frac{1}{3}\right)^n + c_2 3^n$$

   where $c_1 = \frac{81-27a}{24}$, $c_2 = \frac{3a-1}{24}$ (**10 pts**)

   (b): Relative error $= \left|\frac{e_N}{p_N^{\text{exact}}}\right|$.

   Note that

   $$e_n \approx d_1\left(\frac{1}{3}\right)^n + d_2 3^n$$

   Both d_1, d_2 are of $O(\varepsilon_M)$

   Therefore, relative error $= \frac{O(\varepsilon_M)\left(\frac{1}{3}\right)^n + O(\varepsilon_M)3^n}{c_1\left(\frac{1}{3}\right)^n + c_2 3^n}$. (**5 pts**)

   It is stable if and only if $c_2 \neq 0$, or $a \neq 1/3$. (**5 pts**)

3. The file an.txt contains a sequence stored as $'n, a_n'$ at nth line and its limit $L = \lim a_n$ in the comment line. Find its rate of convergence. Express your answer as $O(\beta_n)$ and find β_n explicitly. Extra credits for finding the rate without knowing L. Explain how you get the answer and hand in your code.

Ans:

Method 1:

Try semilogy or loglog plot of $|a_n - L|$ vs n (where $L = \lim a_n$) to determine whether $|a_n - L| \approx Cn^{-p}$ or $|a_n - L| \approx C\alpha^{-n}$ or something else.

Method 2 (Full credit + extra 10 pts):

Proceed to find out the constants C, p or C, α.

Directly try $|a_n - L| \approx Cn^{-p}$ and find p through

$$p \approx \log_2 \frac{a_N - a_{2N}}{a_{2N} - a_{4N}}$$

Different choices of $N = 30, 50, \cdots, 100, 125$ all give consistent answer $p \approx 1$.

Ans: $a_n - L = O(\frac{1}{n})$

4. Find a root of $x = 2\cos x$ with 10 correct decimal digits using any numerical method of your choice. Write down (1): the detail formula of your method, (2): x_0, x_1 and (3): the answer x^*, on the answer sheet. Double check that your answer indeed satisfies the equation before you write them down. Need not hand in the code if you are sure about your answer.

   Ans:

   Correct formula of the chosen method: **(5 pts)**

   Correct answer $x^* \approx 1.029866529$ **(15 pts)**

5. Show that the nonlinear equation $x = 1 + \frac{1}{2}\cos(x)$ has a solution in $[1, 1.5]$. Let $x_0 = 1.25$, give an estimate on N (need not be optimal) such that $|x_n - x^*| < 10^{-5}$ for all $n \geq N$. You can use any theorem(s) on the textbook directly and need not prove the theorem(s) in advance, as long as you state the theorem(s) correctly.

   Ans:

   Existence proof: Show that $g[1, 1.5] \subset [1.1/5]$ and $|g'(x)| \leq k < 1$ on $[1.1/5]$. (**5 pts**);

   Correct error estimation: **(10 pts)**;

   Correct N resulting from the estimate: **(5 pts)**.

   Possible estimate include:

   Fixed point iteration: Let $g(x) = 1 + \frac{\cos x}{2}$. Then $|g'(x)| = |-\frac{\sin x}{2}| \leq \frac{1}{2} =: k$.

   Estimate 1:

$$|x_n - x^*| \leq k^n \max\{x_0 - a, b - x_0\} = \frac{1}{2^{n+2}} < 10^{-5}$$
$$\Rightarrow n \geq 14.60... \Rightarrow N = 15.$$

   or

   Estimate 2:

$$|x_n - x^*| \leq \frac{k^n}{1-k}|x_1 - x_0| = \frac{1}{2^{n+1}}|2\cos(1.25) - 1| < 10^{-5}$$
$$\Rightarrow n \geq 14.17... \Rightarrow N = 15.$$

Bisection: Use Theorem 2.1 to estimate $|x_n - x^*|$.

Newton's Method: Use $x_{n+1} - x^* = \frac{g''(\xi_n)}{2}(x_n - x^*)^2$ (section 2.4) to give an estimate.

Any method with correct estimate will receive full credits for estimate.

Direct simulation to give N will get no credits for estimate and no credits for N.