Queens College, CUNY, Department of Computer Science

Numerical Methods

CSCI 361 / 761

Fall 2018

Instructor: Dr. Sateesh Mane

Quiz 4

Monday November 19, 2018 (in class)

Take home, grade boost

- **Due Saturday November 24, 2018 11:59 pm**
- **Students qualify for a grade boost only if the code used for the in-class exam is submitted (via email) before Tuesday November 20, 2018, 12 noon.**
- **Submit your in-class program code via email, as a file attachment, to** `Sateesh.Mane@qc.cuny.edu`**.**

  `StudentId_first_last_CS361_quiz4_Nov2018.[cpp,java]`
  `StudentId_first_last_CS761_quiz4_Nov2018.[cpp,java]`

- **NOTE: It is the policy of the Computer Science Department to issue a failing grade to any student who either gives or receives help on any test.**
- **A student caught cheating on any question in an exam, project or quiz will fail the entire course.**
- This is an **open-book** test.
- Once you leave the classroom, you cannot come back to the test.
  (Not applicable for take home grade boost.)
- **Any problem to which you give two or more (different) answers receives the grade of zero automatically.**
- **Answers must be written legibly (preferable typed in docx or pdf).**
- **A failing grade will be awarded if the examiner is unable to decipher your handwriting.**
- **Submit your solution, including your program code via email, as a file attachment, to** `Sateesh.Mane@qc.cuny.edu`**.**

  ```
  StudentId_first_last_CS361_quiz4_grade_boost.zip
  StudentId_first_last_CS761_quiz4_grade_boost.zip
  ```

- **Put all your code in one file (C++ or Java).**
- You may employ the program code in the online lecture notes, else write your own code.
- **Programs which display any of the following behaviors will receive a grade of F:**
  1. Programs which do not compile successfully (non-fatal compiler warnings are excluded, e.g. use of deprecated features).
  2. Array out of bounds, reading of uninitialized variables (including null pointers).
  3. Operations which yield NAN or infinity, e.g. divide by zero, square root of negative number, etc. *Infinite loops.*
  4. Also, all debugging statements (for your personal testing) should be commented out.

1 Question 1

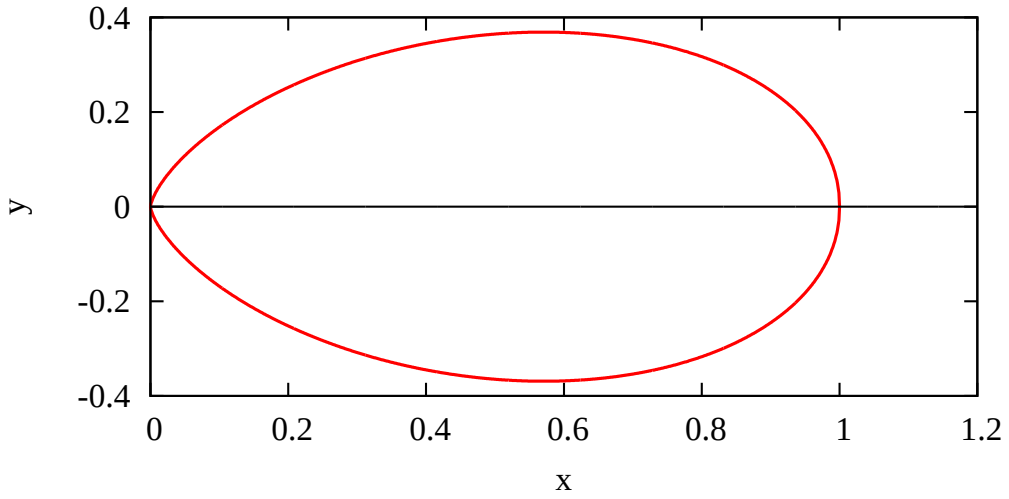

Figure 1: Graph of y as a function of x in Question 1.

- The following curve is plotted in Fig. 1, for $0 \leq x \leq 1$:

$$|x|^{2.1} - |x|^{1.5} + |y|^{2.1} = 0\,.$$

- **Calculate the area enclosed by the curve using (i) midpoint rule, (ii) Simpson's rule, (iii) extended trapezoid rule, (iv) Romberg integration.**

  1. By symmetry, we calculate the area in the upper half-plane ($y \geq 0$) and multiply by 2.
  2. **Express y as a function of x for $0 \leq x \leq 1$:**

$$y = \text{function of } x \qquad (y \geq 0)\,.$$

  3. Then calculate the area via

$$A = 2\int_0^1 y\,dx\,.$$

- **Fill the following table with numbers to 4 decimal places.**

j	$n = 2^j$	Midpoint	Simpson	Extended Trapezoid E_j	$R(j,1)$	$R(j,2)$
0	1	4 d.p.	n/a	4 d.p.	n/a	n/a
1	2	4 d.p.	4 d.p.	4 d.p.	4 d.p.	n/a
2	4	4 d.p.	4 d.p.	4 d.p.	4 d.p.	4 d.p.
3	8	4 d.p.	4 d.p.	4 d.p.	4 d.p.	4 d.p.
$\vdots$	2^j	4 d.p.	4 d.p.	4 d.p.	4 d.p.	4 d.p.

- **Calculate the area A to an accuracy of 4 decimal places.**
  That is to say, stop when the results stop changing to 4 decimal places.

2 Question 2

- **Solutions to this question which employ commercial softwares such as MATLAB or Wolfram Alpha, etc. will receive zero credit.**
- **Write a value of** n where the answer in Question 1 was accurate to 4 d.p. Call it n_*.
- Use the value n_* for all the calculations in this question.
- **Write the value of n_* that you employ to answer this question.**
- Change the equation of the curve to the following:

$$|x|^{2.1} - |x|^{\beta} + |y|^{2.1} = 0\,.$$

- **Define A_* as follows:**

$$A_* = 1 + (\text{your student id}) \times 10^{-8}\,.$$

- **For example if your student id is 23456789 then $A_* = 1.23456789$.**
- **Find a value of β such that the area enclosed by the curve equals A_* to 3 d.p.**

$$|A(\beta) - A_*| < 10^{-3}\,.$$

1. You will probably need to iterate the value of β.
2. You do not need to write a formal bisection of Newton-Raphson program.
3. Just try values of β "by hand" and fill the table below with your iterates.

i	β	Area $= A_i$	$\lvert A_i - A_*\rvert$
0	β_0		
1	β_1		
2	β_2		
$\vdots$	$\vdots$		
i	β_i		stop when $\lvert A_i - A_*\rvert < 10^{-3}$