- Solution

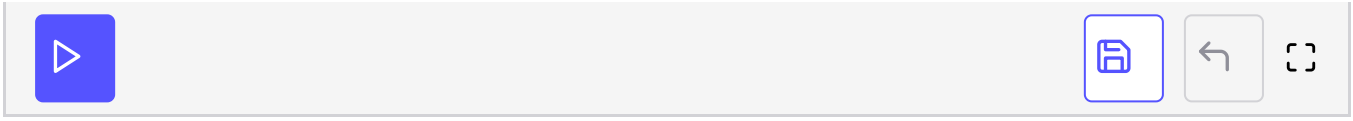
Let's take a look at the solution to the exercise.

WE'LL COVER THE FOLLOWING

Solution

```
#include <algorithm>
#include <functional>
#include <numeric>
#include <iostream>
#include <vector>

int main(){

  std::cout << std::endl;

  std::vector<int> myVec(20);
  std::iota(myVec.begin(), myVec.end(), 0);

  std::cout << "myVec: ";
  for (auto i: myVec) std::cout << i << " ";
  std::cout << std::endl;

  std::function< bool(int)> myBindPred= std::bind( std::logical_and<bool>(),
                                        std::bind( std::greater <int>(), std::placeholders::_

  myVec.erase(std::remove_if(myVec.begin(), myVec.end(), myBindPred), myVec.end());

  std::cout << "myVec: ";
  for (auto i: myVec) std::cout << i << " ";

  std::cout << "\n\n";

  std::vector<int> myVec2(20);
  std::iota(myVec2.begin(), myVec2.end(), 0);

  std::cout << "myVec2: ";
  for (auto i: myVec2) std::cout << i << " ";
  std::cout << std::endl;

  auto myLambdaPred = [](int a){return (a > 9) && (a < 16);};
```

```
  myVec2.erase(std::remove_if(myVec2.begin(), myVec2.end(), myLambdaPred), myVec2.end());


  std::cout << "myVec2: ";
  for (auto i: myVec2) std::cout << i << " ";

  std::cout << std::endl;

}
```



Explanation

- First, we populated the vectors using `std::iota` starting from value 0.
- The function `myBindPred` sets values which are less than 9 or greater than 16 in the count to true.
- In the code above, using a lambda function in line 35 allows us to achieve the same functionality in an easier way.
- The lambda function checks for each integer value greater than 9 and less than 16 and removes them from `myVec2` by using `remove_if`.

Now, let's dive into another utility of the C++ Standard Library – `std::pair`.