Adding Data to a Vector

May 2, 2020

0.1 Vector push_back

Now that you are able to process a string, you may want to store the results of the processing in a convenient container for later use. In the next exercise, you will store the streamed `ints` from each line of the board in a `vector<int>`. To do this, you will add the `ints` to the back of the vector, using the `vector` method `push_back`:

In [ ]:
```
#include <vector>
#include <iostream>
using std::vector;
using std::cout;

int main() {
    // Initial Vector
    vector v {1, 2, 3};

    // Print the contents of the vector
    for (int i=0; i < v.size(); i++) {
      cout << v[i] << "\n";
    }

    // Push 4 to the back of the vector
    v.push_back(4);

    // Print the contents again
    for (int i=0; i < v.size(); i++) {
      cout << v[i] << "\n";
    }
}
```

Run Code

See Explanation
Loading terminal (id_gixuguc), please wait...
The `4` has been added to the end of the vector!

0.2 On to an Exercise

In this section, you have learned about two useful tools:

- String streams, and
- `vector` `push_back`.

The string streaming objects in C++ are very powerful, and there are many more ways that an `istringstream` can be used. We encourage you to have a look at all of the available functions in the C++ reference. However, at this point you are ready for the next exercise. In this exercise, you will parse string lines from the board and store the `int`s in a vector using the `istringstream` and `push_back` methods from above. Have a careful look at the examples again, and copy paste any code you think you might need before clicking the `Next` button below.

In [ ]: