

# Function Related Problems

(Total 27 questions)

| SL  | Problem statement   | Difficulty levels |                             |               |     |                             |     |                             |     |                 |
|-----|---|-------------------|-----------------------------|---------------|-----|-----------------------------|-----|-----------------------------|-----|-----------------|
| 1.  | Function to print a custom message.   | *                 |                             |               |     |                             |     |                             |     |                 |
|     | <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td></td><td>This is a function</td></tr></table>  |                   | Sample input                | Sample output |     | This is a function          |     |                             |     |                 |
|     | Sample input  |                   | Sample output               |               |     |                             |     |                             |     |                 |
|     | This is a function  |                   |                             |               |     |                             |     |                             |     |                 |
|     |   |                   |                             |               |     |                             |     |                             |     |                 |
| 2.  | Function to print an input character value.   | *                 |                             |               |     |                             |     |                             |     |                 |
|     | <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>3</td><td>Value received from main: 3</td></tr><tr><td>A</td><td>Value received from main: A</td></tr></table>                               |                   | Sample input                | Sample output | 3   | Value received from main: 3 | A   | Value received from main: A |     |                 |
|     | Sample input  |                   | Sample output               |               |     |                             |     |                             |     |                 |
|     | 3   |                   | Value received from main: 3 |               |     |                             |     |                             |     |                 |
| A   | Value received from main: A   |                   |                             |               |     |                             |     |                             |     |                 |
|     |   |                   |                             |               |     |                             |     |                             |     |                 |
|     |   |                   |                             |               |     |                             |     |                             |     |                 |
| 3.  | Function to determine if a number is even or odd.   | *                 |                             |               |     |                             |     |                             |     |                 |
|     | <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>3</td><td>odd</td></tr><tr><td>8</td><td>even</td></tr></table>  |                   | Sample input                | Sample output | 3   | odd                         | 8   | even                        |     |                 |
|     | Sample input  |                   | Sample output               |               |     |                             |     |                             |     |                 |
|     | 3   |                   | odd                         |               |     |                             |     |                             |     |                 |
| 8   | even  |                   |                             |               |     |                             |     |                             |     |                 |
|     |   |                   |                             |               |     |                             |     |                             |     |                 |
|     |   |                   |                             |               |     |                             |     |                             |     |                 |
| 4.  | Function to determine if a number is positive, negative or zero.  | *                 |                             |               |     |                             |     |                             |     |                 |
|     | <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>3</td><td>positive</td></tr><tr><td>-5</td><td>negative</td></tr><tr><td>0</td><td>zero</td></tr></table>                                    |                   | Sample input                | Sample output | 3   | positive                    | -5  | negative                    | 0   | zero            |
|     | Sample input  |                   | Sample output               |               |     |                             |     |                             |     |                 |
|     | 3   |                   | positive                    |               |     |                             |     |                             |     |                 |
|     | -5  |                   | negative                    |               |     |                             |     |                             |     |                 |
| 0   | zero  |                   |                             |               |     |                             |     |                             |     |                 |
|     |   |                   |                             |               |     |                             |     |                             |     |                 |
|     |   |                   |                             |               |     |                             |     |                             |     |                 |
|     |   |                   |                             |               |     |                             |     |                             |     |                 |
| 5.  | Function that takes two numbers as input and determines if the first number is greater than, equal to or less than the second number.   | *                 |                             |               |     |                             |     |                             |     |                 |
|     | <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>5 4</td><td>5 is greater than 4</td></tr><tr><td>2 6</td><td>2 is less than 6</td></tr><tr><td>8 8</td><td>8 is equal to 8</td></tr></table> |                   | Sample input                | Sample output | 5 4 | 5 is greater than 4         | 2 6 | 2 is less than 6            | 8 8 | 8 is equal to 8 |
|     | Sample input  |                   | Sample output               |               |     |                             |     |                             |     |                 |
|     | 5 4   |                   | 5 is greater than 4         |               |     |                             |     |                             |     |                 |
|     | 2 6   |                   | 2 is less than 6            |               |     |                             |     |                             |     |                 |
| 8 8 | 8 is equal to 8   |                   |                             |               |     |                             |     |                             |     |                 |
|     |   |                   |                             |               |     |                             |     |                             |     |                 |
|     |   |                   |                             |               |     |                             |     |                             |     |                 |
|     |   |                   |                             |               |     |                             |     |                             |     |                 |

| 6.                     | <p>Function to calculate the sum of <b>n</b> numbers coming from the console.</p> <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>80 33 27</td><td>Sum In Function: 140<br/>Sum In Main: 140</td></tr><tr><td>100 -100</td><td>Sum In Function: 0<br/>Sum In Main: 0</td></tr></table>                                    | Sample input | Sample output | 80 33 27      | Sum In Function: 140<br>Sum In Main: 140 | 100 -100               | Sum In Function: 0<br>Sum In Main: 0 | * |
|------------------------|--|--------------|---------------|---------------|--|------------------------|--------------------------------------|---|
| Sample input           | Sample output  |              |               |               |  |                        |                                      |   |
| 80 33 27               | Sum In Function: 140<br>Sum In Main: 140   |              |               |               |  |                        |                                      |   |
| 100 -100               | Sum In Function: 0<br>Sum In Main: 0   |              |               |               |  |                        |                                      |   |
| 7.                     | <p>Function to calculate the sum of <b>n</b> numbers coming from the console and stored in an array.</p> <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>3<br/>80 33 27</td><td>Sum In Function: 140<br/>Sum In Main: 140</td></tr><tr><td>2<br/>100 -100</td><td>Sum In Function: 0<br/>Sum In Main: 0</td></tr></table> | Sample input | Sample output | 3<br>80 33 27 | Sum In Function: 140<br>Sum In Main: 140 | 2<br>100 -100          | Sum In Function: 0<br>Sum In Main: 0 | * |
| Sample input           | Sample output  |              |               |               |  |                        |                                      |   |
| 3<br>80 33 27          | Sum In Function: 140<br>Sum In Main: 140   |              |               |               |  |                        |                                      |   |
| 2<br>100 -100          | Sum In Function: 0<br>Sum In Main: 0   |              |               |               |  |                        |                                      |   |
| 8.                     | <p>Function that takes an array of n integer numbers as input and prints them in reverse order.</p> <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>3<br/>4 8 2</td><td>2 8 4</td></tr><tr><td>7<br/>5 12 34 8 43 21 9</td><td>9 21 43 8 34 12 5</td></tr></table>  | Sample input | Sample output | 3<br>4 8 2    | 2 8 4                                    | 7<br>5 12 34 8 43 21 9 | 9 21 43 8 34 12 5                    | * |
| Sample input           | Sample output  |              |               |               |  |                        |                                      |   |
| 3<br>4 8 2             | 2 8 4  |              |               |               |  |                        |                                      |   |
| 7<br>5 12 34 8 43 21 9 | 9 21 43 8 34 12 5  |              |               |               |  |                        |                                      |   |
| 9.                     | <p>Function to calculate the factorial of a number.</p> <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>3</td><td>6</td></tr><tr><td>5</td><td>120</td></tr></table>  | Sample input | Sample output | 3             | 6  | 5                      | 120                                  | * |
| Sample input           | Sample output  |              |               |               |  |                        |                                      |   |
| 3                      | 6  |              |               |               |  |                        |                                      |   |
| 5                      | 120  |              |               |               |  |                        |                                      |   |
| 10.                    | <p>Function to take two positive numbers x and y as input and calculate x to the power y.</p> <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>3 4</td><td>3 to the power 4 is 81</td></tr><tr><td>10 3</td><td>10 to the power 3 is 1000</td></tr></table>  | Sample input | Sample output | 3 4           | 3 to the power 4 is 81                   | 10 3                   | 10 to the power 3 is 1000            | * |
| Sample input           | Sample output  |              |               |               |  |                        |                                      |   |
| 3 4                    | 3 to the power 4 is 81   |              |               |               |  |                        |                                      |   |
| 10 3                   | 10 to the power 3 is 1000  |              |               |               |  |                        |                                      |   |

| 11.   | Function to take a string as input and find its length.                 | *            |               |                     |  |                   |                  |  |
|---|---|--------------|---------------|---------------------|--|-------------------|------------------|--|
| <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>hello world</td><td>11</td></tr><tr><td>I love my country</td><td>17</td></tr></table>                                     |   | Sample input | Sample output | hello world         | 11   | I love my country | 17               |  |
| Sample input  | Sample output   |              |               |                     |  |                   |                  |  |
| hello world   | 11  |              |               |                     |  |                   |                  |  |
| I love my country   | 17  |              |               |                     |  |                   |                  |  |
| 12.   | Function to swap two numbers.<br>(Restriction: Pass by value)           | *            |               |                     |  |                   |                  |  |
| <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>10 20</td><td>Value in func: 20 10<br/>Value in main: 10 20</td></tr></table>  |   | Sample input | Sample output | 10 20               | Value in func: 20 10<br>Value in main: 10 20 |                   |                  |  |
| Sample input  | Sample output   |              |               |                     |  |                   |                  |  |
| 10 20   | Value in func: 20 10<br>Value in main: 10 20                            |              |               |                     |  |                   |                  |  |
| 13.   | Function to swap two numbers.<br>(Restriction: Pass by reference)       | **           |               |                     |  |                   |                  |  |
| <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>10 20</td><td>Value in func: 20 10<br/>Value in main: 20 10</td></tr></table>  |   | Sample input | Sample output | 10 20               | Value in func: 20 10<br>Value in main: 20 10 |                   |                  |  |
| Sample input  | Sample output   |              |               |                     |  |                   |                  |  |
| 10 20   | Value in func: 20 10<br>Value in main: 20 10                            |              |               |                     |  |                   |                  |  |
| 14.   | Function to determine only even numbers in an array of input integers.  | *            |               |                     |  |                   |                  |  |
| <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>24 77 117 -512 1024</td><td>24 -512 1024</td></tr><tr><td>45 33 0 256</td><td>0 256</td></tr></table>                      |   | Sample input | Sample output | 24 77 117 -512 1024 | 24 -512 1024                                 | 45 33 0 256       | 0 256            |  |
| Sample input  | Sample output   |              |               |                     |  |                   |                  |  |
| 24 77 117 -512 1024   | 24 -512 1024  |              |               |                     |  |                   |                  |  |
| 45 33 0 256   | 0 256   |              |               |                     |  |                   |                  |  |
| 15.   | Function that finds and returns the minimum value in an array.          | **           |               |                     |  |                   |                  |  |
| <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>157 -28 -37 26 10</td><td>Minimum Value: -37</td></tr><tr><td>12 45 1 10 5 3 22</td><td>Minimum Value: 1</td></tr></table> |   | Sample input | Sample output | 157 -28 -37 26 10   | Minimum Value: -37                           | 12 45 1 10 5 3 22 | Minimum Value: 1 |  |
| Sample input  | Sample output   |              |               |                     |  |                   |                  |  |
| 157 -28 -37 26 10   | Minimum Value: -37  |              |               |                     |  |                   |                  |  |
| 12 45 1 10 5 3 22   | Minimum Value: 1  |              |               |                     |  |                   |                  |  |
| 16.   | Function that multiplies the array elements by 2 and returns the array. | *            |               |                     |  |                   |                  |  |

|                   | <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>157 -28 -37 26 10</td><td>314 -56 -74 52 20</td></tr><tr><td>12 45 1 10 5 3 22</td><td>24 90 2 20 10 6 44</td></tr></table>   | Sample input | Sample output | 157 -28 -37 26 10 | 314 -56 -74 52 20       | 12 45 1 10 5 3 22 | 24 90 2 20 10 6 44             |    |  |     |           |     |       |    |
|-------------------|--|--------------|---------------|-------------------|-------------------------|-------------------|--------------------------------|----|--|-----|-----------|-----|-------|----|
| Sample input      | Sample output  |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 157 -28 -37 26 10 | 314 -56 -74 52 20  |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 12 45 1 10 5 3 22 | 24 90 2 20 10 6 44   |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 17.               | <p>Function to sort and return an input array in ascending order.</p> <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>10 22 -5 117 0</td><td>-5 0 10 22 117</td></tr></table>   | Sample input | Sample output | 10 22 -5 117 0    | -5 0 10 22 117          | **                |                                |    |  |     |           |     |       |    |
| Sample input      | Sample output  |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 10 22 -5 117 0    | -5 0 10 22 117   |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 18.               | <p>Function “<b>IsPrime()</b>” to determine whether a number is prime or not.</p> <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>1</td><td>Not prime</td></tr><tr><td>2</td><td>Prime</td></tr><tr><td>11</td><td>Prime</td></tr><tr><td>39</td><td>Not prime</td></tr><tr><td>101</td><td>Prime</td></tr></table>   | Sample input | Sample output | 1                 | Not prime               | 2                 | Prime                          | 11 | Prime  | 39  | Not prime | 101 | Prime | ** |
| Sample input      | Sample output  |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 1                 | Not prime  |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 2                 | Prime  |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 11                | Prime  |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 39                | Not prime  |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 101               | Prime  |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 19.               | <p>Function “<b>GeneratePrime()</b>” to compute the prime numbers less than <b>N</b>, where <b>N</b> is an input integer. <b>GeneratePrime()</b> uses <b>IsPrime()</b> to check whether a number is prime or not.</p> <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>5</td><td>Prime less than 5: 2, 3</td></tr><tr><td>10</td><td>Prime less than 10: 2, 3, 5, 7</td></tr><tr><td>40</td><td>Prime less than 17: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37</td></tr></table> | Sample input | Sample output | 5                 | Prime less than 5: 2, 3 | 10                | Prime less than 10: 2, 3, 5, 7 | 40 | Prime less than 17: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37 | *** |           |     |       |    |
| Sample input      | Sample output  |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 5                 | Prime less than 5: 2, 3  |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 10                | Prime less than 10: 2, 3, 5, 7   |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 40                | Prime less than 17: 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37   |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 20.               | <p>Function “<b>GenNthPrime()</b>” to compute the <b>N<sup>th</sup></b> prime number, where <b>N</b> is an integer input.</p> <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>5</td><td>5th Prime: 11</td></tr><tr><td>10</td><td>10th Prime: 29</td></tr><tr><td>40</td><td>40th Prime: 173</td></tr></table>  | Sample input | Sample output | 5                 | 5th Prime: 11           | 10                | 10th Prime: 29                 | 40 | 40th Prime: 173  | *** |           |     |       |    |
| Sample input      | Sample output  |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 5                 | 5th Prime: 11  |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 10                | 10th Prime: 29   |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |
| 40                | 40th Prime: 173  |              |               |                   |                         |                   |                                |    |  |     |           |     |       |    |

| 21.                 | <p>Implement the following functions and calculate standard deviation of an array whose values come from the terminal-</p> <div><div>TakeInput()</div><div>CalcMean(array, num_of_elem)</div><div>Calc_Std_deviation(array, num_of_elem)</div></div> <div><div>Formula:</div><div><math display="block">\sigma = \sqrt{\frac{\sum (x - M)^2}{N}}</math></div></div> <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>4 5 5 4 4 2 2 6</td><td>1.32</td></tr><tr><td>600 470 170 430 300</td><td>147.32</td></tr></table>  | Sample input        | Sample output | 4 5 5 4 4 2 2 6 | 1.32 | 600 470 170 430 300 | 147.32 | ***            |   |     |
|---------------------|--|---------------------|---------------|-----------------|------|---------------------|--------|----------------|---|-----|
| Sample input        | Sample output  |                     |               |                 |      |                     |        |                |   |     |
| 4 5 5 4 4 2 2 6     | 1.32   |                     |               |                 |      |                     |        |                |   |     |
| 600 470 170 430 300 | 147.32   |                     |               |                 |      |                     |        |                |   |     |
| 22.                 | <p>Function <b>find_substr( )</b> that takes two string arrays (<b>a, b</b>) as parameters, returns 1 if string <b>b</b> is found anywhere in string <b>a</b>, or returns –1 if no match is found.</p> <p>(Assuming, strlen(a)&gt;strlen(b))</p> <table><tr><th>Sample input (a, b)</th><th>Sample output</th></tr><tr><td>madam     adam</td><td>1</td></tr><tr><td>telescope   less</td><td>0</td></tr><tr><td>101010     101</td><td>1</td></tr></table>  | Sample input (a, b) | Sample output | madam     adam  | 1    | telescope   less    | 0      | 101010     101 | 1 | **  |
| Sample input (a, b) | Sample output  |                     |               |                 |      |                     |        |                |   |     |
| madam     adam      | 1  |                     |               |                 |      |                     |        |                |   |     |
| telescope   less    | 0  |                     |               |                 |      |                     |        |                |   |     |
| 101010     101      | 1  |                     |               |                 |      |                     |        |                |   |     |
| 23.                 | <p>Function <b>find_substr( )</b> that takes two string arrays (<b>a, b</b>) as parameters, uses function <b>str_length()</b> to determine the lengths of the strings, and then looks for the smaller string anywhere in the bigger string. It returns 1 if the substring is found, or returns –1 if no match is found.</p> <p>[Restriction: str_length() cannot uses built-in strlen() function]</p> <table><tr><th>Sample input (a, b)</th><th>Sample output</th></tr><tr><td>madam     adam</td><td>1</td></tr><tr><td>telescope   less</td><td>0</td></tr><tr><td>101010     101</td><td>1</td></tr></table> | Sample input (a, b) | Sample output | madam     adam  | 1    | telescope   less    | 0      | 101010     101 | 1 | *** |
| Sample input (a, b) | Sample output  |                     |               |                 |      |                     |        |                |   |     |
| madam     adam      | 1  |                     |               |                 |      |                     |        |                |   |     |
| telescope   less    | 0  |                     |               |                 |      |                     |        |                |   |     |
| 101010     101      | 1  |                     |               |                 |      |                     |        |                |   |     |

| 24.   | <p>Program that continuously takes two positive integers as inputs and uses two functions to find their GCD (greatest common divisor) and LCM (least common multiple). Both functions take parameters and returns desired values.</p> <p>[Hint: Use infinite loop to process inputs]</p> <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>5 7</td><td>GCD: 1<br/>LCM: 35</td></tr><tr><td>12 12</td><td>GCD: 12<br/>LCM: 12</td></tr><tr><td>12 32</td><td>GCD: 4<br/>LCM: 96</td></tr></table>  | Sample input | Sample output | 5 7  | GCD: 1<br>LCM: 35  | 12 12   | GCD: 12<br>LCM: 12  | 12 32 | GCD: 4<br>LCM: 96 | ** |
|---|--|--------------|---------------|--|--|---|---|-------|-------------------|----|
| Sample input  | Sample output  |              |               |  |  |   |   |       |                   |    |
| 5 7   | GCD: 1<br>LCM: 35  |              |               |  |  |   |   |       |                   |    |
| 12 12   | GCD: 12<br>LCM: 12   |              |               |  |  |   |   |       |                   |    |
| 12 32   | GCD: 4<br>LCM: 96  |              |               |  |  |   |   |       |                   |    |
| 25.   | <p>Program that implements function to perform operations on a 3X5 matrix:</p> <p style="text-align: center;"><i>InputMatrix()</i><br/><i>ShowMatrix()</i><br/><i>ScalarMultiply()</i></p> <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>7 16 55 13 12<br/>12 10 52 0 7<br/>-2 1 2 4 9<br/><br/>2</td><td>Original:<br/>7 16 55 13 12<br/>12 10 52 0 7<br/>-2 1 2 4 9<br/><br/>Multiplied by 2:<br/>14 32 110 26 24<br/>24 20 104 0 14<br/>-4 2 4 8 18</td></tr><tr><td>7 16 55 13 12<br/>12 10 52 0 7<br/>-2 1 2 4 9<br/><br/>-1</td><td>Original:<br/>7 16 55 13 12<br/>12 10 52 0 7<br/>-2 1 2 4 9<br/><br/>Multiplied by -1:<br/>-14 -32 -110 -26 -24<br/>-24 -20 -104 0 -14<br/>4 -2 -4 -8 -18</td></tr></table> | Sample input | Sample output | 7 16 55 13 12<br>12 10 52 0 7<br>-2 1 2 4 9<br><br>2 | Original:<br>7 16 55 13 12<br>12 10 52 0 7<br>-2 1 2 4 9<br><br>Multiplied by 2:<br>14 32 110 26 24<br>24 20 104 0 14<br>-4 2 4 8 18 | 7 16 55 13 12<br>12 10 52 0 7<br>-2 1 2 4 9<br><br>-1 | Original:<br>7 16 55 13 12<br>12 10 52 0 7<br>-2 1 2 4 9<br><br>Multiplied by -1:<br>-14 -32 -110 -26 -24<br>-24 -20 -104 0 -14<br>4 -2 -4 -8 -18 | ***   |                   |    |
| Sample input  | Sample output  |              |               |  |  |   |   |       |                   |    |
| 7 16 55 13 12<br>12 10 52 0 7<br>-2 1 2 4 9<br><br>2  | Original:<br>7 16 55 13 12<br>12 10 52 0 7<br>-2 1 2 4 9<br><br>Multiplied by 2:<br>14 32 110 26 24<br>24 20 104 0 14<br>-4 2 4 8 18   |              |               |  |  |   |   |       |                   |    |
| 7 16 55 13 12<br>12 10 52 0 7<br>-2 1 2 4 9<br><br>-1 | Original:<br>7 16 55 13 12<br>12 10 52 0 7<br>-2 1 2 4 9<br><br>Multiplied by -1:<br>-14 -32 -110 -26 -24<br>-24 -20 -104 0 -14<br>4 -2 -4 -8 -18  |              |               |  |  |   |   |       |                   |    |

| 26.  | <div>Program that implements function to perform operations on a <b>MXN</b> matrix:</div> <div><i>InputMatrix()</i><br/><i>ShowMatrix()</i><br/><i>ScalarMultiply()</i></div> <table><tr><th>Sample input</th><th>Sample output</th></tr><tr><td>2 2<br/><br/>7 16<br/>12 10<br/><br/>2</td><td>Original:<br/>7 16<br/>12 10<br/><br/>Multiplied by 2:<br/>14 32<br/>24 20</td></tr><tr><td>3 5<br/><br/>7 16 55 13 12<br/>12 10 52 0 7<br/>-2 1 2 4 9<br/><br/>-1</td><td>Original:<br/>7 16 55 13 12<br/>12 10 52 0 7<br/>-2 1 2 4 9<br/><br/>Multiplied by -1:<br/>-14 -32 -110 -26 -24<br/>-24 -20 -104 0 -14<br/>4 -2 -4 -8 -18</td></tr></table> | Sample input      | Sample output | 2 2<br><br>7 16<br>12 10<br><br>2 | Original:<br>7 16<br>12 10<br><br>Multiplied by 2:<br>14 32<br>24 20 | 3 5<br><br>7 16 55 13 12<br>12 10 52 0 7<br>-2 1 2 4 9<br><br>-1 | Original:<br>7 16 55 13 12<br>12 10 52 0 7<br>-2 1 2 4 9<br><br>Multiplied by -1:<br>-14 -32 -110 -26 -24<br>-24 -20 -104 0 -14<br>4 -2 -4 -8 -18 | **** |
|--|--|-------------------|---------------|-----------------------------------|--|--|---|------|
| Sample input   | Sample output  |                   |               |                                   |  |  |   |      |
| 2 2<br><br>7 16<br>12 10<br><br>2                                | Original:<br>7 16<br>12 10<br><br>Multiplied by 2:<br>14 32<br>24 20   |                   |               |                                   |  |  |   |      |
| 3 5<br><br>7 16 55 13 12<br>12 10 52 0 7<br>-2 1 2 4 9<br><br>-1 | Original:<br>7 16 55 13 12<br>12 10 52 0 7<br>-2 1 2 4 9<br><br>Multiplied by -1:<br>-14 -32 -110 -26 -24<br>-24 -20 -104 0 -14<br>4 -2 -4 -8 -18  |                   |               |                                   |  |  |   |      |
| 27.  | <div>Program to convert a positive integer to another base using the following functions-</div> <div>I. Get_Number_And_Base () : Takes number to be converted (N) and base value (B) from user. Base must be between 2 and 16.</div> <div>II. Convert_Number () : Does the conversion</div> <div>III. Show_Converted_Number() : Displays the converted value.</div> <table><tr><th>Sample input(N,B)</th><th>Sample output</th></tr><tr><td>100 8</td><td>144</td></tr></table>  | Sample input(N,B) | Sample output | 100 8                             | 144  | ****   |   |      |
| Sample input(N,B)  | Sample output  |                   |               |                                   |  |  |   |      |
| 100 8  | 144  |                   |               |                                   |  |  |   |      |

|  |        |                               |  |  |
|--|--------|-------------------------------|--|--|
|  | 512 16 | 200                           |  |  |
|  | 512 0  | Base not within proper range! |  |  |
|  |        |                               |  |  |