**Министерство образования и науки Российской Федерации**

Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет

«Высшая школа экономики»

Кафедра Компьютерной безопасности

**ОТЧЕТ**

**К ДОМАШНЕЙ РАБОТЕ №1**

**по дисциплине**

**«Языки программирования»**

Работу выполнил

студент группы СКБ182 \_\_\_\_\_\_\_\_\_\_\_\_\_ Р.Р. Модзгвришвили

подпись, дата

Работу проверил \_\_\_\_\_\_\_\_\_\_\_\_\_ С. А. Булгаков

Подпись, дата

Оглавление

[1. Постановка задачи 2](#_Toc529957813)

[2.1 Общая идея решения: 3](#_Toc529957814)

[2.2 Конкретика решения (main.cpp) 3](#_Toc529957815)

[2.3 Заголовочный файл *input.hpp* 3](#_Toc529957816)

[2.4 Файл, задающий функцию. Input.cpp 3](#_Toc529957817)

[3. Тестирование 4](#_Toc529957818)

[Приложение А 5](#_Toc529957819)

[Приложение Б 6](#_Toc529957820)

[Приложение В 6](#_Toc529957821)

[Приложение Г 7](#_Toc529957822)

# Постановка задачи

Разработайте функцию, соответствующую прототипу,

SymbolType TranslateCodeToSymbol ( CodeType code )

выполняющую преобразование «кода клавиши» гипотетической клавиатуры в «символ», где

SymbolType и CodeType задаются как

typedef int CodeType;

typedef char SybmolType;

при этом, младшие 8 бит CodeType содержат код клавиши, а преобразование в SybmolType вы-

полняется согласно таблице 1.

Работоспособность функции TranslateCodeToSymbol() продемонстрировать разрабо-

тав программу выполняющую преобразование входного потока «кодов» в выходной поток «симво-

лов». Протестировать программу на данных из таблицы 2, согласно варианту.

Входные данные в программу передаются из текстового файла с использованием перена-

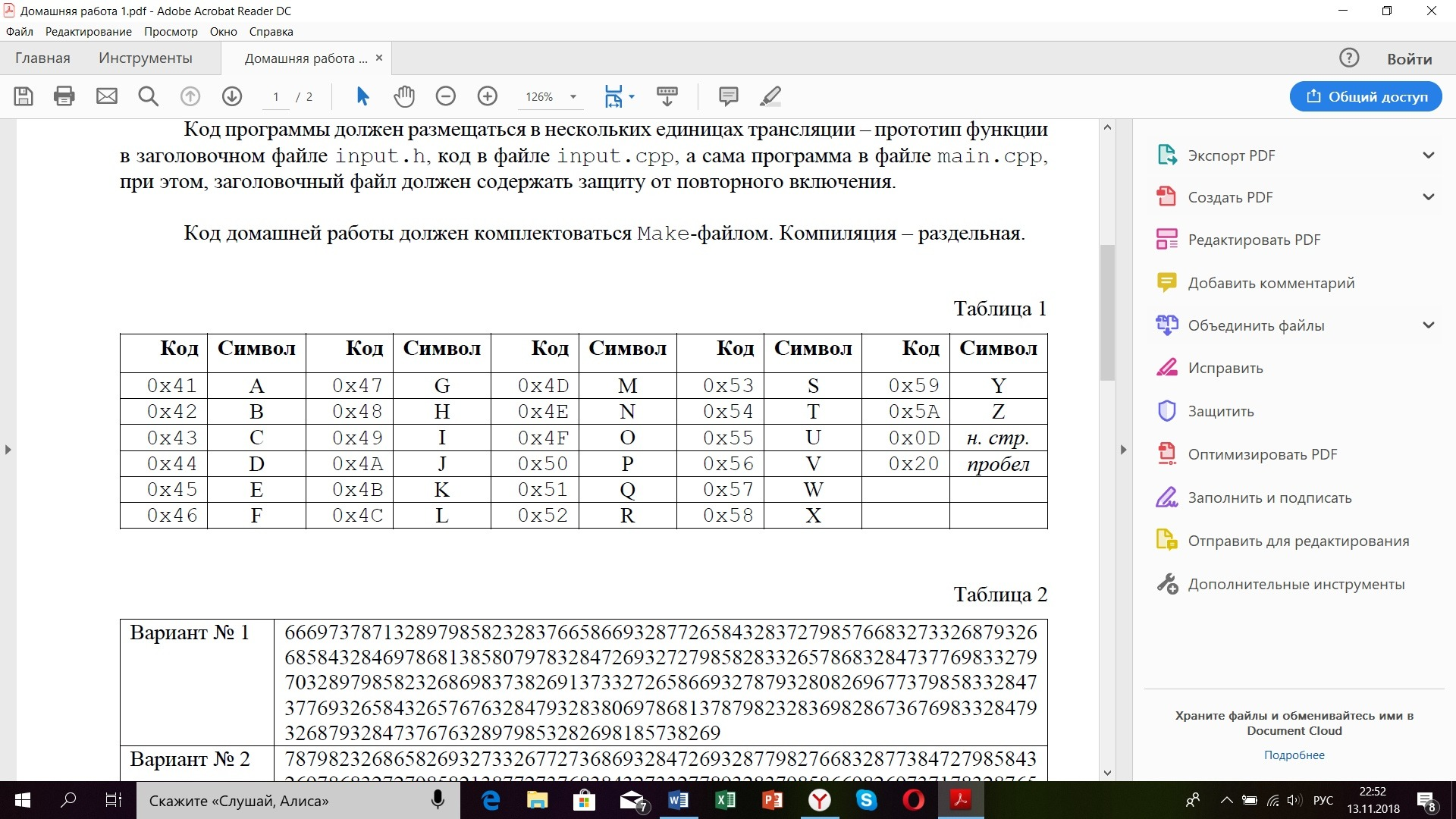
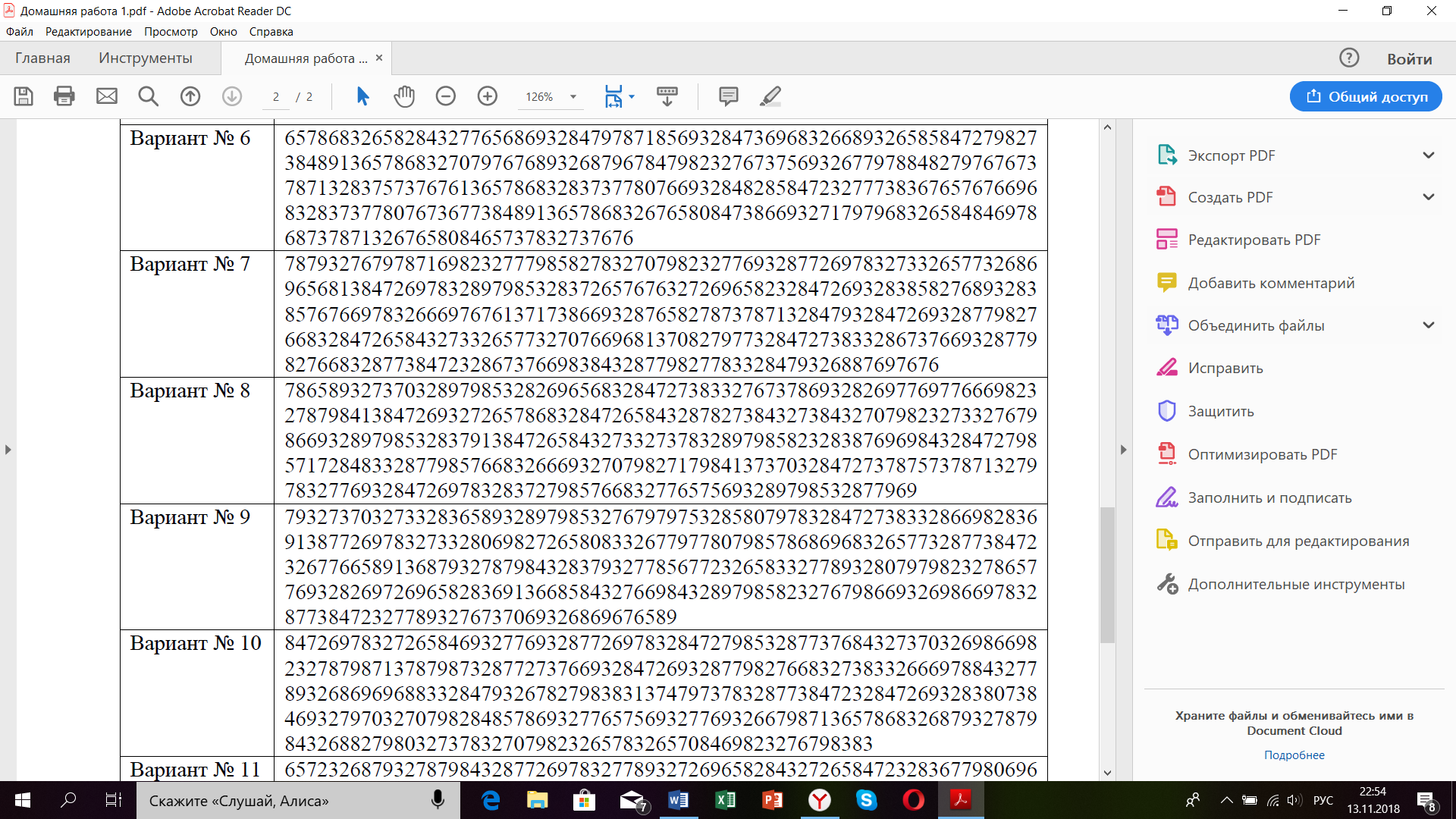
правления ввода.

Запрещается использовать массивы, строки Си++, а также динамическую память.

Код программы должен размещаться в нескольких единицах трансляции – прототип функции

в заголовочном файле input.h, код в файле input.cpp, а сама программа в файле main.cpp,

при этом, заголовочный файл должен содержать защиту от повторного включения.



# Общая идея решения:

В качестве входных данных мы имеем набор чисел, записанных в файл *test.txt*. Далее мы посимвольно считываем по одной цифре из этого файла, чтобы далее попарно их объединить в двузначные числа, которые будут записаны в переменные типа *int*. После чего они будут преобразованы в символы при помощи команды *TranslateCodeToSymbol*.

# Конкретика решения (main.cpp)

В данном пункте мы будем использовать принцип “Считывать пока считывается”. Двузначный символ будет записан в том случае, если переменные, которые отвечают за прием символов будут иметь значения. Причем считывание будет происходить до тех пор, пока в файле есть несчитанные символы.

**while ((std::cin>>sym1) && (std::cin>>sym2))**

После того, как в *Sym1* и в *Sym2* , были записаны некоторые значения, нужно объединить их в единое двузначное число. За это будет отвечать следующая строчка.

**result = 10\*( sym1-'0')+(sym2-'0');**

Данные строки кода содержатся в файле *main.cpp* (Приложение А)

# Заголовочный файл *input.hpp*

Данный файл содержит защиту от повторного включения. Это будет реализовано при помощи включения директив #ifndef, #define, #endif. Так же в файле определены SymbolType и CodeType. Задаются как:

**typedef int CodeType;**

**typedef char SymbolType;**

SymbolType и CodeType использованы в функции TranslateCodeToSymbol

**SymbolType TranslateCodetoSymbol(CodeType code);**

(приложение Б)

# Файл, задающий функцию. Input.cpp

Для начала подключается заголовочный файл. Затем в данном файле определяется работа функции TranslateCodeToSymbol как функция, которая переводит код в символ. (Приложение В)

**SymbolType TranslateCodetoSymbol(CodeType code)**

**{**

**return static\_cast<char>(code);**

**}**

# Тестирование

Подадим программе на ввод значения 50515253. Программа должна вывести результат “2345”.

**./main <test.txt**

**2345**

**Done**

Дадим программе на ввод значения 5013513252. После чего программа должна вывести матрицу.

**./main <test.txt**

**2**

**3 4**

**Done**

А теперь запишем в test.txt задания Варианта №9:

**./main <test.txt**

**O IF I SAY YOU LOOK UPON THIS VERSE**

**WHEN I PERHAPS COMPOUNDED AM WITH CLAY**

**DO NOT SO MUCH AS MY POOR NAME REHEARSE**

**BUT LET YOUR LOVE EVEN WITH MY LIFE DECAY**

**Done**

# Приложение А

#include <iostream>

#include "input.hpp"

int main()

{

char sym1 = '\x1', sym2 = '\x1';

int result;

while ((std::cin>>sym1) && (std::cin>>sym2))

{

result = 10\*(sym1-'0')+(sym2-'0');

if (result==13)

{

result = 10;

}

std::cout<<TranslateCodetoSymbol(result);

}

std::cout<<std::endl;

std::cout<<"Done"<<std::endl;

}

# Приложение Б

#ifndef INPUT\_HPP

#define INPUT\_HPP

typedef int CodeType;

typedef char SymbolType;

SymbolType TranslateCodetoSymbol(CodeType code);

#endif

# Приложение В

#include "input.hpp"

SymbolType TranslateCodetoSymbol(CodeType code)

{

return static\_cast<char>(code);

}

# Приложение Г

Начало работы программы

Считываем данные пока вводятся

Проверка корректности ввода

Объединение в единое значение

Перевод числа в символ

Вывод

Завершение работы программы

Ввод некорректен

Ввод закончен