C++ Vectors

As we have seen, C++ arrays have many shortcomings. The vector class was developed to address these shortcomings.

Creating and Initializing Vectors

Default Constructor

```
vector<int> vec1; // cap = 0, size = 0
```

Specify Initial Capacity

```
vector<int> vec2(6); // cap = 6, size = 6, all zero values
```

Specify Initial Capacity and Initial Values

```
vector<int> vec3(6, 2); // cap = 6, size = 6, all values are 2
```

Specify Initial Values

```
vector<int> vec4{2, 3, 5, 7, 11, 13, 17}; // cap = 7, size = 7
```

Copy Constructor

```
vector<int> vec5(vec4); // cap = 7, size = 7
```

Assignment Operator

```
vector<int> vec6;
vec6 = vec5; // cap = 7, size = 7
```

Size and Capacity

Resize: Specify New Size

```
vector<int> vec1 {1, 2, 3}; // cap = 3, size = 3
vec1.resize(7); // cap = 7, size = 7, new values = 0
```

Resize: Specify New Size and Values

```
vector<int> vec2 {4, 5, 6}; // cap = 3, size = 3
vec2.resize(7, -1); // cap = 7, size = 7, new values = -1
```

Ensuring Capacity

```
vector<int> vec3 {7, 8, 9}; // cap = 3, size = 3
vec3.reserve(8); // cap = 8, size = 3
```

Changing Vector Contents

Initial Vector

```
vector<int> vec1{1, 2, 3}; // [1, 2, 3] cap = 3, size = 3
```

Change Element with []

```
vec1[0] = 10; // [10, 2, 3] cap = 3, size = 3
```

Change Element with "at" Function

```
vec1.at(1) = 20; // [10, 20, 3] cap = 3, size = 3
```

Change Element with "front" Function

```
vec1.front() = 100; // [100, 20, 3] cap = 3, size = 3
vec1.front()++; // [101, 20, 3] cap = 3, size = 3
```

Change Element with "back" Function

```
vec1.back() = 30; // [101, 20, 30] cap = 3, size = 3
vec1.back() += 7; // [101, 20, 37] cap = 3, size = 3
```

Clearing All Elements

```
vec1.clear(); // [] cap = 3, size = 0
```

Adding to the End of a Vector with push_back(int)

```
vector<int> vec1;  // []         cap = 0, size = 0
vec1.push_back(7); // [7]        cap = 1, size = 1
vec1.push_back(8); // [7, 8]     cap = 2, size = 2
vec1.push_back(9); // [7, 8, 9]  cap = 4, size = 3
```

Removing from the End of a Vector with pop_back()

Doesn't return a value. Cannot call on vector with size == 0.

```
vec1.pop_back(); // [7, 8]  cap = 4, size = 2
vec1.pop_back(); // [7]     cap = 4, size = 1
vec1.pop_back(); // []      cap = 4, size = 0
```

Swapping Two Entire Vectors

Swaps contents, capacity, and size

```
vector<int> vec1{1, 2, 3};    // [1, 2, 3]    cap = 3, size = 3
vector<int> vec2{5, 6, 7, 8}; // [5, 6, 7, 8] cap = 4, size = 4

vec1.swap(vec2);
// now vec1 = [5, 6, 7, 8] cap = 4, size = 4
//     vec2 = [1, 2, 3]    cap = 3, size = 3
```

Relational Operators

```
vector<int> vec1{2, 4, 6, 6, 6};
vector<int> vec2{2, 4, 6, 8, 10};

vec1 <  vec2  // true
vec1 <= vec2  // true
vec1 == vec2  // false
vec1 >= vec2  // false
vec1 >  vec2  // false
vec1 != vec2  // true
```