1. Write a C++ function called `negSum` that returns the sum of all negative elements in an array of integers.
   For example, a program that uses the function `negSum` follows.
   ```cpp
   int data[6] = {-5, -4, 1, 3, 2, -3};
   int x;
   x = negSum(data, 6);  // Output is -12 because -5 + -4 + -3 = -12
   ```

2. Write a C++ function called `range` that returns the difference between the largest and smallest elements in an array.
   For example, a program that uses the function `range` follows.
   ```cpp
   int data[6] = {11, 12, 11, 3, 12, 13};
   int x;
   x = range(data, 6);  // Output is 10 because 13 - 3 = 10.
   ```

3. Write a function called `sum2D` that returns the sum of all elements in a 2-dimensional array.
   For example, a program that uses the function `sum2D` follows.
   ```cpp
   int array[3][4] = {{1,2,3,4},{1,2,3,4},{1,2,3,4}};
   cout << sum2D(array, 3, 4) << endl;
   return 0;
   ```
   The input values 3 and 4 specify the number of rows and columns in the array. The program should print an answer of 30.

4. Write a function called `biggestEntry` that uses a two dimensional array and two parameters representing the row and column capacities. The function should return the value of the biggest entry in the array.
   For example, a program that uses the function `biggestEntry` follows.
   ```cpp
   int x[2][3] = {{1,2,3},{4,7,3}};
   cout << biggestEntry(x, 2, 3) << endl;
   return 0;
   ```
   It should print 7 (since 7 is the biggest entry in the array).
Lab exercise (Arrays with functions)

5. Write a function called sixCount that returns a count of the number of entries that are equal to 6 in a 2-dimensional array. The function should use a parameter to specify the array and parameters for the row count and column count.

For example, a program that uses the function sixCount follows.

```cpp
int main() {
    int arr[2][6] = {{6,4,3,1,2,2}, {6,6,5,2,3,6}}; // array has 4 entries of 6
    cout << sixCount(arr, 2, 6) << endl; // prints 4
    return 0;
}
```