simdutf library

- C++11 compiler
- Grab simdutf.h and simdutf.cpp from https://github.com/simdutf/simdutf/releases/download/v1.0.0/singleheader.zip

Validating UTF-8

```
#include <iostream>

#include "simdutf.cpp"

int main() {
  const char *source = "1234";
  bool validutf8 = simdutf::validate_utf8(source, strlen(source));
  if (validutf8) {
    std::cout << "valid UTF-8" << std::endl;
    return EXIT_SUCCESS;
  } else {
    std::cerr << "invalid UTF-8" << std::endl;
    return EXIT_FAILURE;
  }
}
```

```
c++ example1.cpp -std=c++11
```

Predicting UTF-16 size

```
#include <iostream>

#include "simdutf.cpp"

int main() {
  const char *source = "1234";
  size_t expected_utf16words = simdutf::utf16_length_from_utf8(source, strlen(source));
  std::cout << "expected UTF-16 words: " << expected_utf16words << std::endl;
  return EXIT_SUCCESS;
}
```

Converting UTF-16

```
#include <iostream>

#include "simdutf.cpp"

int main() {
  const char *source = "1234";
  size_t utf8_length = strlen(source);
  size_t expected_utf16words = simdutf::utf16_length_from_utf8(source, utf8_length);
  std::unique_ptr<char16_t[]> utf16_output{new char16_t[expected_utf16words]};
  size_t utf16words =
      simdutf::convert_utf8_to_utf16(source, utf8_length, utf16_output.get());
  std::cout << "wrote " << utf16words << " UTF-16 words." << std::endl;
  return EXIT_SUCCESS;
}
```

Validating UTF-16

```
#include <iostream>

#include "simdutf.cpp"

int main() {
  const char16_t * source = u"1234";
  bool validutf16 = simdutf::validate_utf16(source, 4);
  if (validutf16) {
    std::cout << "valid UTF-16" << std::endl;
    return EXIT_SUCCESS;
  } else {
    std::cerr << "invalid UTF-16" << std::endl;
    return EXIT_FAILURE;
  }
}
```

Predicting UTF-8 size

```
#include <iostream>

#include "simdutf.cpp"

int main() {
  const char16_t * source = u"1234";
  size_t expected_utf8words = simdutf::utf8_length_from_utf16(source, 4);
  std::cout << "expected UTF-8 words: " << expected_utf8words << std::endl;
  return EXIT_SUCCESS;
}
```

Converting UTF-16

```
#include <iostream>

#include "simdutf.cpp"

int main() {
  const char16_t * source = u"1234";
  size_t expected_utf8words = simdutf::utf8_length_from_utf16(source, 4);
  std::unique_ptr<char[]> utf8_output{new char[expected_utf8words]};
  size_t utf8words =
      simdutf::convert_utf16_to_utf8(source, 4, utf8_output.get());
  std::cout << "wrote " << utf8words << " UTF-8 words." << std::endl;
  return EXIT_SUCCESS;
}
```