

Качество и надежность программного обеспечения		
Внутренняя спецификация	ФИО студента	Козарез М.В.
	Группа	ПрИн-266
	Дата выполнения	
	Преподаватель	Сычев О.А.
	Оценка	

Оглавление

1 Общие сведения.....	2
2 Функциональное назначение	2
3 Описание логической структуры.....	2
4 Декомпозиция программы.....	2
5 Вызов и загрузка.....	3
6 Входные и выходные данные.....	3
7. Псевдокоды основных функций	4
7.1 Псевдокод функции inputFile.....	4
7.2 Псевдокод функции translit	4
7.3 Псевдокод функции decodeString	4
7.4 Псевдокод функции outputFile.....	5
Приложение А	6
Основные типы и структуры данных программы	6
Приложение Б	7
Спецификация функций программы и выделение подпрограммы	7
Приложение В.....	8

Дерево вызова функций.....	8
Приложение Г	9
DFD-диаграмма	9

1 Общие сведения

Наименование программы: «Программа для обратной транслитерации

русского текста с латиницы».

В состав технических средств должен входить персональный компьютер, включающий в себя:

- процессор Intel Pentium с тактовой частотой 2.10 ГГц, не менее;
- операционная система Windows 7 и более поздней версии;
- оперативную память объёмом 512 Мб, не менее;
- клавиатуру;
- компьютерную мышь.

2 Функциональное назначение

Программа предназначена для обратной транслитерации русского текста с латиницы

3 Описание логической структуры

Программа считывает входные данные из файла формата .txt, в котором должна находиться строка с текстом, состоящим только из латинских букв и пробелов.

4 Декомпозиция программы

- Основные типы и структуры данных программы представлены в приложении А.

- Спецификации функций программы представлены в приложении Б.
- Иерархия вызова подпрограмм в приложении В.
- Диаграмма потоков данных представлена в приложении Г.

5 Вызов и загрузка

Программа запускается посредством файла с расширением .exe с указанием таких параметров как:

- название файла с входными данными, находящегося в директории с запускаемым .exe файлом;
- название файла с расширением .txt, в который будет записан результат работы программы;

6 Входные и выходные данные

Пример текста во входном файле	Результат
qwe	куве
ezhik	езхик ежик
yozhik	йозхик йожик ёзхик ёжик
Ya Proveryayu Programmu	Йа Проверйайу Программу Йа Проверйаю Программу Йа Проверяйу Программу Йа Проверяю Программу Я Проверйайу Программу Я Проверйаю Программу Я Проверяйу Программу

7. Псевдокоды основных функций

7.1 Псевдокод функции inputFile

```
// Открываем файл с входными данными
// Проверяем, был ли файл открыт успешно
// Если файл не был открыт успешно – выдать ошибку
// Проверяем количество строк в файле
// Если строк больше 1 – выдать ошибку
// Проверяем содержимое файла
    /* Если в файле имеются символы, отличные от латиницы в
    верхнем и нижнем регистре и пробелов – выдать ошибку */
// Записываем строку в переменную
// Возвращаем переменную из функции
```

7.2 Псевдокод функции translit

```
// Перевести копию входной строки в нижний регистр
/* Вектор подгоняем размеры вектора, в котором будет происходить
транслитерация под размеры строки */
// Производим транслитерацию строки
// Возвращаем вектор с готовым транслитом
```

7.3 Псевдокод функции decodeString

```
/* Если мы вышли за пределы вектора с транслитом – вернуть вектор
с транслитами */
// Иначе
    /* Пока мы не дошли до максимальной длины комбинации
символов и пока текущая позиция в сумме с текущей длиной
комбинации символов не выходят за пределы строки */
    /* Ищем в алфавите перевода нужную подстроку и заменяем
```

```
текущий символ на неё */  
    /* Запускаем функцию decodeString для суммы текущей позиции  
и длины комбинации символов */  
    /* Если текущий символ был в верхнем регистре – переводим  
текущий символ в верхний регистр */  
    // Добавляем текущий символ к концу строки  
    /* Добавляем строку с транслитом к текущей позиции вектора с  
транслитами */  
    // Возвращаем вектор с текущей строкой
```

7.4 Псевдокод функции outputFile

```
// Открываем/создаем файл для выходных данных  
// Проверяем, был ли файл открыт успешно  
// Если файл не был открыт успешно – выдать ошибку  
// Для каждого элемента вектора с транслитом  
// Переписываем в файл строки начиная с первой  
// Закрываем файл
```

Основные типы и структуры данных программы

```
string line;
```

```
map<string, vector<string>>decMap;
```

```
vector<vector<string>>decodes;
```

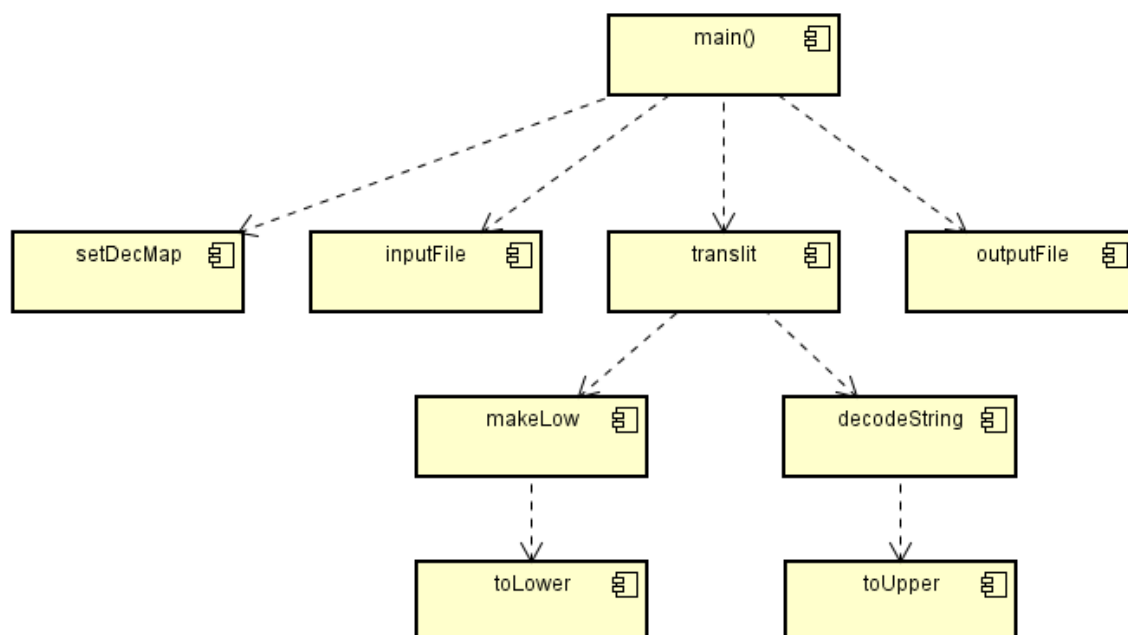
```
vector<bool>upper;
```

Спецификация функций программы и выделение подпрограммы

Алгоритм:

- 1) Инициализировать карту перевода
- 2) Получить название входного файла
- 3) Если невозможно открыть файл
 - 2.1) Завершить работу программы с текстом ошибки: «Ошибка при открытии файла»
- 4) Если количество строк во входном файле больше одной
 - 3.1) Завершить работу программы с текстом ошибки: «Количество строк в файле не равно 1!»
- 5) Если во входном файле есть символы, отличные от латинских букв в верхнем и нижнем регистре и пробелов
 - 4.1) Завершить работу программы с текстом ошибки: «В исходной строке есть недопустимые символы!»
- 6) Произвести транслитерацию исходного текста
- 7) Создать/Открыть файл, в который будет записана транслитерация исходного текста
- 8) Если невозможно открыть файл
 - 7.1) Завершить работу программы с текстом ошибки: «Ошибка при открытии файла!»

Дерево вызова функций



DFD-диаграмма

