Match

We learned about std::match_results. Now, we will look at one of the functions which allow us to send data to match_results.

std::regex_match determines if text matches a text pattern. You can further
analyze the search result of type std::match_results.

The code snippet below shows three simple applications of std::regex_match:
a C string, a C++ string and a range returning only a boolean. The three
variants are available for std::match_results objects respectively.

```
#include <iostream>
#include <regex>
#include <string>
#include <vector>
int main(){
  std::cout << std::endl;</pre>
  // regular expression for a number, not including an exponent
  std::string numberRegEx(R''([-+]?([0-9]*\.[0-9]+|[0-9]+))'');
  // regular expression holder
  std::regex rgx(numberRegEx);
  // using const char*
  const char* numChar{"2011"};
  if (std::regex match(numChar, rgx)){
    std::cout << numChar << " is a number." << std::endl;</pre>
  }
  // using std::string
  const std::string numStr{"3.14159265359"};
  if (std::regex_match(numStr, rgx)){
    std::cout << numStr << " is a number." << std::endl;</pre>
  }
  // using bidirectional iterators
  const std::vector<char> numVec{{'-', '2', '.', '7', '1', '8', '2', '8', '1', '8', '2', '8'}
  if (std::regex_match(numVec.begin(), numVec.end(), rgx)){
   for (auto c: numVec){ std::cout << c ;};</pre>
    std::cout << " is a number." << std::endl;</pre>
  }
  std::cout << std::endl;</pre>
```

std::match