# AMS326 (Numerical Analysis) 2025

Yuefan Deng | yuefan.deng@stonybrook.edu

## Final: 05/13/2025 5:30-8 pm

**Notes:**

1. *You earn 22.5 Points for doing any 3 of the 4 problems correctly.*
2. *You need to compose a self-contained report for each problem, as you did for HW sets.*
3. *You may use any language, e.g., Python, C, C++, Fortran, Java, MATLAB, etc.*
4. *You may use any external resources including programs or AI tools like ChatGPT or Grok or Gemini as long as you quote the sources.*
5. *You may use any computer systems as long as you can e-submit your solutions.*
6. *You are not allowed to collaborate with anyone inside or outside the class.*

--- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- --- ---

**Problem F.1** (7.5 points) We know by some "magical" calculations that

$$I = \int_{-1}^{1}\int_{-1}^{1} u(x,y)dxdy \approx 4.028423$$

where the integrand is

$$u(x,y) = e^{(x^6 - y^4)}$$

which looks like

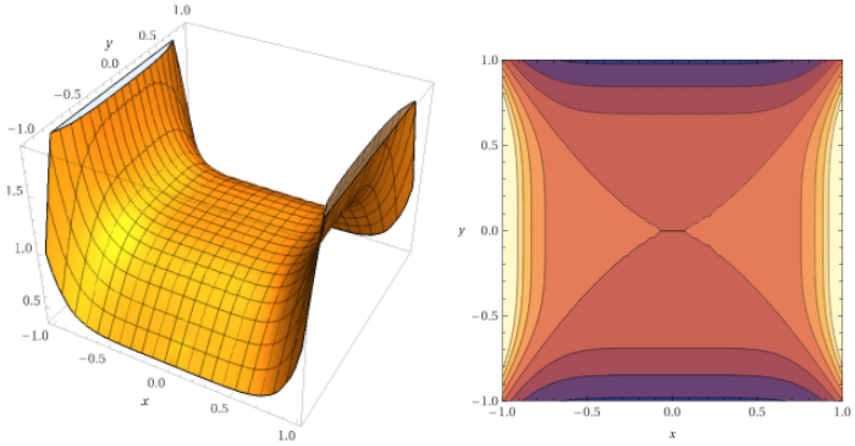


**Figure 1**. The integrand for this problem: the 3D plot (left) and the contour plot (right).

Please compute the integral by writing programs for two implementations of the Monte Carlo method with $N$ samples and other given parameters/conditions. You need sufficient programming details in your implementations.

| Implementation | N | Your Integral | Your Error* | Credits to Give |
|---|---|---|---|---|
| **#1**: Using samples drawn randomly and uniformly in $[-1,1] \times [-1,1]$ | $10^6$ | | | 3.0 |
| **#2**: Using samples drawn randomly with density proportional to $\lvert\nabla u(x,y)\rvert$ in $[-1,1] \times [-1,1]$ | $10^6$ | | | 4.5 |

**Note**: *Your Error = Your Integral – Given Integral = Your Integral – 4.028423

**Problem F.2** (7.5 points) Please write program(s) to carry out the following tasks:

(1) (0.5 point) To generate matrices $A^{n\times n}$ and $B^{n\times n}$ with elements $a_{ij}, b_{ij}\sim U(-1,1)$ where $n = 2^{10}$.
(2) (4.0 point) To compute $C = A \times B$ by Strassen algorithm of 2 levels.
(3) (3.0 point) To compute $C^{-1}$ by any algorithm.

**Note**: Direct calls of MATLAB or some AI tools get you no credits, although borrowing them to verify your results is encouraged.

**Problem F.3** (7.5 points) Please solve the following IVP numerically, and complete other relevant tasks, by programming any algorithm(s) of your choice.

(1) (2.5 points) Select $N = 10^4$ uniform mesh points for $t \in [0,5]$ to find the numerical solution $x(t)$ of

$$\begin{cases} \exp(-x') = x' + x^3 - 3\exp(-t^3) \\ x(t=0) = 1 \end{cases}$$

For this part, you do not need to tabulate $N$ pairs of $(t, x)$ values, but you must plot the solution.

(2) (2.5 points) Take 6 points from your solution above: $x(t=0), x(t=1), \ldots, x(t=5)$ and interpolate them by a polynomial $P_5(t)$. Plot $P_5(t)$ and report the 6 coefficients.

(3) (2.5 points) Fit the above 6 points by the t-exp fit to a function with fitting parameters $\alpha$ and $\beta$. Plot $x_{\text{Fit}}(t)$ and report the values of $\alpha$ and $\beta$.

$$x_{\text{Fit}}(t) = 1 + \alpha t * e^{-\beta t}$$

**Note**: I deliberately included "-1" to make fitting easier and $x(t) - 1$ may look like:

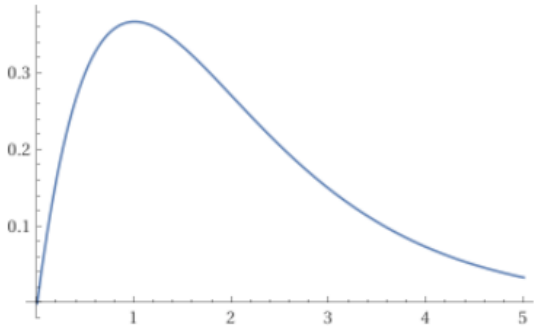


**Figure 2.** The likely shape of the solution for this problem.

**Note**: (1) Example 8 (p. 30) of Topic6A Lecture Notes might help. (2) The three curves required for this problem can be plotted in one figure, which is highly encouraged, or in three separate ones. Your call.

**Problem F.4** (7.5 points) Given a boundary value problem (BVP),

$$\begin{cases} y'' + xy = 0 \\ y(0) = 1 \\ y(2) = 2 \end{cases}$$

Please solve this BVP for reasonable accuracy by writing a program to apply any method and use any reasonable associated parameters. A table like the following is proposed to record your solution:

| $x$ | $y(x)$ |
|---|---|
| 0.00 | 1.000 |
| 0.01 | |
| 0.02 | |
| … | |
| … | |
| … | |
| 1.98 | |
| 1.99 | |
| 2.00 | 2.000 |

**Grading guide**: 2.0 points for correct description of the algorithm(s); 2.0 points for correct program(s); 3.5 points for getting 99 correct numbers (with ≥ 2 digits of accuracy) for the table.

**Note**: We lectured two methods for BVP: (1) shooting method and (2) finite-difference method. For the mesh size, $h = \Delta x = \frac{2}{200}$ is suggested.

To save your time looking up GPT, I plot a suggested solution. Although I know you rely on GPT for most of the solution, I hope you have sufficient details of the description/implementation of the algorithm and program.

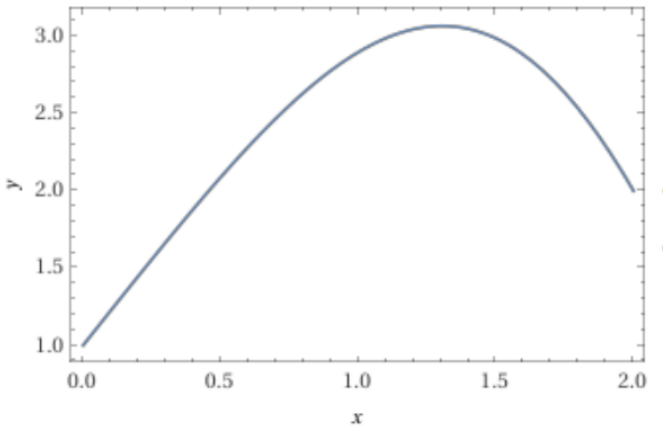


Figure 3. Suggested solution for the assigned BVP. 2.24086