Topics: Review, iteration using for

Reading (ML): Sec 4.2–4.5 (exclude 4.2.2)

Example 1

Write a program segment to evaluate an \( n \)th order polynomial of \( x \):

\[
a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n
\]

A given vector \( p \) has length \( n + 1 \) and contains the coefficients of the polynomial where \( p(1) \) is the coefficient for the term \( x^0 \). Your program should request a user input value for variable \( x \) and then evaluate the polynomial. Use a while loop. Do not use MATLAB predefined functions other than length and input.

Example 2

Write a program segment that determines whether a given integer \( n \) is prime. Assume \( n > 2 \). (Hint: MATLAB function \( \text{mod}(x,y) \) returns the value of the remainder of \( x \) divided by \( y \) assuming integer values of \( x, y \).)

Example 3

Sketch a program that will list all the prime numbers in the range of \([m,n] \) given integers \( m,n > 1 \) and \( m < n \).

Example 4

Develop an algorithm for calculating the mode of a sequence. The mode is the number in the sequence that occurs with maximum frequency. Assume that the sequence is (a) non-negative, (b) entered one by one and terminated by a negative number, and (c) entered in non-decreasing order. E.g., the mode of the sequence 87,92,92,98,98,98,100 is 98. Assume that only scalar variables are allowed.

Programming Rules of Thumb

- Learn program patterns of general utility and use relevant pattern for the problem at hand.
- Seek inspiration by systematically working test data by hand. Be introspective; ask yourself: “what am I doing?”
- Declare variables for each piece of information you maintain when working problem by hand. Write comments that precisely describe the contents of each variable.
- Decompose problem into manageable tasks.
- Remember the problem’s boundary conditions.
- Validate your program by tracing it on simple test data.

Two patterns for doing something \( n \) times

<table>
<thead>
<tr>
<th>for ( i = 1 : n )</th>
<th>while ( i &lt;= n )</th>
</tr>
</thead>
<tbody>
<tr>
<td>% do something</td>
<td>% do something</td>
</tr>
<tr>
<td>% ...</td>
<td>% ...</td>
</tr>
<tr>
<td>end</td>
<td>( i = i + 1; )</td>
</tr>
<tr>
<td></td>
<td>end</td>
</tr>
</tbody>
</table>