

Взаимно однозначный англо-русский транслитератор

Документация к программам

1 Общее описание

Так как автор предполагает, что основная область применения взаимно однозначной транслитерации – это компьютерные системы, им был создан набор программ, производящих ее. Он состоит из нескольких исполняемых файлов для запуска из командной строки и библиотеки на языке C++.

2 Исполняемые файлы

russian_to_english:

./russian_to_english

./russian_to_english <файл ввода>

./russian_to_english <файл ввода> <файл вывода>

Однозначно транслитерирует текст: все слова из русских букв заменяет соответствующими словами из английских (см. инструкцию по транслитерации), прочие символы оставляет без изменений.

Ввод и вывод в формате UTF-8.

В случае неверного числа аргументов или отсутствия файла ввода выведет сообщение об ошибке и завершится с кодом 1.

english_to_russian:

./english_to_russian

./english_to_russian <файл ввода>

./english_to_russian <файл ввода> <файл вывода>

Однозначно транслитерирует текст: все слова из английских букв заменяет соответствующими словами из русских (см. инструкцию по транслитерации), прочие символы оставляет без изменений.

Ввод и вывод в формате UTF-8.

В случае неверного числа аргументов или отсутствия файла ввода выведет сообщение об ошибке и завершится с кодом 1.

tests:

./tests

Тестирует корректность транслитерации: перебирает слова из определенных наборов русских и английских букв, переводит на другой язык, затем переводит обратно и сравнивает с исходным словом. Выводит результаты проверки.

Вывод в формате UTF-8.

3 Библиотека на C++

Пространство имен **RuEnTransliterat**or.

std::wstring russianToEnglish(const std::wstring& text):

text – широкосимвольная строка русского текста

Однозначно транслитерирует текст: все слова из русских букв заменяет соответствующими словами из английских (см. инструкцию по транслитерации), прочие символы оставляет без изменений.

Внимание: функция не потокобезопасна!

std::wstring englishToRussian(const std::wstring& text):

text – широкосимвольная строка английского текста

Однозначно транслитерирует текст: все слова из английских букв заменяет соответствующими словами из русских (см. инструкцию по транслитерации), прочие символы оставляет без изменений.

Внимание: функция не потокобезопасна!

std::wstring wstringFromUTF8(const std::string& bytes):

bytes – стандартная строка, содержащая текст в формате UTF-8; в частности, подходит строка со стандартного ввода при кодировке UTF-8 в командной строке

Возвращает широкосимвольную строку, содержащую данный текст в ее внутренней кодировке.

std::string wstringToUTF8(const std::wstring& text):

text – широкосимвольная строка, содержащая текст в своей внутренней кодировке

Возвращает стандартную строку, содержащую данный текст в кодировке UTF-8; в частности, строка подходит для стандартного вывода при кодировке UTF-8 в командной строке.

bool readFileUTF8(const std::string& filename, std::wstring& text, EncodingInfo& info):

filename – имя файла;

text – широкосимвольная строка текста;

info – информация о файле (о наличии маркера BOM)

Читает содержимое файла в формате UTF-8 в широкосимвольную строку. Информация о файле сохраняется в поле *info.has_bom*. Возвращает *true* в случае успеха, *false* иначе.

bool writeFileUTF8(const std::string& filename, const std::wstring& text, const EncodingInfo& info):

filename – имя файла;

text – широкосимвольная строка текста;

info – информация о файле (о наличии маркера BOM)

Пишет содержимое широкосимвольной строки в файл в формате UTF-8. В файл пишется маркер BOM, если в таковое указано в поле *info.has_bom*. Возвращает *true* в случае успеха, *false* иначе.

Функции *russianToEnglish* и *englishToRussian* не потокобезопасны из-за использования объектов в статической памяти. Для многопоточных программ использовать класс **Transliterator**:

- *Transliterator::Transliterator()*;
- *void Transliterator::setup()* – инициализация, вызвать перед использованием;
- *std::wstring Transliterator::russianToEnglish(const std::wstring&)*;
- *std::wstring Transliterator::englishToRussian(const std::wstring&)*.