Submit homeworks via the Student Portal. All code must be executable by Matlab. Submit files of code and computer-generated output. Other working may be hand-written on A4 paper and scanned to a PDF file (photographs will not be graded),or written using on a computer and submitted both in source and PDF formats.

1. *(Due by 23:59 on Tuesday 19th April 2022)*

Write Matlab code to solve a differential equation using the 3rd-order Adams-Bashforth method,with a given fixed step size. Bootstrap your method using a Runga-Kutta method of appropriate order.

Apply your both your Runge-Kutta boostrapping and the Adams-Bashforth method to solve the initial-value problem

$$\dot{y} = \cos(t) + y - y^3; \quad y(0) = 2;$$

up to time $t = 6$ using a step-size of $h = 0.1$. Compute the (absolute) error at $t = 6$,given that the exact value is -0.4845092473 (10 dp). What happens to the error if you halve the step size?

Use the secant method to find the time at which $y(t) = 0$ to an accuracy of $\pm 10^{-3}$.
Hint: Find the step k such that $y(t)$ crosses 0 between t_k and t_{k+1}. Then find a step size h_k such that $y(t_k + h_k) = 0$.

Note: You should give the estimated values of y for times up to 0.5,and for the final time.

2. *(Due by 23:59 on Thursday 28th April 2022)*

Write Matlab function using divided differences to compute the coefficients a_i of the Newton nested form of a polynomial interpolation,and to evaluate a polynomial in Newton nested form given the coefficients a_i and values x_i.

Use your method to compute the polynomial interpolants for (x_i, y_i) and $(x_i, \tilde{y}_i)$ for the following data:

i	0	1	2	3	4	5	6	7	8	9	10
x_i	2.5	2.0	3.0	1.5	3.5	1.0	4.0	0.5	4.5	0.0	5.0
y_i	0.06	0.19	0.00	0.43	0.01	0.77	0.07	1.25	0.14	2.00	0.21
$\tilde{y}_i$	0.09	0.13	0.00	0.41	0.12	0.69	0.02	1.26	0.19	2.09	0.13

You should display the coefficients of the nested form resulting polynomials,and the table of divided differences for interpolant of (x_i, y_i),and plot the graphs of the data and the polynomial over the interval $[-0.1, 5.1]$. It is also recommended to check your answer by evaluating at the x_i.

The data for y is given by a smooth function,and for $\tilde{y}$ by adding random Gaussian noise with standard deviation 0.06. Explain your results,paying attention to the intervals $[0, 0.5]$ and $[4.0, 5.0]$. Comment on the suitability of polynomial interpolation in the presence of noise.

Use that fact that for $p(x) = a_0 + (x - x_0)q(x)$,we have $p'(x_0) = q(x_0)$ to estimate the derivative at $x_0 = 2.5$.
Hint: Take $q(x) = a_1 + (x - x_1)(a_2 + (x - x_2)(a_3 + \cdots))$

3. *(Due by 23:59 on Thursday 12th May 2022)*

Write a Matlab function to evaluate integrals using Simpson's rule or Romburg integration.

Write a Matlab function to compute the coefficients of the best approximation in the least-squares sense over the interval $[-\pi, +\pi]$ by a trigonometric polynomial (Fourier series) of degree k:

$$s_k(x) = \frac{a_0}{2} + \sum_{j=1}^{k} \big(a_j \cos(jx) + b_j \sin(jx)\big)$$

Also write a Matlab code to evaluate the resulting trigonometric polynomial.

Let

$$f(x) = \frac{\cos(2x)}{4 + 3\sin(x)}.$$

Compute the Fourier coefficients up to degree 6,and plot the s_k for $k = 1, 2, 3$. Use the error formula for least-squares approximation by orthogonal functions to give the errors of s_k for $k = 1, \ldots, 6$.

4. *(Due by 23:59 on Wednesday 25th May 2021)*

Write a Matlab function approximating the eigenvalues and eigenvectors of a symmetric matrix A using the inverse power method to a given accuracy. You may assume that the standard unit basis vectors $\mathbf{e}_i$ give suitable starting vectors for the iteration,with corresponding eigenvalue estimates $\mu_i = \mathbf{e}_i^T A \mathbf{e}_i = a_{i,i}$.

Apply your method to compute the eigenvalues and eigenvectors of the 4×4 matrix to an accuracy of 10^{-6}.

$$A = \begin{pmatrix} 6.35 & 0.71 & 0 & 0 \\ 0.71 & 3.25 & 0.71 & 0 \\ 0 & 0.71 & 0.47 & 1.21 \\ 0 & 0 & 1.21 & -1.88 \end{pmatrix}.$$