Search

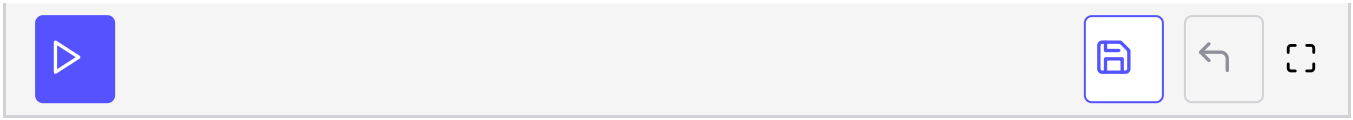
In this lesson, we'll see the implementation of the second look-up function for regex statements: regex_search.

`std::regex_search` checks if a text contains a text pattern. We can use the function with and without an `std::match_results` object and apply it to a C string, a C++ string, or a range.

The example below shows how to use `std::regex_search` with texts of type `const char*`, `std::string`, `const wchar_t*`, and `std::wstring`.

```
#include <iostream>
#include <regex>
#include <string>

int main(){

  std::cout << std::endl;

  // regular expression holder for time
  std::regex crgx("([01]?[0-9]|2[0-3]):[0-5][0-9]");

  // const char*
  std::cout << "const char*" << std::endl;
  std::cmatch cmatch;

  const char* ctime{"Now it is 23:10."};

  if (std::regex_search(ctime, cmatch, crgx)){

     std::cout << ctime << std::endl;
     std::cout << "Time: " << cmatch[0] << std::endl;

   }

  std::cout << std::endl;

  // std::string
  std::cout << "std::string" << std::endl;
  std::smatch smatch;

  std::string stime{"Now it is 23:25."};
  if (std::regex_search(stime, smatch, crgx)){

    std::cout << stime << std::endl;
    std::cout << "Time: " << smatch[0] << std::endl;

  }
```

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

  // regular expression holder for time
  std::wregex wrgx(L"([01]?[0-9]|2[0-3]):[0-5][0-9]");

  // const wchar_t
  std::cout << "const wchar_t* " << std::endl;
  std::wcmatch wcmatch;

  const wchar_t* wctime{L"Now it is 23:47."};

  if (std::regex_search(wctime, wcmatch, wrgx)){

       std::wcout << wctime << std::endl;
       std::wcout << "Time: " << wcmatch[0] << std::endl;

  }

  std::cout << std::endl;

  // std::wstring
  std::cout << "std::wstring" << std::endl;
  std::wsmatch wsmatch;

  std::wstring  wstime{L"Now it is 00:03."};

  if (std::regex_search(wstime, wsmatch, wrgx)){

    std::wcout << wstime << std::endl;
    std::wcout << "Time: " << wsmatch[0] << std::endl;

  }

  std::cout << std::endl;

}
```



std::search

In the next lesson, we'll solve an exercise to test our knowledge of `std::regex`.