

10.4.1.2.2

EE24BTECH11006 - Arnav Mahishi

Question:

The product of two consecutive integers is 306. We need to find the integers.

Variable	Description
x	The bigger integer out of the two we need to find
$g(x)$	The function we take to update x_n in the point iteration method
$x_{\alpha,n}$	The value of x after n iterations for the first root
$x_{\beta,n}$	The value of x after n iterations for the second root

TABLE 0: Caption

TABLE 0: Variables Used

Theoretical Solution:

Lets start by assuming the bigger integer as x and the smaller integer as $\frac{306}{x}$

$$\Rightarrow x - \frac{306}{x} = 1 \quad (0.1)$$

$$\Rightarrow x^2 - 306 = x \quad (0.2)$$

$$\Rightarrow x^2 - x - 306 = 0 \quad (0.3)$$

Using the quadratic formula:

$$x = \frac{1 \pm \sqrt{1^2 - (4 \cdot -306)}}{2} \quad (0.4)$$

$$x_1 = \frac{1 + \sqrt{1225}}{2} = 18 \quad (0.5)$$

$$x_2 = \frac{1 - \sqrt{1225}}{2} = -17 \quad (0.6)$$

If $x = 18$ the other integer will be 17 if $x = -17$ the other integer will be -18

$$p(2) = 41 - 72(-2) - 18(-2)^2 = 113 \quad (0.7)$$

\therefore The integers can be 18, 17 or $-18, -17$

Computational Solution:

Below are three methods to find the solutions of this quadratic equation, QR decomposition on Hessenberg matrix with Single Shift:

It is a Numerical method for finding eigenvalues of a given matrix
The QR algorithm is as follows:

1) QR decomposition

$$A = QR \quad (1.1)$$

a) Q is an $m \times n$ orthogonal matrix

b) R is an $n \times n$ upper triangular matrix.

Given a matrix $A = [a_1, a_2, \dots, a_n]$, where each a_i is a column vector of size $m \times 1$.

2) Normalize the first column of A :

$$q_1 = \frac{a_1}{\|a_1\|} \quad (2.1)$$

3) For each subsequent column a_i , subtract the projections of the previously obtained orthonormal vectors from a_i :

$$a'_i = a_i - \sum_{k=1}^{i-1} \langle a_i, q_k \rangle q_k \quad (3.1)$$

Normalize the result to obtain the next column of Q :

$$q_i = \frac{a'_i}{\|a'_i\|} \quad (3.2)$$

Repeat this process for all columns of A .

4) Finding R :-

After constructing the ortho-normal columns q_1, q_2, \dots, q_n of Q , we can compute the elements of R by taking the dot product of the original columns of A with the columns of Q :

$$r_{ij} = \langle a_j, q_i \rangle, \text{ for } i \leq j \quad (4.1)$$

For the given polynomial equation the companion matrix will be

$$\Lambda = \begin{pmatrix} 0 & 1 \\ 306 & 1 \end{pmatrix} \quad (4.2)$$

This algorithm involves decomposing a given matrix A into two matrices, Q and R , where Q is an orthogonal matrix and R is an upper triangular matrix. After performing the decomposition, the matrix is updated as $A' = RQ$. This process is repeated iteratively. In theory, as the number of iterations approaches infinity, A' will converge to an upper triangular matrix with its diagonal elements representing the eigenvalues of A . To accelerate the convergence, a shift is introduced into the matrix during the iterations, as described below.

1) Initialization

Let $A_0 = A$, where A is the given matrix.

2) QR Decomposition

For each iteration $k = 0, 1, 2, \dots$:

a) Compute the QR decomposition of A_k , such that:

$$A_k = Q_k R_k \quad (2.1)$$

where:

- i) Q_k is an orthogonal matrix ($Q_k^\top Q_k = I$).
- ii) R_k is an upper triangular matrix.

The decomposition ensures $A_k = Q_k R_k$.

b) Form the next matrix A_{k+1} as:

$$A_{k+1} = R_k Q_k \quad (2.2)$$

3) Convergence

Repeat Step 2 until A_k converges to an upper triangular matrix T . The diagonal entries of T are the eigenvalues of A .

4) The eigenvalues of matrix will be the roots of the equation.

This is the basic QR algorithm but we can introduce a shift to make the convergence faster

$$\sigma = 1 \quad (4.1)$$

$$\Lambda_{\text{shifted}} = \begin{pmatrix} 0 & 1 \\ 306 & 1 \end{pmatrix} - \sigma I \quad (4.2)$$

$$\Lambda_{\text{shifted}} = \begin{pmatrix} -1 & 1 \\ 306 & 0 \end{pmatrix} \quad (4.3)$$

Where σ is the last diagonal element. Since Q zeroes out the lower triangular element, which in our case, there is only such element. We will construct an orthogonal matrix such that the element will become 0. Since it is orthogonal

$$Q = \begin{pmatrix} c & s \\ -s & c \end{pmatrix} \quad (4.4)$$

$$(4.5)$$

Where

$$c = \cos \phi \quad (4.6)$$

$$s = \sin \phi \quad (4.7)$$

Where ϕ is the rotation angle. Since it should zero out that one element

$$\begin{pmatrix} c & s \\ -s & c \end{pmatrix} \times \begin{pmatrix} -1 & 1 \\ 306 & 0 \end{pmatrix} = \begin{pmatrix} a & b \\ 0 & c \end{pmatrix} \quad (4.8)$$

$$c(1) + s(-306) = 0 \quad (4.9)$$

$$c^2 + s^2 = 1 \quad (4.10)$$

Solving for c and s gives

$$c = \frac{-1}{\sqrt{(1)^2 + (306)^2}} \quad (4.11)$$

$$s = \frac{306}{\sqrt{(1)^2 + (306)^2}} \quad (4.12)$$

Now, as we got the Q matrix we will do the following

$$\Lambda_{\text{new}} = Q\Lambda_{\text{shifted}}Q^T + \sigma I \quad (4.13)$$

$$\Lambda_{\text{new}} = \begin{pmatrix} c & s \\ -s & c \end{pmatrix} \begin{pmatrix} -1 & 1 \\ 306 & 0 \end{pmatrix} \begin{pmatrix} c & -s \\ s & c \end{pmatrix} + \sigma I \quad (4.14)$$

$$\Lambda_{\text{new}} \approx \begin{pmatrix} -17.4965 & -304.9422 \\ -0.0578 & 18.4965 \end{pmatrix} \quad (4.15)$$

As the matrix A is undergoing similarity transformation, the eigenvalues will not change. Run the same sequence of steps for 19 iterations after which you end up with the following matrix

$$\Lambda_{\text{new}} = \begin{pmatrix} -17 & -305 \\ 0 & 18 \end{pmatrix} \quad (4.16)$$

Since its an upper triangular matrix, the eigenvalues are same as its diagonal elements
Hence the roots of given equation are -17 and 18

We will now generalize how to find the eigenvalues of any given matrix. We say a matrix A is in hessenberg form if it is in form shown below

$$H = \begin{pmatrix} \times & \times & \times & \cdots & \times \\ \times & \times & \times & \cdots & \times \\ 0 & \times & \times & \cdots & \times \\ 0 & 0 & \times & \cdots & \times \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \times \end{pmatrix} \quad (4.17)$$

We will use householder method to reduce any matrix into hessenberg form.

It reduces an $n \times n$ matrix to hessenberg form by $n - 2$ orthogonal trasformations. Each transformations annihilates the required part of a whole column at a time rather than element wise elimination. The basic ingredient for a house holder matrix is P which is in the form

$$P = I - 2\mathbf{w}\mathbf{w}^T \quad (4.18)$$

where \mathbf{w} is a vector with $|\mathbf{w}|^2 = 1$. The matrix P is orthogonal as

$$P^2 = (I - 2\mathbf{w}\mathbf{w}^T) \cdot (I - 2\mathbf{w}\mathbf{w}^T) \quad (4.19)$$

$$= I - 4\mathbf{w}\mathbf{w}^T + 4\mathbf{w} \cdot (\mathbf{w}^T \mathbf{w}^T) \cdot \mathbf{w}^T \quad (4.20)$$

$$= I \quad (4.21)$$

Therefore, $P = P^{-1}$ but $P = P^\top$, so $P = P^\top$

We can rewrite P as

$$P = I - \frac{\mathbf{u}\mathbf{u}^\top}{H} \quad (4.22)$$

where the scalar H is

$$H = \frac{1}{2} |\mathbf{u}|^2 \quad (4.23)$$

Where \mathbf{u} can be any vector. Suppose \mathbf{x} is the vector composed of the first column of A . Take

$$\mathbf{u} = \mathbf{x} \mp |\mathbf{x}| \mathbf{e}_1 \quad (4.24)$$

Where $\mathbf{e}_1 = (1 \ 0 \ \dots)^\top$, we will take the choice of sign later. Then

$$P \cdot \mathbf{x} = \mathbf{x} - \frac{\mathbf{u}}{H} \cdot (\mathbf{u} \mp |\mathbf{x}| \mathbf{e}_1)^\top \cdot \mathbf{x} \quad (4.25)$$

$$= \mathbf{x} - \frac{2\mathbf{u}(|\mathbf{x}|^2 \mp |\mathbf{x}| x_1)}{2|\mathbf{x}|^2 \mp |\mathbf{x}| x_1} \quad (4.26)$$

$$= \mathbf{x} - \mathbf{u} \quad (4.27)$$

$$= \mp |\mathbf{x}| \mathbf{e}_1 \quad (4.28)$$

To reduce a matrix A into Hessenberg form, we choose vector \mathbf{x} for the first householder matrix to be lower $n - 1$ elements of the first column, then the lower $n - 2$ elements will be zeroed.

$$P_1 \cdot A = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & p_{11} & p_{12} & \cdots & p_{1n} \\ 0 & p_{21} & p_{22} & \cdots & p_{2n} \\ 0 & p_{31} & p_{32} & \cdots & p_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & p_{n1} & p_{n2} & \cdots & p_{nn} \end{pmatrix} \begin{pmatrix} a_{00} & \times & \times & \cdots & \times \\ a_{10} & \times & \times & \cdots & \times \\ a_{20} & \times & \times & \cdots & \times \\ a_{30} & \times & \times & \cdots & \times \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n0} & \times & \times & \cdots & \times \end{pmatrix} \quad (4.29)$$

$$= \begin{pmatrix} a'_{00} & \times & \times & \cdots & \times \\ 0 & \times & \times & \cdots & \times \\ 0 & \times & \times & \cdots & \times \\ 0 & \times & \times & \cdots & \times \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \times & \times & \cdots & \times \end{pmatrix} \quad (4.30)$$

Now if we multiply the matrix $P_1 A$ with P_1 , the eigenvalues will be conserved as it is a similarity transformation.

Now we choose the vector \mathbf{x} for the householder matrix to be the bottom $n - 2$ elements

of the second column, and from it construct the P_2

$$P_2 = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & p_{22} & \cdots & p_{2n} \\ 0 & 0 & p_{32} & \cdots & p_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & p_{n2} & \cdots & p_{nn} \end{pmatrix} \quad (4.31)$$

By performing a similarity transformation PAP , we can zero out the $n-3$ elements in the second column. Continuing this process systematically results in the Hessenberg form of the matrix A . Since our matrix was 2×2 , it was already in Hessenberg form.

Next, we decompose the matrix in Hessenberg form into two matrices, Q and R , where Q is an orthogonal matrix and R is an upper triangular matrix. After this decomposition, the matrix is updated as $A' = RQ$. This process is repeated iteratively.

Theoretically, as the number of iterations approaches infinity, A' will converge to an upper triangular matrix whose diagonal elements correspond to the eigenvalues of A . However, a minor issue arises when the matrix entries are real but the eigenvalues are complex. This issue will be addressed shortly. Importantly, the eigenvalues of A remain unchanged because of the following reasoning:

$$A = QR \quad (4.32)$$

$$R = Q^\top A \quad (4.33)$$

$$A' = RQ \quad (4.34)$$

$$A' = Q^\top A Q \quad (4.35)$$

As the matrix A is undergoing similarity transformation, the eigenvalues will not change. The rate of convergence of A depends on the ratio of absolute values of the eigenvalues. That is, if the eigenvalues are $|\lambda_1| \geq |\lambda_2| \geq |\lambda_3| \cdots \geq |\lambda_n|$ then, the elements of A_k below the diagonal to converge to zero like

$$|a_{ij}^{(k)}| = O\left(\left|\frac{\lambda_i}{\lambda_j}\right|^k\right) \quad i > j \quad (4.36)$$

We define a rotation matrix G , to zero out the elements which are non-diagonal, since the matrix which we are dealing is a Hessenberg matrix, we need to zero out the elements which are just below the diagonal elements.

$$G = \begin{pmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & c & s & \cdots & 0 \\ 0 & \cdots & -s & c & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 1 \end{pmatrix} \quad (4.37)$$

Where the value of c and s are

$$c = \frac{\overline{x_{i,i}}}{\sqrt{|x_{i,i}|^2 + |x_{i,i+1}|^2}} \quad (4.38)$$

$$s = \frac{\overline{x_{i,i+1}}}{\sqrt{|x_{i,i}|^2 + |x_{i,i+1}|^2}} \quad (4.39)$$

If we multiply G and A , we can see easily that it nulls out the element in $(i+1)^{\text{th}}$ row and i^{th} column. The following matrix multiplication eliminates all the elements below the diagonal of A

$$A \implies G_{n-1}G_{n-2} \cdots G_2G_1A \quad (4.40)$$

Now, we store $G_{n-1}G_{n-2} \cdots G_2G_1$ in Q and then

$$A' \implies QAQ^T \quad (4.41)$$

$$(4.42)$$

If we carry out these transformation infinite times, the A will be an upper triangular matrix with diagonal elements as eigenvalues. If all the entries in the matrix are real but the eigenvalues are complex, the matrix A will converge to a Quasi-triangular form, that is

$$A = \begin{pmatrix} B_1 & 0 & \cdots & 0 \\ 0 & B_2 & \cdots & 0 \\ \vdots & \cdots & \ddots & 0 \\ 0 & 0 & 0 & B_n \end{pmatrix} \quad (4.43)$$

Where B_i is a 2×2 block matrix. These matrices are called Jordan blocks. In this case, the eigenvalues are determined by solving the characteristic equation of the 2×2 matrix. Since this equation is quadratic, it can be solved easily, and the solutions of the characteristic equation will be the eigenvalues.

The major drawback of the QR decomposition algorithm is that the rate of convergence can sometimes be very slow. The Rayleigh Quotient method addresses this issue by increasing the rate of convergence through the use of a shift. According to the order of convergence provided in equation (0.35), if the last element satisfies $(\lambda_i = 0)$, the order of convergence becomes significantly higher.

To implement this, we shift the Hessenberg matrix by a certain amount, apply the QR decomposition algorithm, and then add the shift back. If the chosen shift exactly matches an eigenvalue, the algorithm completes in very few iterations (in the best case, just one iteration). Since the eigenvalue is not known, we use the last diagonal element as an initial guess for the shift.

$$H' = H - \sigma I \quad (4.44)$$

$$H' \implies H'_{transformed} \quad (4.45)$$

$$H_{next} = H'_{transformed} + \sigma I \quad (4.46)$$

This method does not change the eigenvalues as

$$\overline{H} = Q(H - \lambda I) Q^\top \quad (4.47)$$

$$= QHQ^\top - \lambda QIQ^\top \quad (4.48)$$

$$= QHQ^\top - \lambda I \quad (4.49)$$

$$\overline{H} + \lambda I = QHQ^\top \quad (4.50)$$

which is a similarity transformation.

Here, once we finding the eigenvalue and it is in the last diagonal element, we will leave it as it is and then focus on smaller matrix block present diagonally above the eigenvalue and then use the same technique to push the next eigenvalue to the next diagonal element. We will continue to do this till all the eigenvalues are present in the diagonal. This is know as deflation.

$$H - \lambda I = QR \quad (4.51)$$

$$R = \begin{pmatrix} \times & \times & \cdots & \times \\ 0 & \times & \cdots & \times \\ \vdots & \vdots & \ddots & \times \\ 0 & 0 & \cdots & 0 \end{pmatrix} \quad (4.52)$$

RQ Will also be in the same form

$$\overline{H} = RQ + \lambda I = \begin{pmatrix} \overline{H}_1 & \mathbf{h}_1 \\ 0^\top & \lambda \end{pmatrix} \quad (4.53)$$

In our case:

$$\Lambda = \begin{pmatrix} 0 & 1 \\ 306 & 1 \end{pmatrix} \quad (4.54)$$

$$\Rightarrow B = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 306 & 1 \end{pmatrix} \quad (4.55)$$

$$\Rightarrow d = \frac{a_{11} - a_{22}}{2} = -0.5 \quad (4.56)$$

$$\Rightarrow \text{shift} = a_{22} - \text{sgn}(d) \sqrt{d^2 + a_{12}a_{21}} = 18.5 \quad (4.57)$$

$$\hat{A}_0 = A_0 - \text{shift} \cdot I = \begin{pmatrix} -18.5 & 1 \\ 306 & -17.5 \end{pmatrix} \quad (4.58)$$

$$\Rightarrow Q_0 = \begin{pmatrix} -0.0603 & 0.9982 \\ 0.9982 & 0.0603 \end{pmatrix} \quad (4.59)$$

$$R_0 = \begin{pmatrix} 306.5586 & -17.5285 \\ 0 & -0.0579 \end{pmatrix} \quad (4.60)$$

$$\Rightarrow A_1 = R_0 Q_0 + \text{shift} \cdot I = \begin{pmatrix} -17.4965 & -304.9422 \\ -0.0578 & 18.4965 \end{pmatrix} \quad (4.61)$$

After 19 iterations the matrix converges to

$$A_{19} = \begin{pmatrix} -17 & -305 \\ 4.97 \times 10^{-11} & 18 \end{pmatrix} \quad (4.62)$$

The sub-diagonal elements are nearly zero confirming convergence and the diagonal elements are the eigenvalues. The eigen values given by the code are

$$x_1 = 18.000000000433293, x_2 = -17.000000000043303 \quad (4.63)$$

Theoretically we can solve it using:

$$|\Lambda - \lambda I| = 0 \quad (4.64)$$

$$\Rightarrow \left| \begin{pmatrix} 0 & 1 \\ 306 & 1 \end{pmatrix} - \lambda \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right| = 0 \quad (4.65)$$

$$\Rightarrow \left| \begin{pmatrix} -\lambda & 1 \\ 306 & 1 - \lambda \end{pmatrix} \right| = 0 \quad (4.66)$$

$$\Rightarrow \lambda^2 - \lambda - 306 = 0 \quad (4.67)$$

$$\lambda = 18, -17 \quad (4.68)$$

\therefore The roots of the equation are 18 and -17

Fixed Point Iterations:

Take an initial guess x_0 . The update difference equation will use the following function:

$$x = g(x) \quad (4.69)$$

For our problem,

$$g(x) = \sqrt{x^2 - 306} \quad (4.70)$$

To get both roots we do two iterations, now the update equations will be

$$x_{\alpha,n+1} = g(x_{\alpha,n}) \quad (4.71)$$

$$x_{\beta,n+1} = g(x_{\beta,n}) \quad (4.72)$$

$$(4.73)$$

We take two initial guesses $x_{\alpha,0}$ and $x_{\beta,0}$ close to each root. Then we continue calculating the each $x_{\alpha,n}$ and $x_{\beta,n}$ until

$$|x_{n+1} - x_n| < \epsilon \quad (4.74)$$

Where ϵ is the tolerance which we have taken as $1e-6$. In each of the α series and β series we get a root.

This is proved by a theorem as follows:

Let $x = s$ be a solution of $x = g(x)$ and suppose that g has a continuous derivative in some interval J containing s . Then if $|g'| \leq K < 1$ in J , the iteration process defined above converges for any x_0 in J . The limit of the sequence $[x_n]$ is s .

This can also be solved by the Newton-Raphson Method,

Start with an initial guess x_0 , and then run the following logical loop. The update equation is as follows

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad (4.75)$$

where ,

$$f(x) = x^2 - x - 306 \quad (4.76)$$

$$f'(x) = 2x - 1 \quad (4.77)$$

The problem with this method is if the roots are complex but the coefficients are real, x_n either converges to an extrema or grows continuously without any bound. To get the complex solutions, however , we can just take the initial guess point to be a random complex number.

The output of a program written to find roots is shown below:

Fixed-Point Iteration for Positive Root:

Iteration 1: x = 17.577532, f(x) = -14.607907

Iteration 2: x = 17.988261, f(x) = -0.410729

Iteration 3: x = 17.999674, f(x) = -0.011413

Iteration 4: x = 17.999996, f(x) = -0.000237

Iteration 5: x = 18.000000, f(x) = -0.000009

Converged to solution: x = 18.000000

Fixed-Point Iteration for Negative Root:

Iteration 1: x = -9.767519, f(x) = -200.828057

Iteration 2: $x = -17.211406$, $f(x) = 7.443887$
 Iteration 3: $x = -16.993781$, $f(x) = 0.217625$
 Iteration 4: $x = -17.001183$, $f(x) = -0.006402$
 Iteration 5: $x = -16.999995$, $f(x) = 0.000018$
 Iteration 6: $x = -17.000000$, $f(x) = -0.000000$
 Converged to solution: $x = -17.000000$

Newton-Raphson Iteration for Positive Root:

Iteration 1: $x = 19.448724$, $f(x) = 52.804159$
 Iteration 2: $x = 18.055381$, $f(x) = 1.941406$
 Iteration 3: $x = 18.000887$, $f(x) = 0.003057$
 Iteration 4: $x = 18.000001$, $f(x) = 0.000007$
 Iteration 5: $x = 18.000000$, $f(x) = 0.000000$
 Converged to solution: $x = 18.000000$

Newton-Raphson Iteration for Negative Root:

Iteration 1: $x = -19.809749$, $f(x) = 106.235914$
 Iteration 2: $x = -17.194357$, $f(x) = 6.840276$
 Iteration 3: $x = -17.007687$, $f(x) = 0.073361$
 Iteration 4: $x = -17.000017$, $f(x) = 0.000307$
 Iteration 5: $x = -17.000000$, $f(x) = 0.000000$
 Converged to solution: $x = -17.000000$

Eigenvalues (Roots of the equation):

Eigenvalue 1: $-17.000000 + 0.0000000i$
 Eigenvalue 2: $18.000000 + 0.0000000i$