

- Example

Let's have a look at an example of template instantiation.

WE'LL COVER THE FOLLOWING ^

- Example: Template Instantiation
- Explanation

Example: Template Instantiation

```
// templateInstantiation.cpp

#include <iostream>
#include <vector>

template <typename T, int N>
class Array{

public:
    Array()= default;

    int getSize() const{
        return N;
    }

    std::vector<T> elem;
};

template<typename T>
bool isSmaller(T fir, T sec){
    return fir < sec;
}

template class std::vector<int>;
template bool std::vector<double>::empty() const;

template class Array<int, 20>;
template int Array<double, 5>::getSize() const;

template bool isSmaller(double, double);
template bool isSmaller<int>(int, int);

int main(){

    std::cout << std::endl;
```

```

std::cout << std::boolalpha << "implicit" << std::endl;

std::cout << std::endl;

std::vector<int> vec{};
std::cout << "vec.size(): " << vec.size() << std::endl;

Array<int, 10> arr;
std::cout << "arr.getSize(): " << arr.getSize() << std::endl;

std::cout << std::endl;

std::cout << "isSmaller(5, 10): " << isSmaller(5,10) << std::endl;

std::cout << "isSmaller<double>(5.5, 6.5): " << isSmaller<double>(5.5, 6.5) << std::endl;

std::cout << std::endl;
}

```



Explanation

In the above example, we have implemented a template class `Array` which includes a function `getSize()` that returns the size of the element `N` passed into the constructor. We have also defined a template function `isSmaller bool` and its return type is declared explicitly which returns `true` if the first passed argument is less than the second argument.

Lines 24 – 31 contain explicit template instantiation. The `main` program contains implicit template instantiation. Line 24 is an explicit instantiation for `int` and line 25 is an explicit instantiation of the method `getSize` for `double`. The lines 27 and 28 are quite similar for `Array`. The compiler can automatically deduce the template argument for the function argument in line 30.

In the next lesson, we'll solve a small exercise on template instantiation.