Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

Специальность: Обеспечение информационной безопасности автоматизированных систем

**ОТЧЕТ**

ПО УЧЕБНОЙ ПРАКТИКЕ  
*(указать вид (этап) практики)*

Профессиональный модуль:

Машинно-ориентированное программирование в защите информации

*(наименование профессионального модуля\_)*

Выполнил:

обучающийся учебной группы   
№3ОИБАС-818

В.А. Ефременков

*(подпись) (И.О. Фамилия)*

Руководитель практики от колледжа:

И.В. Сибирев

*(И.О. Фамилия)*

*(оценка) (подпись)*

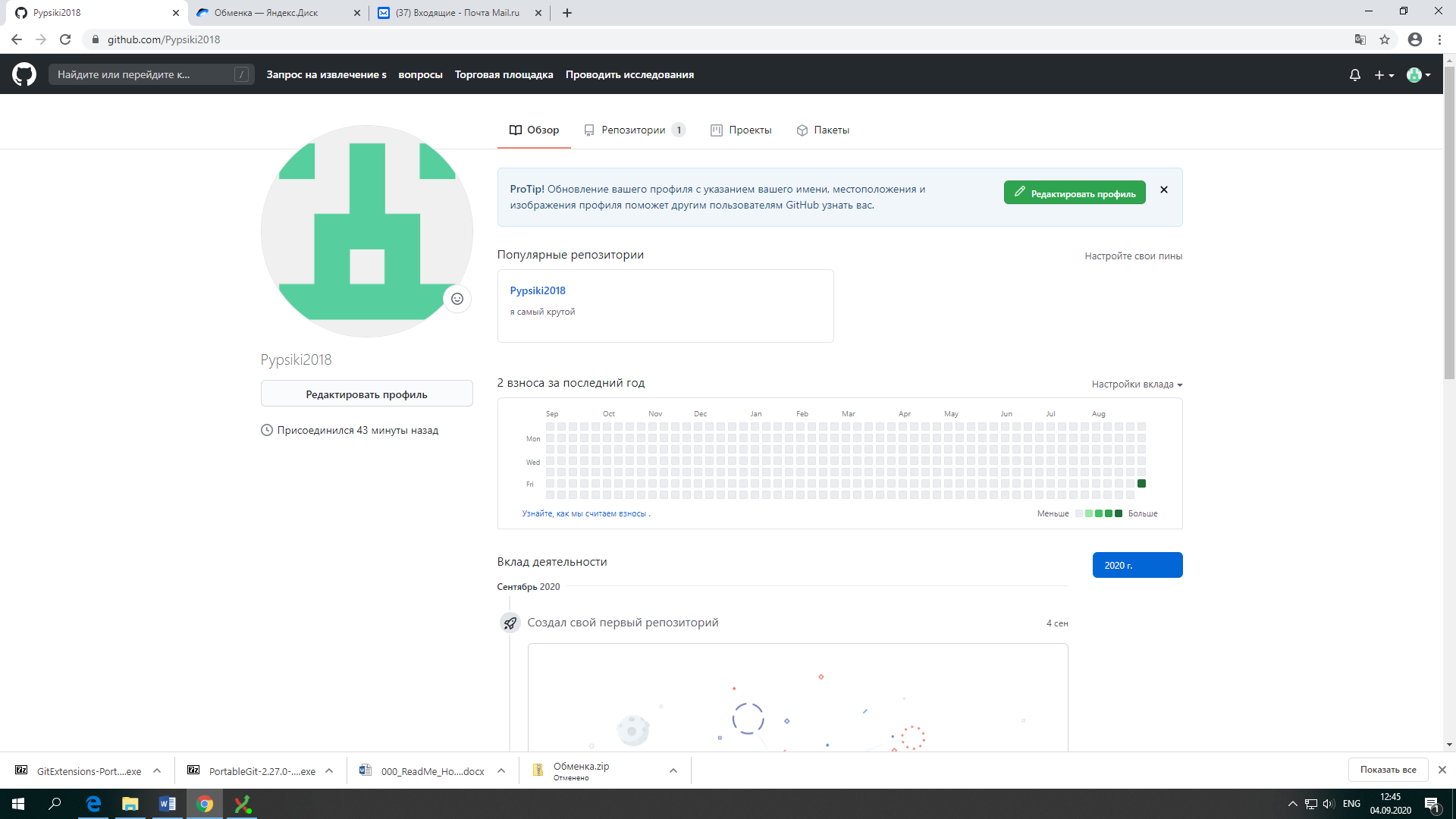
**Москва – 2020г.**

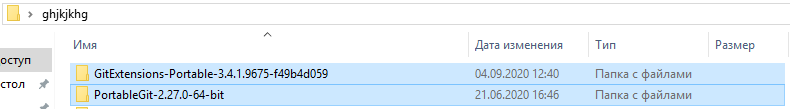
**L1.0 Знакомство с GitHub**

**Задание 1**

Требуется зарегистрироваться на GitHub.com (<https://github.com>). Склонировать к себе репозиторий, создать в рабочем каталоге файл, сделать комит и отправить на центральный сервер.

Ход выполнения лабораторной работы

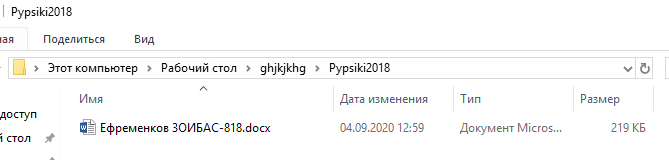
Я зарегистрировался на GitHub.com, Логин: Pypsiki2018

Дальше я скачал с Яндес.Диска «PortableGit-2.27.0-64-bit» и «GitExtensions-Portable-3.4.1.9675-f49b4d059». Затем поместил это все в папку с англоязычным названием.

После этого открыл «GitExtensions-Portable-3.4.1.9675-f49b4d059» и запустил файл «GitExtensions.exe».

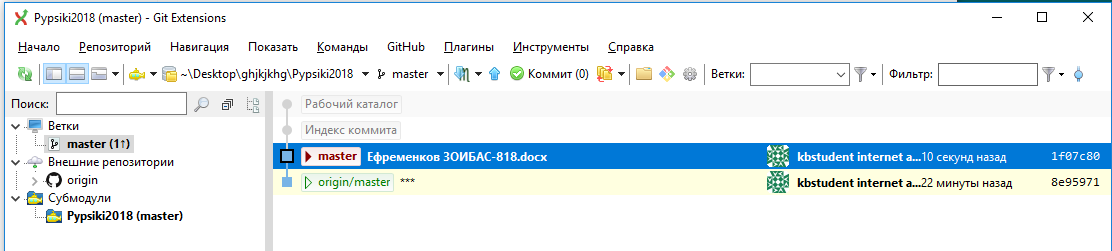
Далее нажал «GitHub», «Клонировать репозиторий».

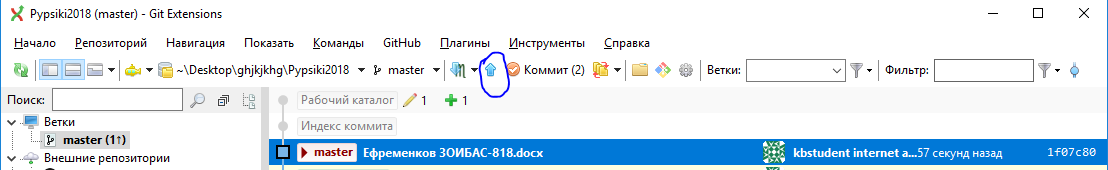
Выбрал созданную ранее папку, поставил протокол Https и нажал «Клонировать»

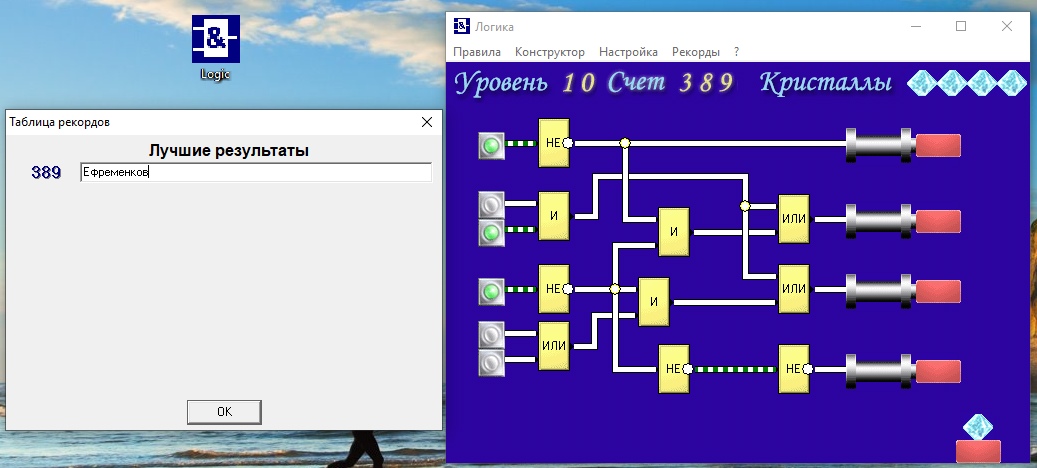
Далее закидываем этот Word в клонированную папку.

Открываем Git Extensions и нажимаем Коммит.

Далее выбираем свой Word документ и нажимаем зафиксировать.

У нас появился Word файл, который мы зафиксировали

После чего нажимаем кнопку «Push» и наш файл улетает в нашу репозиторию на сайте.

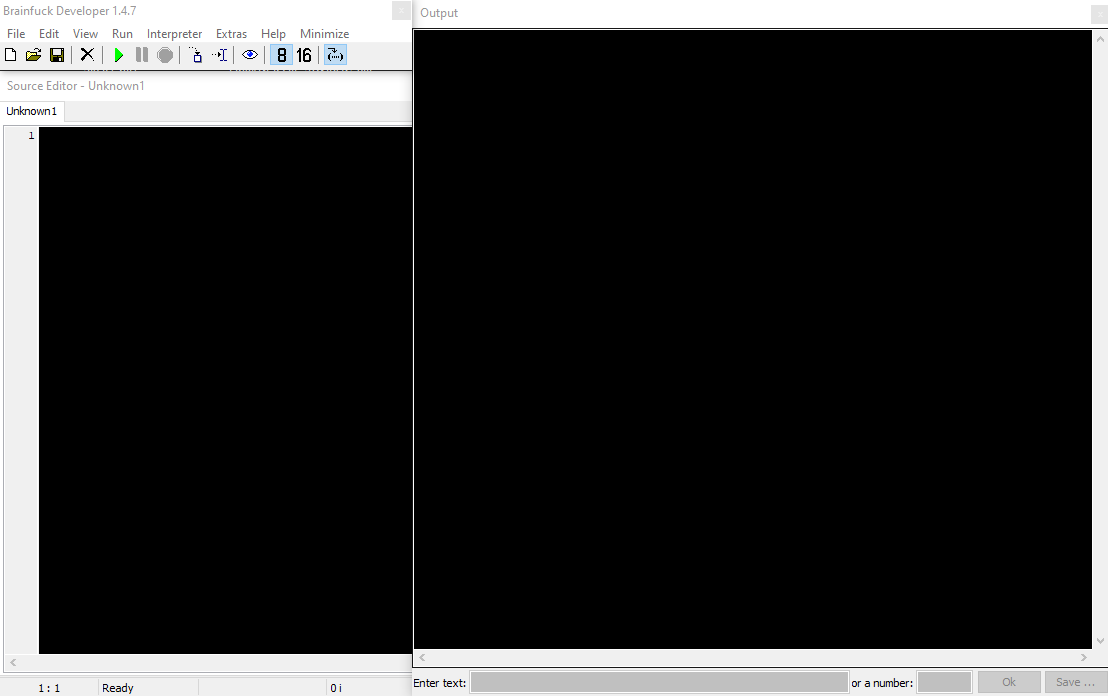
**L1.1 Logic**  
Открываем Яндекс.Диск и скачиваем программу «Logic». После открытия проходим 10 уровней и видим следующее:

И находим пасхалку:

**L1.2 BrainFОнлайн**

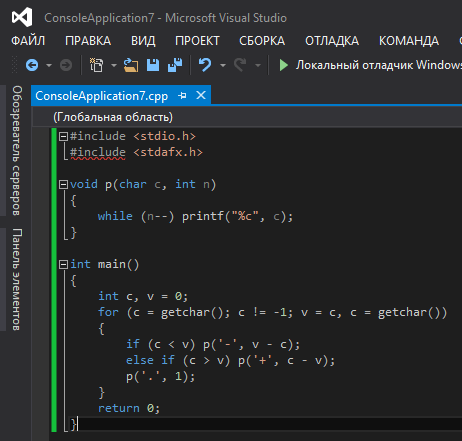
Скачиваем файл bfdev-1-4-7.exe с Яндекс.Диска (<https://yadi.sk/d/YHk1VtihG3L1Rw>)

Открываем его

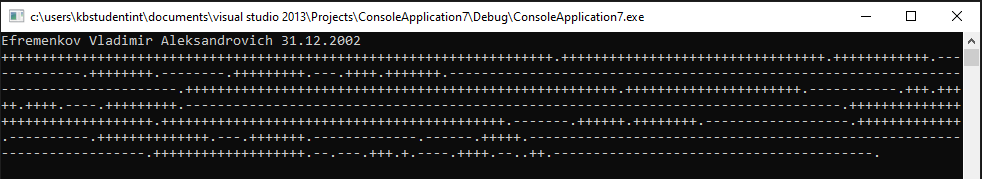


Далее я решил написать конвертер в код BrainF на C++.

Открываем C++ и пишем следующий программный код:

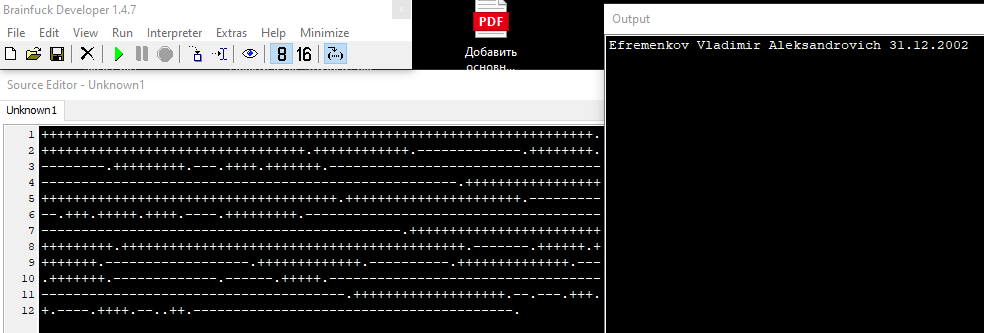


Запускаем отладчик и пишем в нем Efremenkov Vladimir Aleksandrovich 31.12.2002:



На выходе получаем код на языке BrainF.

Копируем его и вставляем в BrainF Developer, затем компилируем.

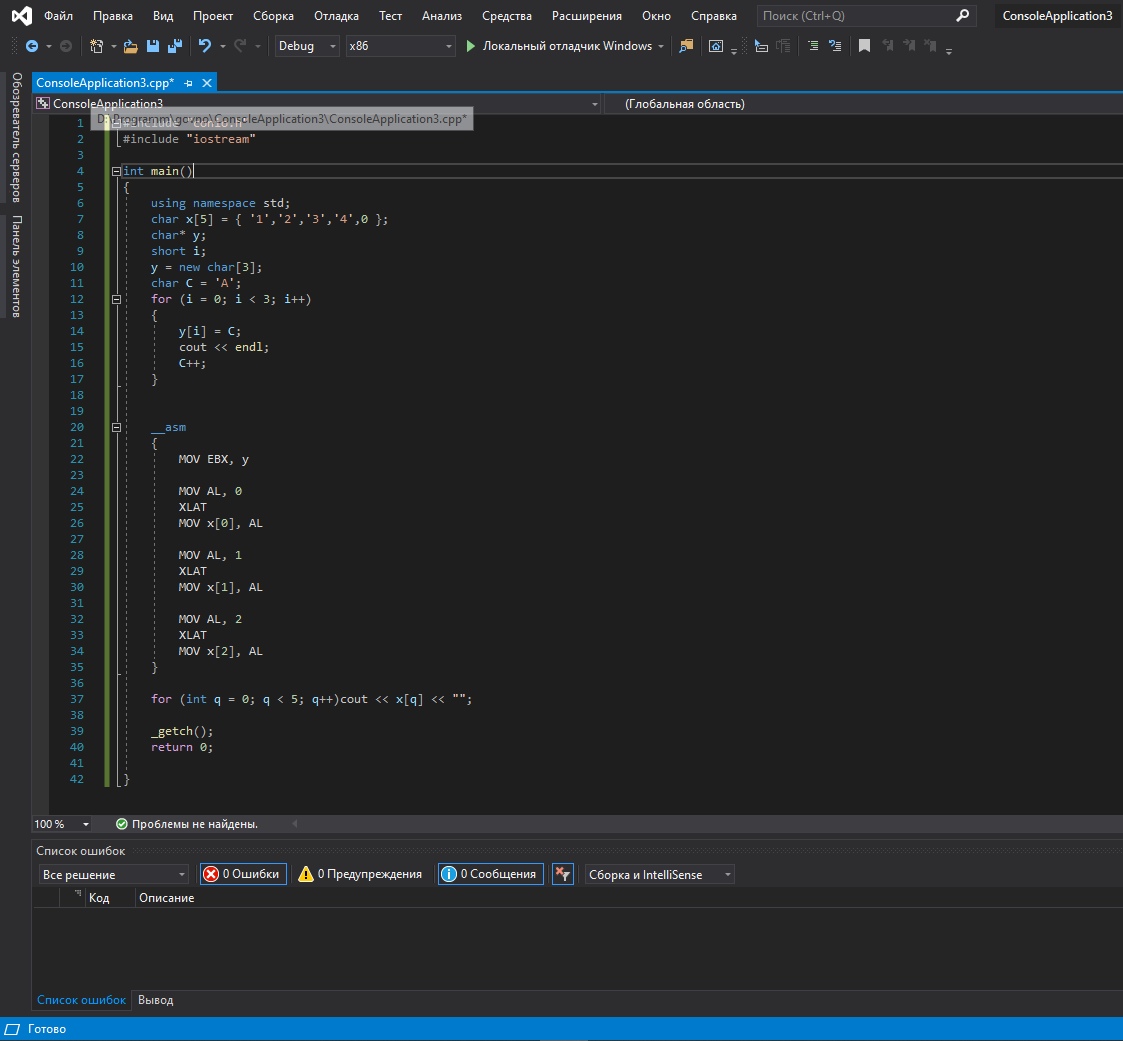


Получаем программный код на языке BrainF с ФИО и датой рождения.

**L4.1 Изучение команд пересылки данных.**

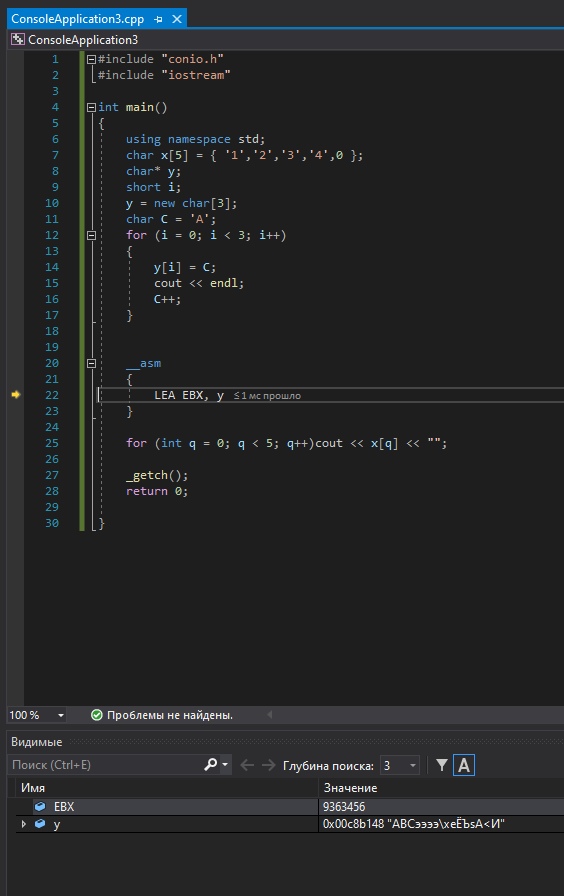
Задание 1 Обменять значения в переменных языка Паскаль x[4] и y^[3]; при выборке значения из массива y^ используйте команду XLAT.

Запускаем Visual Studio 2017 и пишем следующий программный код:

Как мы видим, значения в переменных поменялись местами, при использовании команды XLAT.

Задание 2 Определите, сколько байт требуется на запись в оперативной памяти команды LEA EBX,M, и какие числа записаны в этих байтах.

Создаем новый проект и пишем следующий программный код:

Запускаем его через F10 и видим его значение в оперативной памяти и какие числа записались в эти байты.

**L4.2 Изучение арифметических команд**

**Задание №1**

При выполнении задания необходимо обратить внимание на формат

получаемого результата, как изменяются флаги при выполнении арифмети-

ческих команд в зависимости от исходных данных. Рассмотреть отличия

команд INC и DEC от команд сложения ADD и вычитания SUB (состояние

флага C). Используя окно дизассемблера, посмотреть, во что транслируются

арифметические операции языка Паскаль.

А также были выполнены следующие требования к отчету:

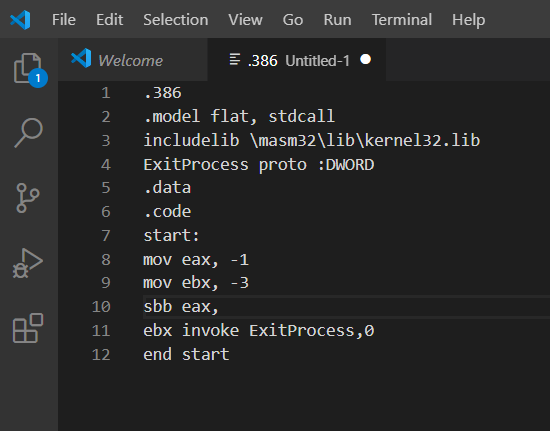
Все арифметические команды должны содержать исходные данные ря-

дом с командой.

1. Команды умножения и деления показал над десятичными числами.
2. Команды сложения и вычитания показал с такими исходными дан-

ными, чтобы изменялись флаги переполнения и переноса (на каждую

команду два примера на изменение этих флагов по отдельности).

1. Результат выполнения команды присутствует в виде комментария.

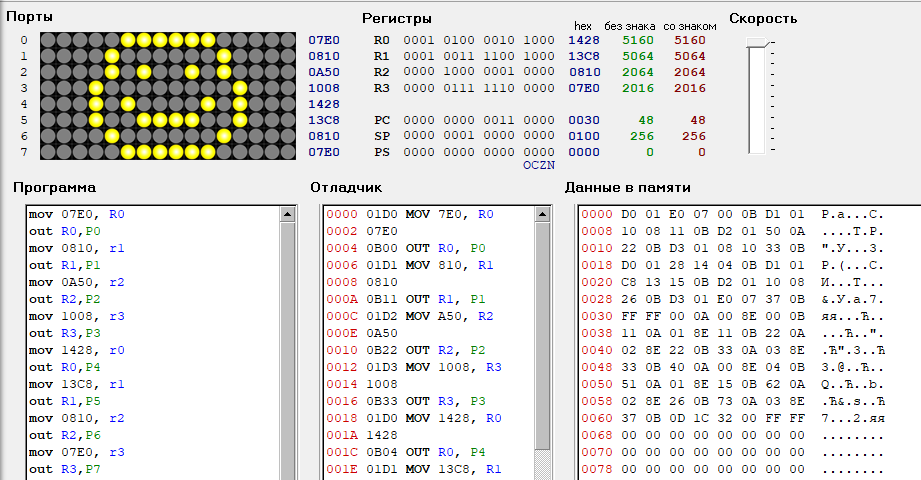
**Вывод**

Была выполнена лабораторная работа 4.2 «изучение арифметических команд». В ходе работы я усвоил и закрепил навыки написания кода в ассемблере.

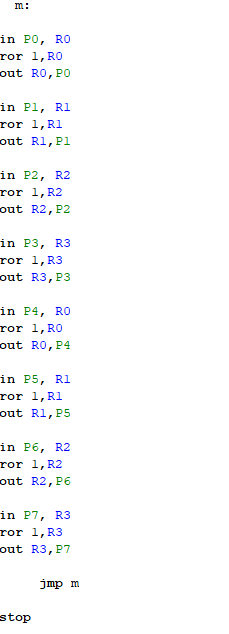
**L2 Работа с LampPanel**

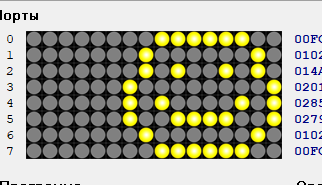
**Ход работы:**

1. Пишем код программы для загорания лампочек на ламп панели.



1. Зацикливаем наш рисунок, чтобы он перемещался с помощью цикла и команды ROR



1. Пишем цикл jmp m и запускаем программу

**L4.3 Изучение логических команд и команд сдвигов**

При выполнении задания требуется все исходные данные и результат представлять в двоичном виде. Кроме демонстрации работы логических команд требуется особо выделить и продемонстрировать: отличия команды

NOT от команды NEG, отличия команд SHR и SAR, отличия команд SUB и

TEST, отличия циклических сдвигов и циклических сдвигов через флаг C.

Работу команды SAR продемонстрируйте для положительных и отрицательных чисел.

Примерные варианты контрольных задач:

1. Реализовать циклический сдвиг влево 32-разрядного числа, оперируя

только 16-разрядными регистрами.

1. Реализовать циклический сдвиг вправо 32-разрядного числа, оперируя

только 16-разрядными регистрами.

1. Реализовать умножение на 7 (или 15, или 17, или 33), используя коман-

ды сдвигов и (только один раз) сложение или вычитание.

1. Реализовать умножение числа 2000000099 на 10.

**Задание №1**

Пишем код на

Получаем следующий вывод: