Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования  
«Волгоградский государственные технический университет»

УТВЕРЖДАЮ:  
 заведующий кафедрой ПОАС   
 \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Орлова Ю.А.  
 «\_\_\_»\_\_\_\_\_\_\_\_\_ 2020 г.

КАЧЕСТВО И НАДЕЖНОСТЬ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ  
«Расшифровать сообщение на азбуке Морзе.»  
Описание программы

СОГЛАСОВАНО: Разработчик:  
руководитель работы: студент ПрИн-266  
доцент кафедры ПОАС \_\_\_\_\_\_\_\_\_\_\_\_ Тарапатина Е.С.  
\_\_\_\_\_\_\_\_\_\_ Сычев О.А. «\_\_\_»\_\_\_\_\_\_\_\_\_ 2020 г.  
«\_\_\_»\_\_\_\_\_\_\_\_\_ 2020 г.

Нормоконтролер:  
 преподаватель кафедры ПОАС  
 \_\_\_\_\_\_\_\_\_\_ Матюшечкин Д.С.  
 «\_\_\_»\_\_\_\_\_\_\_\_\_ 2020 г.

2020

Содержание

[1 Общие сведения 4](#_Toc38635257)

[2 Функциональное назначение 4](#_Toc38635258)

[3 Описание логической структуры 4](#_Toc38635259)

[4 Используемые технические средства 5](#_Toc38635260)

[5 Вызов и загрузка 5](#_Toc38635261)

[6 Входные и выходные данные 5](#_Toc38635262)

[Приложение 1 6](#_Toc38635263)

[Приложение 2 7](#_Toc38635264)

# 1 Общие сведения

Программа называется «Программа для расшифровки сообщения на азбуке Морзе».

Для корректного функционирования программы необходима операционная система Windows XP и выше.

Данная программа будет написана на языке программирования С++.

# 2 Функциональное назначение

Программа «Программа для расшифровки сообщения на азбуке Морзе» является программой для предоставления вариантов расшифровки сообщения на азбуке Морзе.

Разработка может быть использована для расшифровки и предоставления вероятных вариантов расшифровки сообщения на азбуке Морзе.

# 3 Описание логической структуры

Данный алгоритм рекурсивный. Состоит из двух этапов, которые подробнее описаны в приложении 1.

Первый этап строит возможный перевод исходной строки путем поиска нужной буквы в словаре и собиранием ее через рекурсию.

Второй этап состоит же в том, что когда исходная строка переведена (т.е. пустая), возможный промежуточный вариант добавляется к возможным вариантам перевода.

# 4 Используемые технические средства

Описание требуемых технических средств содержится в Техническом задании в пункте 3.3.

# 5 Вызов и загрузка

Для запуска программы нужно в командной строке ввести следующие параметры: название exe-файла программы и имя файла с текстовым описанием.

# 6 Входные и выходные данные

Формат входных и выходных данных описан в пункте 3.4. Технического задания.

# Приложение 1

Описание структур данных

Вспомогательный словарь вида буква на азбуке Морзе – ее перевод.

const vector<pair<string, char>> Dictionary = {

{".-", 'A'}, {"-…", 'Б'}, {".--", 'В'},

{"--.",'Г'}, {"-..",'Д'}, {".",'Е'},

{"…-",'Ж'}, {"--..",'З'}, {"..",'И'},

{".---",'Й'}, {"-.-",'К'}, {".-..",'Л'},

{"--",'М'}, {"-.",'Н'}, {"---",'О'},

{".--.",'П'}, {".-.",'Р'}, {"…",'С'},

{"-",'Т'}, {"..-",'У'}, {"..-.",'Ф'},

{"….",'Х'}, {"-.-.",'Ц'}, {"---.",'Ч'},

{"----",'Ш'}, {"--.-",'Щ'}, {".--.-.",'Ъ'},

{"-.--",'Ы'}, {"-..-",'Ь'}, {"…-…" , 'Э'},

{"..--" , 'Ю'}, {".-.-" , 'Я'}, {"-----" , '0'},

{".----" , '1'}, {"..---" , '2'}, {"…--" , '3'},

{"….-" , '4'}, {"….." , '5'}, {"-…." , '6'},

{"--…" , '7'}, {"---.." , '8'}, {"----." , '9'},

};

# Приложение 2

Описание алгоритмов функций

Функция void DecodeMorze (const string& DecodeString, vector<string>& DecodedVariants, const string& BuildString);

Входные данные:

DecodeString - исходная строка, зашифрованная в азбуке Морзе

BuildString - строка для временного хранения промежуточных данных

Выходные данные:

DecodedVariants - вектор расшифрованных вариантов

Алгоритм:

Изначально работаем с копией строки промежуточных данных

Изначально имеем словарь вида буква-ее перевод

Если исходная строка пустая, то необходимо добавить строку с промежуточными данными в список возможных переводов

Иначе

Поиск в словаре буквы, которая находится в начале исходной строки

Записать перевод найденной буквы в строку с промежуточными данными

Создать копию исходной строки без первой переведённой буквы

Рекурсивно вызвать эту же функцию, используя новые значения промежуточных данных и копию подстроки

Очистить строку с промежуточными данными от перевода текущей буквы

Функция bool CheckStringForInvalidChar(const string& CheckedString);

Входные данные:

CheckedString – строка, которая проверяется на наличие посторонних символов

Выходные данные:

Наличие посторонних символов в строке (true – есть в наличие, false – нет в наличии)

Алгоритм:

Изначально считаем, что посторонних символов в строке нет

Просматриваем каждый символ в строке

если обнаружен посторонний символ, то выдаем ошибку

Если посторонних символов не обнаружено, то говорим об отсутствии посторонних символов

# Приложение 3

Диаграмма вызовов и диаграмма потоков данных

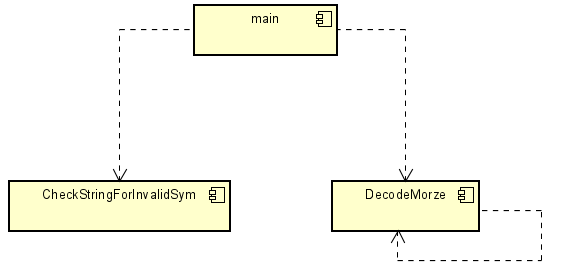


Рис. 1. Дерево вызовов

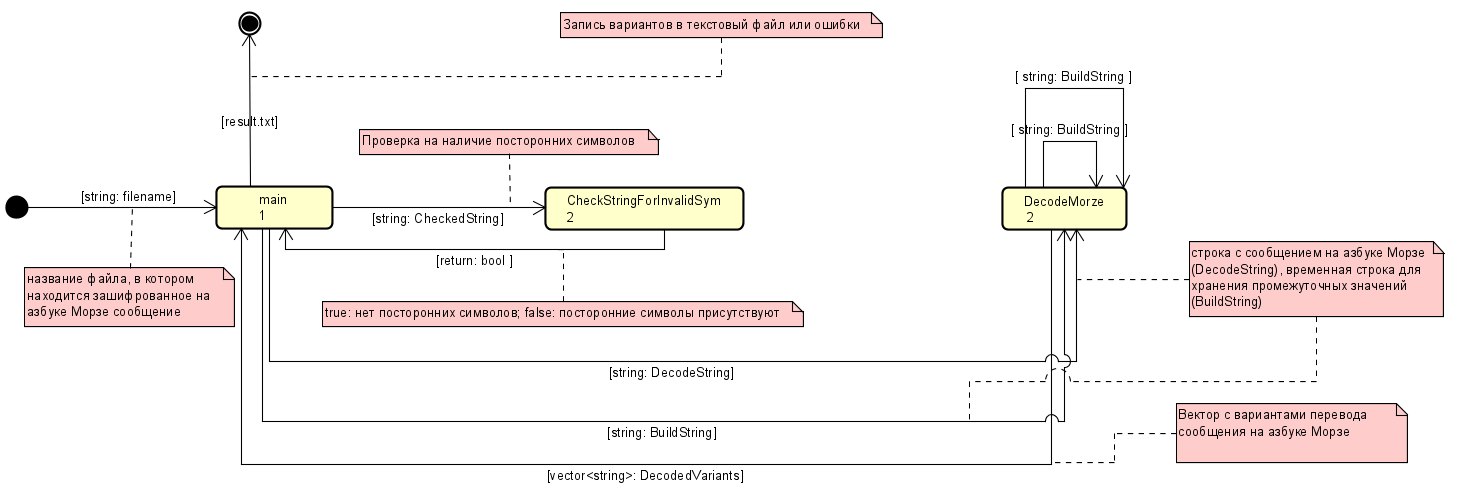


Рис. 2. Диаграмма потоков данных