1. Compute the QR-decomposition of the matrix
\[ A = \begin{bmatrix} 3 & 7 \\ 0 & 12 \\ 4 & 1 \end{bmatrix}. \]

2. Consider the function \( f(x) = x^4 \). Find the Lagrange-polynomial that interpolates the function \( f \) at the points \( x_0 = -1, \ x_1 = 0 \) and \( x_2 = 2 \).

3. Consider the sum
\[ \sum_{k=0}^{n-1} \cos(j \ t_k) \sin(\hat{j} \ t_k), \]
where \( t_k = k \frac{2\pi}{n} \) are the grid points. Using the formulas
\[ \cos(j \ t_k) \sin(\hat{j} \ t_k) = \frac{1}{2} \text{Im} \left\{ e^{i(j+\hat{j})t_k} - e^{i(j-\hat{j})t_k} \right\}, \]
and
\[ \sin(j \ t_k) \sin(\hat{j} \ t_k) = \frac{1}{2} \text{Re} \left\{ e^{i(j+\hat{j})t_k} - e^{i(j-\hat{j})t_k} \right\}, \]
calculate the above sums for all the possible values of \( j, \hat{j} \in \{0, 1, ..., \frac{n-1}{2} \} \) following the same procedure as in the lecture notes.

4. Reminder: If \( p_i \) is a linear polynomial of the form
\[ p_i(x) = f(x_i) + \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i} (x - x_i), \quad i = 0, ..., n \]
then \( s(x) = p_i(x) \) for \( x \in [x_i, x_{i+1}] \), where \( s \in S_{1,\Delta} \) is the linear spline for the grid \( \Delta = \{a = x_0 < x_1 < \ldots < x_n = b\} \) of the interval \([a, b]\).

Let \( f(x) = x^2, \ x \in [0, 3] \). Approximate the function \( f \) at the nodal points \( x_i = i, \ i = 0, 1, 2, 3 \) with a linear spline \( s \in S_{1,\Delta} \).

5. Create a MATLAB-Function that implements the linear spline interpolation for a given function \( f : [a, b] \to \mathbb{R} \) and any grid \( \Delta \) on the interval \([a, b]\). The algorithm should have the following structure:

**INPUT:** \( \Delta \), i.e. the number of grid points.
Main body: Algorithm based on the polynomial of exercise 4.

Output: Graph of $f$ and $s \in S_{1,\Delta}$.

Test your algorithm with the function

$$f(x) = \frac{1}{1 + 25x^2}, \quad x \in [-1, 1],$$

and $n = 4, 8, 12$ equidistant grid points.

6. An efficient implementation in MATLAB to calculate the Fourier coefficients $a_k$

of the DFT

$$x_n = \sum_{k=0}^{N-1} a_k e^{ik \frac{2\pi}{N} n}, \quad n = 0, \ldots, N - 1$$

is the function `fft`. To illustrate some of the properties of the DFT, consider

different rectangular pulses of the form

$$x_n = \begin{cases} 1, & n \in \left[\frac{N-1}{2} - a, \frac{N-1}{2} + a\right] \\ 0, & \text{otherwise} \end{cases},$$

for a given positive integer $a$ and $N$ odd number. Additionally, consider the
pulses,

$$x_{bn} = \begin{cases} 1, & n \in \left[\frac{N-1}{2} - a/b, \frac{N-1}{2} + a/b\right] \\ 0, & \text{otherwise} \end{cases},$$

and

$$x_n^{(b)} = \begin{cases} x_n/b, & n \in \left[\frac{N-1}{2} - ab : b : \frac{N-1}{2} + ab\right] \\ 0, & \text{otherwise} \end{cases},$$

related to decimation and time expansion, respectively. Using `fft` for $N = 31$, $a = 2$ and $b = 1, 2, 3$ present the Fourier transforms (using the graphs of $a_k$) for the different values of $b$. 
