1. A dynamic system is described by equation:

$$\dot{x}(t) = Ax(t) + Bu(t)$$

Where *A* is the system dynamics matrix. To design a kalman filter, the state transition function needs to be determined as follows:

$$\hat{x}_{n+1,n} = F\hat{x}_{n,n} + G\hat{u}_{n,n}$$

Given *A* matrix as follows, derive the state transition matrix *F*.

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$$

Provide key steps to demonstrate how you derive the matrix F.

2. In numerical optimization, the line search method is described by the iteration:

$$x_{k+1} = x_k + \alpha_k p_k$$

Where α_k is the step length and p_k is the search direction depending on the method you choose. Your task is:

a) program the optimization algorithm with search direction determined by the two different methods: steepest descent and Newton algorithms

b) using the backtracking line search algorithm described in the following block to determine step length α_k of each iteration *k*:

Algorithm 3.1 (Backtracking Line Search).
Choose $\bar{\alpha} > 0, \rho \in (0, 1), c \in (0, 1)$; Set $\alpha \leftarrow \bar{\alpha}$;
repeat until $f(x_k + \alpha p_k) \leq f(x_k) + c\alpha \nabla f_k^T p_k$
$\alpha \leftarrow \rho\alpha$;
end (repeat)
Terminate with $\alpha_k = \alpha$.

c) Implement your optimization algorithm to minimize the Rosenbrock function:

$$f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2.$$

With initial step length $\alpha_0 = 1$, initial point $x_0 = (1.2,\ 1.2)^T$. Print out the minimization value you get from the two different search directions and step length at each iteration of the two methods.

Note: solve the problem with C or C++, both your implementation and code architecture will be taken into consideration when evaluating your submission.

3. How would you evaluate these two methods? What metrics would you use to evaluate these two methods?

4. Present your findings in your on-site interview. Expect to present to the team using Excel, Python, Google Sheets, anything that you think will help you get your point across on the pros & cons of each method/approach.

Grading Criteria:
Is the solution correct?
Code architecture. Looking for:

- Well defined optimization functions that will enable us to specify a target function and method

Concepts:

- Derive something new
- Find the bug
- Experiment design
- Presentation of results

Sensor behavior & Calibration questions