Photonics – series 1 (Fundamentals of programming in Matlab)

Problem 1: *(Complex numbers, complex modulus/argument/real and imaginary parts, arithmetic operations on complex numbers; functions: real, imag, log, log10, exp, conj, abs, angle)*
- Solve equation $az^2+bz+c=0$ for $a=1, b=-1-i, c=2\mathrm{i}$ (Hint – you can use explicit formula or (advanced!) *fzero*). Display real and imaginary parts, complex modules and arguments of solutions.
- Prove that $z=e^{2\pi i l/5}$ is the 5-th root of 1 for $l=(0,1,2,3,4)$.

Problem 2: *(Matrices, basic matrix operation – indices, addition, multiplication, inversion; operator ':' (range), functions inv, mean)*
- Solve vector equation $A\cdot x=b$ for $A=\begin{bmatrix} e^{\pi} & \sqrt{2} \\ \ln(3) & \log_{10}(4)+i \end{bmatrix}$, $b=\begin{bmatrix} \arcsin(0.5) \\ tgh(4) \end{bmatrix}$.
- Represent matrix A and vectors x,b in coordinate system, rotated by 45º (Hint – create rotation matrix).
- Display matrix A^2 ; matrix, elements of which are squares of elements of matrix A ; matrix, elements of which are modulus squares of elements of matrix A .
- Find maximal and minimal values of last matrix.
- Create vector from second row of matrix A , display mean value of vector and sum of modulus squares of vector elements.

Problem 3: *(Vectors, scalar and vector products, vector arithmetic operations, sum and mean of vector elements; functions: norm, sum, randn, dot, times)*
- Create two three-dimensional vectors a,b , elements of which are chosen randomly from normal (Gaussian) distribution with mean value 5 and standard distribution 1.
- Calculate cosine of angle between vectors a,b (Hint – use scalar product).
- Create unit vectors $a/|a|, b/|b|$ and unit vector perpendicular to both of them $(a\times b)/|a\times b|$.

Problem 4: *(Functions, input and output of functions, loop 'for')*
- Write function which uses loops to calculate $f(x,y)=\mathrm{Re}(\exp(i(k_x\cdot x+k_y\cdot y-\varphi_0)))$, where $k_x=\sin(\alpha)\cdot 2\pi/\lambda, k_y=\cos(\alpha)\cdot 2\pi/\lambda$. Function output is a square array of values; function input consists of incidence angle, wavelength, range of x and y values, size of output matrix and initial phase $\alpha,\lambda,x_{min},x_{max},y_{min},y_{max},N,\varphi_0$.

Problem 5: *(Logical operators, conditions 'if')*
- Use function from problem 4 to generate matrix of size $N=300$ for $\lambda=405\,nm, \alpha=30\,deg$ and (x,y) range of $5\mu m$.
- Find all negative matrix elements.
- Substitute all matrix elements outside of range (-0.25,0.25) with zeros.

Problem 6: *(Grid creation; functions: linspace, meshgrid)*
- Write function which calculates $f(x,y) = x^2 + y^2$ for given x and y. Check that function works if input arguments are arrays.
- Create x vector, containing 41 numbers in range [-10,10], and y vector with 81 numbers in range [-20,20].
- Use *meshgrid* according to Matlab example. Display generated matrices.
- Use function from first part of problem on matrices, created with meshgrid.
- Find all pairs of points (x,y) inside circle of radius 5.

<u>Problem 7:</u> *(One-dimensional plots; functions: plot, figure, title, xlabel, ylabel, axis, legend)*
Plot first-order Bessel function (*besselj*) $J_m(x)$ and $(x/2)^m/m!$ in range $x \in]0,2]$. Plot both functionsin one figure for chosen natural *m*. Use different colors and line types for both functions. Use markers to plot points of one chosen function. Add legend with functions description. Add grid to the plot. Add axis labels and plot title. (Hint – you may use functions of Matlab or GUI editor).

<u>Problem 8:</u> *(Two-dimensional plots; functions: imagesc, mesh, contour, colormap)*
Display values of function from problem 4 on two-dimensional plot. Assume that λ is a length unit and that (x,y) are in range of 6λ . Add axis labels and plot title. Change used color scheme.

<u>Problem 9:</u> *(Input/output operations; functions: save, load, clear, clc)*
Save chosen variable in your home folder. Clear Matlab workspace. Load previously saved variable. Save chosen plot both as Matlab figure ('.fig') and as image ('.jpg', '.png').