Counting Elements

Count algorithms assist us in counting the number of elements in a range which satisfy a certain predicate.

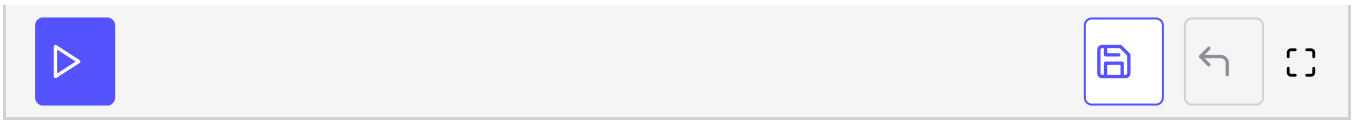
We can count elements (with or without a predicate) using STL algorithms.

They return the number of elements:

```
Num count(InpIt first, InpIt last, const T& val)
Num count(ExePol pol, FwdIt first, FwdIt last, const T& val)

Num count_if(InpIt first, InpIt last, UnPred pre)
Num count_if(ExePol pol, FwdIt first, FwdIt last, UnPred pre)
```

Count algorithms take input iterators as arguments and return the number of elements matching `val` or the predicate.

```
#include <algorithm>
#include <cctype>
#include <iostream>
#include <string>

int main(){

  std::cout << std::endl;

  std::string str{"abcdabAAAaefaBqeaBCQEaadsfdewAAQAaafbd"};
  std::cout << "count: " << std::count(str.begin(), str.end(), 'a') << std::endl;
  std::cout << "count_if: " << std::count_if(str.begin(), str.end(), [](char a){ return std::

  std::cout << std::endl;

}
```



std::count, and std::count_if

In the next lesson, we'll discuss a few algorithms which tell whether a value or values in a range fulfill our given condition.