

Vectors Worksheet

1. What header file must you `#include` in order to define vector objects?

```
#include <vector>
```

2. Write a definition statement for a vector named `frogs`. `frogs` should be an empty vector of `ints`.

```
vector<int> frogs;
```

3. Write a definition statement for a vector named `lizards`. `lizards` should be a vector of 20 `floats`.

```
vector<int> lizards(20);
```

4. Write a definition statement for a vector named `toads`. `toads` should be a vector of 100 `chars`, with each element initialized to `'Z'`.

```
vector<char> toads(100, 'Z');
```

5. `gators` is an empty vector of `ints`. Write a statement that stores the value 27 in `gators`.

```
vector<int> gators;  
gators.push_back(27);
```

6. `snakes` is a vector of `doubles`, with 10 elements. Write a statement that stores the value 12.897 in element 4 of the `snakes` vector.

```
snakes[3] = 12.897;
```

7. To store a value in a vector that does not have a starting size, or that is already full, use the `push_back` member function.
8. Use the `pop_back` member function to remove the last element from a vector.
9. To completely clear the contents of a vector, use the `clear` member function.
10. Define a vector of `integers` that contains the first five prime numbers (2, 3, 5, 7, and 11). Use `push_back` to add the elements.

```
vector<int> primes;  
primes.push_back(2);  
primes.push_back(3);  
primes.push_back(5);  
primes.push_back(7);  
primes.push_back(11);
```

11. Answer the previous exercise without using `push_back`

```
vector<int> primes(5);
primes[0] = 2;
primes[1] = 3;
primes[2] = 5;
primes[3] = 7;
primes[4] = 11;
```

12. What is the contents of the vector `names` after the following statements?

```
vector<string> names;
names.push_back("Ann");
names.push_back("Bob");
names.pop_back();
names.push_back("Cal");
```

```
names = ["Ann", "Cal"]
```

13. Suppose you want to store a set of temperature measurements that is taken every five minutes. Should you use a vector or an array?

A vector

14. Suppose you want to store the names of the weekdays. Should you use a vector or an array of seven strings?

An array

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

a. Test that two vectors contain the same elements in the same order.

```
vec1 == vec2
```

b. Copy one vector to another.

```
for (int i = 0; i < vec1.size(); ++i)
    vec2.push_back(vec1[i]);
```

c. Fill a vector with zeroes, overwriting all elements in it.

```
for (int i = 0; i < vec1.size(); ++i)
    vec1[i] = 0;
```

d. Remove all elements from a vector.

```
vec1.clear()
```

16. Write a function

`vector<int> append(vector<int> a, vector<int> b)`

that appends one vector after another. For example, if a is

1 4 9 16

and b is

9 7 4 9 11

then append returns the vector

1 4 9 16 9 7 4 9 11

```
vector<int> append(vector<int> a, vector<int> b)
```

```
{
```

```
    vector<int> out(a);
```

```
    for (int i = 0; i < b.size(); ++i)
```

```
        out.push_back(b[i]);
```

```
    return out;
```

```
}
```

17. Write a function

```
vector<int> merge(vector<int> a, vector<int> b)
```

that merges two vectors, alternating elements from both vectors. If one vector is

shorter than the other, then alternate as long as you can and then append the remaining elements from the longer vector. For example, if a is

1 4 9 16

and b is

9 7 4 9 11

then merge returns the vector

1 9 4 7 9 4 16 9 11

```
vector<int> merge(vector<int> a, vector<int> b)
```

```
{  
    vector<int> out;  
    int k = 0;  
    int i = 0, j = 0;  
    while (i < a.size() && j < b.size()) {  
        if (k % 2 == 0) {  
            out.push_back(a[i++]);  
        } else {  
            out.push_back(b[j++]);  
        }  
        k++;  
    }  
  
    while (i < a.size()) {  
        out.push_back(a[i++]);  
    }  
    while (j < b.size()) {  
        out.push_back(b[j++]);  
    }  
    return out;  
}
```

18. Write a predicate function `bool same_elements(vector<int> a, vector<int> b)` that checks whether two vectors have the same elements in some order, with the same multiplicities. For example, 1 4 9 16 9 7 4 9 11 and 11 1 4 9 16 9 7 4 9 would be considered identical, but 1 4 9 16 9 7 4 9 11 and 11 11 7 9 16 4 1 would not. You will probably need one or more helper functions.

```
bool same_elements(vector<int> a, vector<int> b)
{
    if (a.size() != b.size()) {
        return false;
    }

    for (int i = 0; i < a.size(); ++i) {
        int a_count = 0;
        int b_count = 0;

        for (int j = 0; j < a.size(); ++j) {
            if (a[j] == a[i]) {
                ++a_count;
            }

            if (b[j] == a[i]) {
                ++b_count;
            }
        }

        if (a_count != b_count) {
            return false;
        }
    }

    return true;
}
```

19. What is the difference between the size and capacity of a vector?

The size of a vector is number of actual elements in the vector while the capacity is the maximum number of elements that can be in the vector without having to allocate new memory.