1. (10 points) My older son is two years older than my younger son. My younger son is two years older than my cat. My cat’s age plus twice my dog’s age is equal to my older son’s age. The sum of all of their ages is 53. Using this information, set up a linear system satisfied by their ages. Then determine their ages using Gaussian elimination with partial pivoting, either by hand or with code that you have written. (Don’t just use the MATLAB “x=A\b” capability.)

2. (10 points) Consider the simple static system illustrated. The system consists of 13 rigid members connected at 8 joints, each of which allows free rotation. All horizontal and vertical members are of length one; all diagonal members are of length $\sqrt{2}$. The system is loaded by 1000 pound weights at joints 3, 5, and 7.

![Diagram of the system](image)

The (contractive) member forces are denoted by $f_1, \ldots, f_{13}$. They are such that the system is at rest, i.e., they are in equilibrium at the joints. Thus these forces are determined by the following system of linear equations ($\alpha = \sin \frac{\pi}{4} = \cos \frac{\pi}{4}$):

Joint 2: \[
\begin{align*}
-\alpha f_1 + f_4 + \alpha f_5 &= 0 \\
-\alpha f_1 - f_3 - \alpha f_5 &= 0
\end{align*}
\]

Joint 3: \[
\begin{align*}
-f_2 + f_6 &= 0 \\
f_3 - 1000 &= 0
\end{align*}
\]

Joint 4: \[
\begin{align*}
-f_4 + f_8 &= 0 \\
-f_7 &= 0
\end{align*}
\]

Joint 5: \[
\begin{align*}
-\alpha f_5 - f_6 + \alpha f_9 + f_{10} &= 0 \\
\alpha f_5 + f_7 + \alpha f_9 - 1000 &= 0
\end{align*}
\]

Joint 6: \[
\begin{align*}
-f_8 - \alpha f_9 + \alpha f_{12} &= 0 \\
-\alpha f_9 - f_{11} - \alpha f_{12} &= 0
\end{align*}
\]
Joint 7: \[
\begin{align*}
-f_{10} + f_{13} &= 0 \\
f_{11} - 1000 &= 0
\end{align*}
\]
Joint 8: \[
-\alpha f_{12} - f_{13} = 0
\]
Solve this system using Gaussian elimination with partial pivoting. You may use either a program you’ve written or the Matlab “\(x=A\backslash b\)” capability.

3. (10 points) Consider approximately solving the integral equation

\[
u(x) - \int_0^1 e^{-(x-t)^2} u(t) \, dt = x \cos(10x), \quad 0 \leq x \leq 1
\]

by discretizing the integral with the midpoint numerical integration rule, as follows:

For a given positive integer \(n\), define \(n\) grid or mesh points \(t_i\) in \([0,1]\) by setting \(h = 1/n\) and \(t_i = (i - 1/2)h\) for \(i = 1, \ldots, n\). Then obtain approximate solution values \(u_i \approx u(t_i)\) by solving the system of linear equations

\[
u_i - \sum_{j=1}^n he^{-(t_i-t_j)^2} u_j = t_i \cos(10t_i), \quad i = 1, \ldots, n.
\]

Solve this system with \(n = 64\). You may use any software you choose: your own code, software from a library, or MATLAB’s “\(x=A\backslash b\)” capability. If possible, plot the approximate solution values. (In MATLAB, this is done with \(\text{plot}(t,u)\), where \(t\) and \(u\) are the vectors of \(t_i\)’s and \(u_i\)’s, respectively.) If you can’t plot the solution for some reason, print out the \(t_i\)’s and \(u_i\)’s. Hand in your code and your plot or printout.

\textit{Note:} The linear system can be written in matrix-vector form as \((I - K)u = b\), where \(K_{ij} = he^{-(t_i-t_j)^2}\) and \(b_i = t_i \cos(10t_i)\).

\textit{Note:} In fact, you can safely use Gaussian elimination without pivoting on this problem. A matrix \(A \in \mathbb{R}^{n \times n}\) is symmetric positive-definite (SPD) if \(A = A^T\) and \(v^TAv > 0\) for all non-zero \(v \in \mathbb{R}^n\). It can be shown that Gaussian elimination without pivoting is stable on SPD matrices, and the matrix for this problem, \(I - K\), is SPD.