1. **Rozwiąż układ z macierzą Hilberta.**
   - Niech $n = 10$. Utwórz macierz Hilberta $A \in \mathbb{R}^{n \times n}$ oraz wektor prawej strony b (np. $b = \mathbf{1}$ lub wektor z elementami $b_i = \sum_j A_{ij}$).
   - Rozwiąż układ $Ax = b$ trzema sposobami: `numpy.linalg.solve`, `numpy.linalg.lstsq` oraz przez `inv(A).dot(b)`.
   - Porównaj otrzymane rozwiązania, policz błędy (np. norma różnicy względem `solve` lub norma residuum $|Ax - b|_2$) i oblicz `cond(A)` (np. `numpy.linalg.cond`).
2. **Dopasowanie wielomianów metodą najmniejszych kwadratów.**
   - Wygeneruj dane bez szumu: $y = \sin(2\pi x)$ na odcinku $[0, 1]$.
   - Wybierz 15 równoodległych węzłów w $[0, 1]$ i wygeneruj obserwacje z szumem (np. dodaj `np.random.normal(0, sigma, size)`; wybierz rozsądne `sigma`).
   - Dopasuj wielomiany stopnia 1, 2 i 3 metodą najmniejszych kwadratów (np. `np.polyfit` lub tworząc macierz Vandermonde i wywołując `np.linalg.lstsq`).
   - Policz błąd dopasowania (np. `RMSE` na węzłach) i wstaw wykres: oryginalna krzywa $y = sin(2\pi x)$, dane z szumem oraz dopasowane wielomiany.
3. **Wartości własne macierzy tridiagonalnej (symetrycznej).**
   - Zdefiniuj symetryczną tridiagonalną macierz (np. z elementami na diagonali (d_i) i poza diagonalą (a_i)).
   - Policzyć wartości własne dwiema metodami: `scipy.linalg.eig` (dla ogólnych macierzy) oraz `scipy.linalg.eigvalsh` (dla macierzy hermitowskich/symetrycznych).
   - Porównaj wyniki numeryczne (różnice) oraz zmierz czas wykonania obu metod (np. `time.perf_counter`).
4. **Znaleźć wszystkie rzeczywiste pierwiastki wielomianu.**
   - Dany wielomian: $p(x) = x^4 - 3x^3 + 2x - 1$.
   - Najpierw narysuj wykres funkcji na odpowiednim przedziale, aby zidentyfikować przedziały, gdzie funkcja zmienia znak (sugerowane przedziały np. $[-5, 5]$ lub zawężone po wstępnej obserwacji).
   - Na znalezionych przedziałach zastosuj metodę `scipy.optimize.brentq` do znalezienia pierwiastków rzeczywistych.