Kayley Seow

Chapter 6

Pages 292 – 295

R6.1, 2, 6, 10, 11, 13, 14, 23, 25

Chapter 6 Written Homework

R6.1 Write code that fills an array double values[10] with each set of values below.

a. 1 2 3 4 5 6 7 8 9 10
double values[10] ={1, 2, 3, 4, 5, 6, 7, 8, 9, 10};

b. 0 2 4 6 8 10 12 14 16 18
double values[10] ={0, 2, 4, 6, 8, 10, 12, 14, 16, 18};

c. 1 4 9 16 25 36 49 64 81 100
double values[10] ={1, 4, 9, 16, 25, 36, 49, 64, 81, 100};

d. 0 0 0 0 0 0 0 0 0 0
double values[10] ={0, 0, 0, 0, 0, 0, 0, 0, 0, 0};

e. 1 4 9 16 9 7 4 9 11
double values[10] ={1, 4, 9, 16, 9, 7, 4, 9, 11};

f. 0 1 0 1 0 1 0 1 0 1
double values[10] ={0, 1, 0, 1, 0, 1, 0, 1, 0, 1};

g. 0 1 2 3 4 0 1 2 3 4
double values[10] ={0, 1, 2, 3, 4, 0, 1, 2, 3, 4};

R6.2 Consider the following array: int a[] = { 1, 2, 3, 4, 5, 4, 3, 2, 1, 0 }; What is the value of total after the following loops complete?

a. int total = 0; for (int i = 0; i < 10; i++) { total = total + a[i]; }
total = 25

b. int total = 0; for (int i = 0; i < 10; i = i + 2) { total = total + a[i]; }
total = 13

c. int total = 0; for (int i = 1; i < 10; i = i + 2) { total = total + a[i]; }
total = 12

d. int total = 0; for (int i = 2; i <= 10; i++) { total = total + a[i]; }
total = 22

e. int total = 0; for (int i = 0; i < 10; i = 2 * i) { total = total + a[i]; }
Loop breaks, forever stuck at i = 0

f. int total = 0; for (int i = 9; i >= 0; i--) { total = total + a[i]; }
total = 25

g. int total = 0; for (int i = 9; i >= 0; i = i - 2) { total = total + a[i]; }
total = 12

h. int total = 0; for (int i = 0; i < 10; i++) { total = a[i] - total; }
total = -1

R6.6 What is wrong with the following loop?

```
int values[10];
for (int i = 1; i <= 10; i++)
```

```
{
	values[i] = i * i;
}
```

Explain two ways of fixing the error.

The for loop should start at i = 0, and the array should end at 9.

R6.10 Trace the flow of the element separator loop in Section 6.2.5 with the given example. Show two columns, one with the value of i and one with the output.

i	**Output**
0	1
1	|4
2	|9
3	|16
4	|25

R6.11 Trace the flow of the finding matches loop in Section 6.7.4, where values contains the elements 110 90 100 120 80. Show two columns, for i and matches.**321**

i	**matches**
0	110
1	
2	
3	120
4	

R6.13 Trace both mechanisms for removing an element described in Section 6.2.7. Use an array values with elements 110 90 100 120 80 and remove the element at index 2.

a. Element to be removed with the last element
pos = 2
current_size = 5
values[2] = values[size-1] = values[4] = 80
current_size = current_size - 1 = 4
values[]: 110 90 80 120
b. Move all elements following element to lower index, then decrement variable
pos = 2
current_size = 5
i = pos + 1 = 3
values[2] = values[3] = 120
        i++, i=4
values[3] = values[4] =80.
        i++, i=5
        exits out of the loop
current_size--, current_size=4.
values[]: 110 90 120 80

R6.14 For the operations on partially filled arrays below, provide the header of a function.

a. Sort the elements in decreasing order.
int sort_decreasing(int array[], int size)

b. Print all elements, separated by a given string.
Void print_element(int array[], int size, string given)

c. Count how many elements are less than a given value.
int count_less_elements(int array[], int size, int valuecounter)

d. Remove all elements that are less than a given value.
int remove_less_value(int array[], int size)

e. Place all elements that are less than a given value in another array.
int new_array_less_value(int array[], int size, int value)

R6.23 Write pseudocode for an algorithm that fills the first and last column as well as the first and last row of a two-dimensional array of integers with –1.

```
int negOneFill(int array[x][y])
{
    For all of the rows
        Set everything in first row to -1
        Set everything in last row to -1
    For all of the columns
        Starting from second row, set first and last to -1
}
```

R6.25 How do you perform the following tasks with vectors in C++?

a. Test that two vectors contain the same elements in the same order.
We can use == to compare.

b. Copy one vector to another.
Create a for loop, loop through the vector to be copied from, and use push_back() to copy from the other vector.

c. Fill a vector with zeroes, overwriting all elements in it.
Use a function that breaks up the vector so you can use pop back and push back and override in order to write 0’s in the vector.

d. Remove all elements from a vector
You can use clear() from according to shady docs from google.