- Exercise

Let's solve an exercise in this lesson.

WE'LL COVER THE FOLLOWING

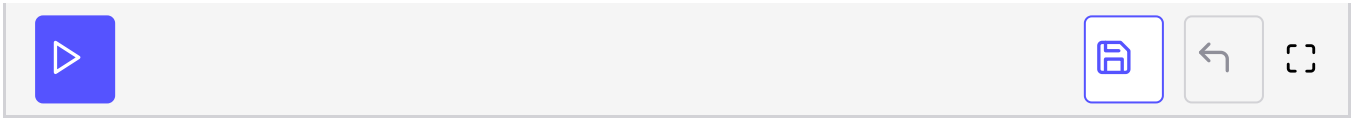
Problem Statement

Make the `public` member `elem` a `private` member of `Array`. How does that affect the implementation of the assignment operator?

Uncomment the code in `main()` to see what happens.

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

template <typename T, int N>
class Array{

public:
  Array()= default;

  template <typename T2>
  Array<T, N>& operator=(const Array<T2, N>& arr){
          static_assert(std::is_convertible<T2, T>::value, "Cannot convert source type to des
    elem.clear();
          elem.insert(elem.begin(), arr.elem.begin(), arr.elem.end());
          return *this;
  }

  int getSize() const;
  // you need to make the `elem` private and then run it
  std::vector<T> elem;
};

template <typename T, int N>
int Array<T, N>::getSize() const {
  return N;
}


int main(){
```

```
  // uncomment these line after the above implementation

  /*
  Array<double, 10> doubleArray{};
  Array<int, 10> intArray{};

  doubleArray = intArray;

  */
}
```



Let’s move on to the solution review of this exercise in the next lesson.