

Chapter 7 Homework – Single Dimensional Arrays

Reading Assignment:

Read Chapter 7 in Introduction to Programming with C++, 3rd Edition by Liang

Problem Assignment:

1. (12 points) Work *Checkpoint Exercise 7.7* (parts a,b,c,d,e, g only) on p. 273. Include the instructions.
2. (42 points – 12 points per part) Write a C++ main program that calls each function listed below. Load values into each array using a list to test each function. Print out results to show original array contents and the result of the contents of the array after using the functions. All printing should be done in the main programs – **do not print from the functions**. Turn in copies of the program and all results.
 - A) Include a function to find the sum of the elements of an array.
Form of function call: Sum = ArraySum(A, Size)
 - B) Include a function to swap the contents of two equal size arrays named A and B.
Form of function call: ArraySwap(A, B, Size)
 - C) Include a function to reverse the contents of an array. For example, an array that originally contained the elements 1,2,3,4 would contain 4,3,2,1 after the function call.
Form of function call: ArrayReverse(A, Size)

Example output: The output should clearly show original array contents, new array contents (if modified), and the result. For example, the output for parts B, C, and A might look as follows:

```

Contents of array A before swap:  1  2  3  4  5  6  7
Contents of array B before swap:  8  9 10 11 12 13 14
Contents of array A after swap:   8  9 10 11 12 13 14
Contents of array B after swap:   1  2  3  4  5  6  7
Contents of array C before reverse: 3  4  5  6  7  8  9
Contents of array C after reverse:  9  8  7  6  5  4  3
Sum of array C: 42

```

Test your program for the example above (use the original contents for arrays A, B, and C above).

3. (15 pts) Work *Programming Exercise 7.4* (Analyze scores) on p. 299 in the text. Hint: Use one loop to read the scores, then find the average, and then use another loop to count the number of scores equal to or above the average. Test the program for the following cases:
 - 100, 99, 98, 88, 77, 66, 55
 - Enter 20 grades or your choice (with some in the 90s, 80s, 70, and 60s)
4. (15 pts) Write a C++ program that defines three 1D arrays with 21 elements in each array. The arrays will be used to store temperatures in degrees F, degrees C, and Kelvin. Use a loop to initialize the array with degrees C values to 0, 5, 10, 15, ..., 100. The program should then calculate the corresponding array values for degrees F and Kelvin. Display the results neatly in a table with three columns (one for degrees C, one for degrees F, and one for Kelvin) and include table headings with units. Use 0 digits after the decimal point for degrees C and use 1 digit after the decimal point for degrees F and Kelvin.

5. (13 points) Determine the output for each part below by hand (do not compile the programs).

<pre>// Array Homework: Problem 5A #include <iostream> using namespace std; int main(void) { int const ArraySize=10; int A[ArraySize] = {2, 4, 6, 8, 10, 12, 14, 16}; double Avg, Sum = 0; for (int j = 0; j < ArraySize; j++) Sum += A[j]; Avg = Sum/ArraySize; cout << "Avg = " << Avg << endl; return 0; }</pre>	<p>Avg = _____</p>
<pre>// Array Homework: Problem 5B #include <iostream> using namespace std; int main(void) { int const ArraySize=10; int Sum1 = 0, Sum2 = 0, A[ArraySize]; for (int j = 0; j < ArraySize; j++) A[j] = 2*j+1; for (int j = 0; j < ArraySize; j += 2) Sum1 += A[j]; for (int j = 1; j < ArraySize; j += 2) Sum2 += A[j]; cout << "Sum1 = " << Sum1 << endl; cout << "Sum2 = " << Sum2 << endl; return 0; }</pre>	<p>Sum1 = _____</p> <p>Sum2 = _____</p>
<pre>// Array Homework: Problem 5C #include <iostream> using namespace std; int main(void) { int const ArraySize=10; int Sum1 = 0, Sum2 = 0, A[ArraySize]; for (int j = 0; j < ArraySize; j++) A[j] = j*j; for (int j = 0; j < ArraySize/2; j++) Sum1 += A[j]; for (int j = ArraySize-1; j >= ArraySize/2; j--) Sum2 += A[j]; cout << "Sum1 = " << Sum1 << endl; cout << "Sum2 = " << Sum2 << endl; return 0; }</pre>	<p>Sum1 = _____</p> <p>Sum2 = _____</p>

<pre>// Array Homework: Problem 5D #include <iostream> using namespace std; int main(void) { int const ArraySize=10; int Sum = 0, A[ArraySize] = {2,4,6,8,10,12,14,16,18,20}; int B[ArraySize] = {11,10,9,8,7,6,5,4,3,2},C[ArraySize]; for (int j = 0; j < ArraySize; j++) C[j] = A[j]-B[j]; for (int k = 0; k < ArraySize; k++) Sum += C[k]; cout << "Sum = " << Sum << endl; return 0; }</pre>	Sum = _____
<pre>// Array Homework: Problem 5E #include <iostream> using namespace std; int main(void) { int const ArraySize=10; int Sum = 0, A[ArraySize], B[ArraySize], C[ArraySize]; for (int j = 0; j < ArraySize; j++) { A[j] = ArraySize/2 - j; B[j] = j%2; } for (int k = 0; k < ArraySize; k++) { if (A[k] > B[k]) C[k]=A[k]; else C[k] = B[k]; Sum += C[k]; } cout << "Sum = " << Sum << endl; return 0; }</pre>	Sum = _____
<pre>// Array Homework: Problem 5F #include <iostream> using namespace std; int main(void) { int const ArraySize=10; int Sum = 0, A[ArraySize], B[ArraySize], C[ArraySize]; for (int j = 0; j < ArraySize; j++) { A[j] = 3*j; B[j] = 3*ArraySize -2*j; } for (int k = 0; k < ArraySize; k++) { if (A[k]>B[k]) C[k]=1; else C[k] = 2; Sum += C[k]; } cout << "Sum = " << Sum << endl; return 0; }</pre>	Sum = _____