Федеральное государственное автономное образовательное учреждение

высшего образования «Национальный исследовательский университет

ИТМО»

*Факультет программной инженерии и компьютерной техники*

*Направление подготовки: 09.03.04 – Программная инженерия, Системное и прикладное программное обеспечение.*

*Дисциплина «Основы профессиональной деятельности»*

**Отчет**

**По лабораторной работе №4**

**Выполнение комплекса программ**

**Вариант №** **41004**

Выполнил:

Молчанов Фёдор Денисович

Группа: Р3113

Преподаватель:

Блохина Елена Николаевна

Г. Санкт-Петербург, 2024 г.

Оглавление

[Задание 3](#_Toc162115569)

[Текст исходной программы 4](#_Toc162115570)

[Подпрограмма 5](#_Toc162115571)

[Описание программы 6](#_Toc162115572)

[Описание и назначение исходных данных: 7](#_Toc162115573)

[Область представления 8](#_Toc162115574)

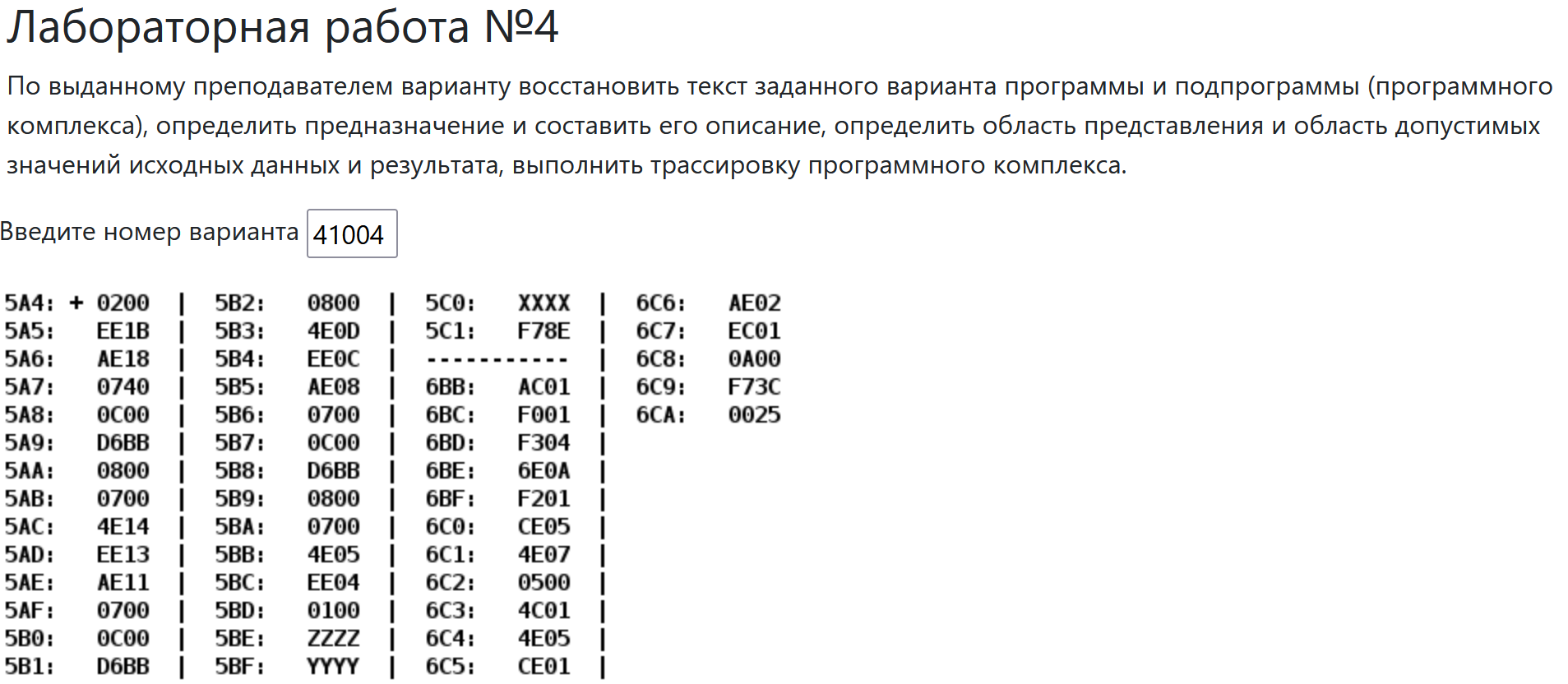
[ОДЗ переменных и результата 8](#_Toc162115575)

[Расположение в памяти БЭВМ программы, исходных данных и результатов: 8](#_Toc162115576)

[Адреса первой и последней исполняемых команд программы: 8](#_Toc162115577)

[Выводы 9](#_Toc162115578)

# Задание



# Текст исходной программы

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Код команды | Мнемоника | Комментарий |
| 0x5A4 | + 0200 | CLA | Начало программы. Очистить AC |
| 0x5A5 | EE1B | ST (IP+1B) | Прямая относительная адресация. AC -> (IP+1B =5C1) |
| 0x5A6 | AE18 | LD (IP + 18) | Прямая относительная адресация. (IP+18 = 5BF) -> AC. Загрузили Y в AC. |
| 0x5A7 | 0740 | DEC | AC – 1 |
| 0x5A8 | 0C00 | PUSH | AC -> -(SP) |
| 0x5A9 | D6BB | CALL 0x6BB | SP – 1; IP -> SP; 6BB -> IP. Вызов подпрограммы |
| 0x5AA | 0800 | POP | SP -> AC; SP + 1. Снимаем элемент с верхушки стека (там находится операнд после вычислений в подпрограмме) |
| 0x5AB | 0700 | INC | AC + 1 |
| 0x5AC | 4E14 | ADD (IP + 14) | Прямая относительная адресация. AC + (IP+14 = 5C1) |
| 0x5AD | EE13 | ST (IP + 13) | Прямая относительная адресация. AC -> (IP+13 = 5С1) |
| 0x5AE | AE11 | LD (IP + 11) | Прямая относительная адресация. (IP+11 = 5C0) -> AC  Загрузили X в AC |
| 0x5AF | 0700 | INC | AC + 1 |
| 0x5B0 | 0C00 | PUSH | AC -> -(SP) |
| 0x5B1 | D6BB | CALL 0x6BB | SP – 1; IP -> SP; 6BB -> IP |
| 0x5B2 | 0800 | POP | (SP)+ -> AC |
| 0x5B3 | 4E0D | ADD (IP + 0D) | Прямая относительная адресация. AC + (IP+0D = 5C1) |
| 0x5B4 | EE0C | ST (IP + 0C) | Прямая относительная адресация. AC -> (IP+0C= 5C1) |
| 0x5B5 | AE08 | LD (IP + 8) | Прямая относительная адресация. (IP + 8= 5BE) -> AC  Загрузили Z в AC |
| 0x5B6 | 0700 | INC | AC + 1 |
| 0x5B7 | 0C00 | PUSH | AC -> -(SP) |
| 0x5B8 | D6BB | CALL 0x6BB | SP – 1; IP -> SP; 6BB -> IP |
| 0x5B9 | 0800 | POP | (SP)+ -> AC |
| 0x5BA | 0700 | INC | AC + 1 |
| 0x5BB | 4E05 | ADD (IP + 5) | Прямая относительная адресация. AC + (IP +5 = 5C1) |
| 0x5BC | EE04 | ST (IP + 4) | Прямая относительная адресация. AC -> (IP + 4 = 5C1) |
| 0x5BD | 0100 | HLT | END |

# Подпрограмма

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Код команды | Мнемоника | Комментарий |
| 0x6BB | AC01 | LD &01 | LD 1-й эл. стека -> AC |
| 0x6BC | F001 | BEQ IP + 1 | Если равенство (Z == 1), то IP + 1 = 6BE |
| 0x6BD | F304 | BPL IP + 4 | Если плюс (N== 0), то IP + 4 = 6C2 |
| 0x6BE | 6E0A | SUB (IP + A) | Прямая относительная адресация. AC – (IP + A = 6C9) |
| 0x6BF | F201 | BMI IP + 1 | Если минус (N == 1), то IP + 1 = 6С1 |
| 0x6C0 | CE05 | JUMP (IP + 5) | Прямая относительная адресация. IP + 5= 6C6 -> IP |
| 0x6C1 | 4E07 | ADD (IP + 7) | Прямая относительная адресация. ADD (IP + 7 = 6C9) |
| 0x6C2 | 0500 | ASL | AC \* 2. Положительный X, X\*2 |
| 0x6C3 | 4C01 | ADD &1 | AC + 1-й эл. Стека, X\*2 + X |
| 0x6C4 | 4E05 | ADD (IP + 5) | Прямая относительная адресация. ADD (IP + 5 = 6CA) |
| 0x6C5 | CE01 | JUMP (IP + 1) | Прямая относительная адресация. IP + 1 = 6C7 -> IP |
| 0x6C6 | AE02 | LD (IP + 2) | Прямая относительная адресация. IP + 2 = 6C9 -> AC |
| 0x6C7 | EC01 | ST &1 | AC -> 1-й эл. Стека |
| 0x6C8 | 0A00 | RET | SP++ -> IP. Возврат |

Пусть операнд, над которыми произв. вычисления в ПП, называется X.

1 случай. x != 0  
 1.1 Если x > 0, то x = x\*2 + x + 6СA(=2510) = 3x + 3710 Загружаем это в первый элемент стека (перезаписываем x)  
 1.2 Если x < 0, то x = x – 6C9(=F73C16)  
## т.к. F73C отрицательный, то x = x – (-2244) = x + 2244  
 1.2.1 если x < 0, то x += 6C9 (возвращаем x) #x < F73C(-224410)  
 x = x\*2 + x + 6CA(=2510­) = 3x + 3710  
 Загружаем это в первый элемент стека (перезаписываем x)  
 1.2.2 если x >= 0, то x = 6C9(=F73C) #x >=F73C (-224410)  
 Загружаем это в первый элемент стека (перезаписываем x)

2 случай. Операнд == 0.  
Тогда x = x – 6C9.  
 2.1 Если 6C9 – положительный, то x += 6C9 (Возвращаем к x = 0)  
 x = 0\*2 + 0 + 6CA = 2516 = 3710  
 Загружаем это в первый элемент стека (перезаписываем x)  
 2.2 Если 6С9 – отрицательный, то x = 6C9 = F739  
 Загружаем это в первый элемент стека (перезаписываем x)

## 2.2 никогда не случится, т.к. 6C9 – положительная константа, не изменяемая в ходе программы. То есть если x = 0, то всегда 2.1.

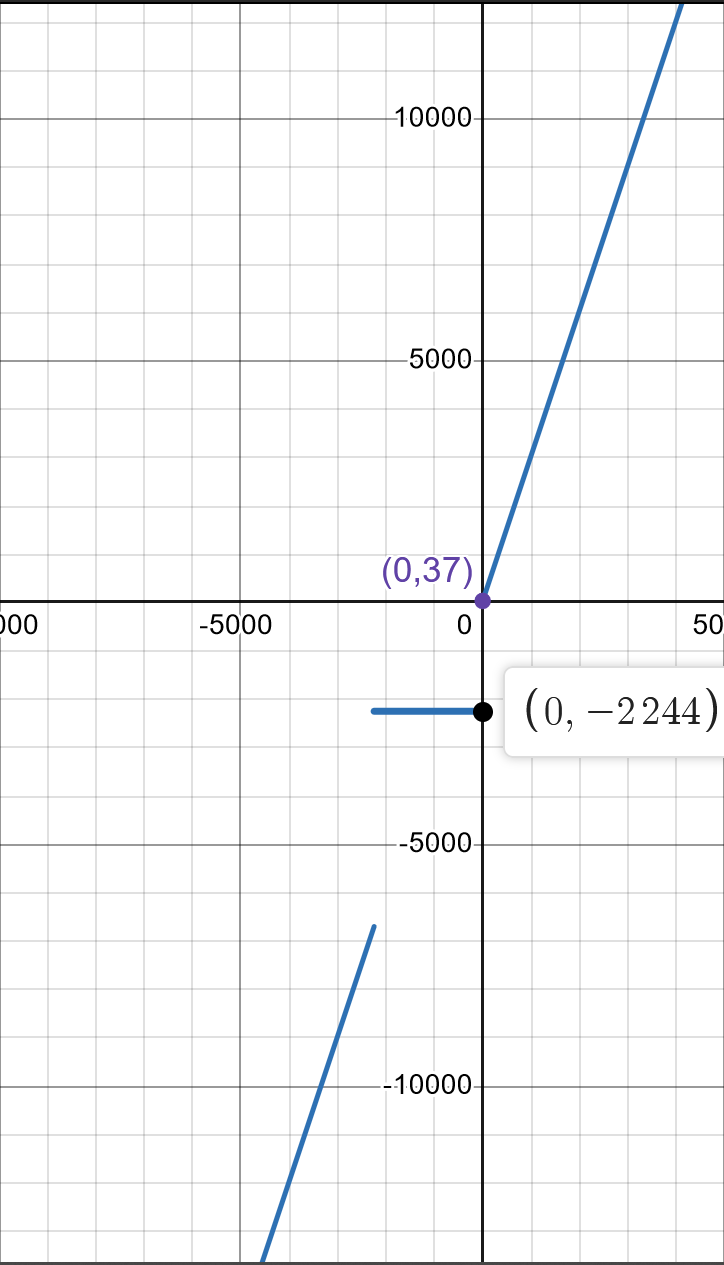
# Описание программы

1. Описание программы

Программа 3 раза выполняет вызывает подпрограмму передавая в неё Y, X, Z последовательно, между вызовами выполняет арифметические операции над R.

1. Описание подпрограммы

Подпрограмма выполняет вычисления по формуле:

****

1. Описание программного комплекса

Программный комплекс выполняет вычисления по формуле:

R = F(Y-1) + 1 + F(X + 1) + F(Z+1) + 1

# Описание и назначение исходных данных:

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Данные | Мнемоника | Комментарий |
| 0x5BE | ZZZZ | 500 (10) | z |
| 0x5BF | YYYY | -7000 (10) | y |
| 0x5C0 | XXXX | 6000 (10) | x |
| 0x5C1 | F78E | R | Результат. В начале равен 0. |
| … | | | |
| 0x6C9 | F73C | A | константа |
| 0x6CA | 0025 | B | константа |

# Область представления

R – знаковое 16-ти разрядное число:

X, Y, Z, A, B – константы – знаковые 16-ти разрядные числа,

# 

# ОДЗ переменных и результата

F(x):

1. Делим каждое значение на 3. А потом еще раз на 3, чтобы исключить случай переполнения при сумме трёх положительных чисел.
2. Рассмотрим случай, когда X <0.

Тогда самое минимальное значение, которое он может принять, это   
-32768/3 – 37 = -10959.

Значит Y, Z смогут принимать значения, при которых сумма их значений от подпрограмм не будет переполнять разрядную сетку в следующем случае:

# Расположение в памяти БЭВМ программы, исходных данных и результатов:

0x5A4 – 0x5BC0 & 0x6BB – 0x6CA: Программа

0x5A4 – 0x5BD: Исходный код

0x6BB - 0x6C8: Подпрограмма

0x5BE – 0x5C0: Константы для функции

0x5C1: Результат

0x6C9, 0x6CA: Константы для ПП

# 

# Адреса первой и последней исполняемых команд программы и подпрограммы??:

0x5A4 – первая команда

0x5BD – последняя команда

# Трассировка

R = (F(Y-1) + 1) + F(X + 1) + (F(Z+1) + 1)

|  |  |
| --- | --- |
| ZZZZ | 500 (10) = 1F416 |
| YYYY | -7000 (10) = E4A816 |
| XXXX | 6000 (10) = 177016 |

R = F(E4A7) + 1 + F(1771) + F(1F5) + 1 = AE1A + 1 + 4678 + 604 + 1 = FA98(-138410)

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Выполняемая команда | | Содержимое регистров процессора после выполнения команды | | | | | | | | Ячейка, содержимое которой изменилось после выполнения команды | |
| Адр | Код | IP | CR | AR | DR | SP | BR | AC | NZVC | Адр | Новый код |
| 5A4 | 0200 | 5A5 | 0200 | 5A4 | 0200 | 000 | 05A4 | 0000 | 0100 |  |  |
| 5A5 | EE1B | 5A6 | EE1B | 5C1 | 0000 | 000 | 001B | 0000 | 0100 | 5C1 | 0000 |
| 5A6 | AE18 | 5A7 | AE18 | 5BF | E4A8 | 000 | 0018 | E4A8 | 1000 |  |  |
| 5A7 | 0740 | 5A8 | 0740 | 5A7 | 0740 | 000 | 05A7 | E4A7 | 1001 |  |  |
| 5A8 | 0C00 | 5A9 | 0C00 | 7FF | E4A7 | 7FF | 05A8 | E4A7 | 1001 | 7FF | E4A7 |
| 5A9 | D6BB | 6BB | D6BB | 7FE | 05AA | 7FE | D6BB | E4A7 | 1001 | 7FE | 05AA |
| 6BB | AC01 | 6BC | AC01 | 7FF | E4A7 | 7FE | 0001 | E4A7 | 1001 |  |  |
| 6BC | F001 | 6BD | F001 | 6BC | F001 | 7FE | 06BC | E4A7 | 1001 |  |  |
| 6BD | F304 | 6BE | F304 | 6BD | F304 | 7FE | 06BD | E4A7 | 1001 |  |  |
| 6BE | 6E0A | 6BF | 6E0A | 6C9 | F73C | 7FE | 000A | ED6B | 1000 |  |  |
| 6BF | F201 | 6C1 | F201 | 6BF | F201 | 7FE | 0001 | ED6B | 1000 |  |  |
| 6C1 | 4E07 | 6C2 | 4E07 | 6C9 | F73C | 7FE | 0007 | E4A7 | 1001 |  |  |
| 6C2 | 0500 | 6C3 | 0500 | 6C2 | E4A7 | 7FE | 06C2 | C94E | 1001 |  |  |
| 6C3 | 4C01 | 6C4 | 4C01 | 7FF | E4A7 | 7FE | 0001 | ADF5 | 1001 |  |  |
| 6C4 | 4E05 | 6C5 | 4E05 | 6CA | 0025 | 7FE | 0005 | AE1A | 1000 |  |  |
| 6C5 | CE01 | 6C7 | CE01 | 6C5 | 06C7 | 7FE | 0001 | AE1A | 1000 |  |  |
| 6C7 | EC01 | 6C8 | EC01 | 7FF | AE1A | 7FE | 0001 | AE1A | 1000 | 7FF | AE1A |
| 6C8 | 0A00 | 5AA | 0A00 | 7FE | 05AA | 7FF | 06C8 | AE1A | 1000 |  |  |
| 5AA | 0800 | 5AB | 0800 | 7FF | AE1A | 000 | 05AA | AE1A | 1000 |  |  |
| 5AB | 0700 | 5AC | 0700 | 5AB | 0700 | 000 | 05AB | AE1B | 1000 |  |  |
| 5AC | 4E14 | 5AD | 4E14 | 5C1 | 0000 | 000 | 0014 | AE1B | 1000 |  |  |
| 5AD | EE13 | 5AE | EE13 | 5C1 | AE1B | 000 | 0013 | AE1B | 1000 | 5C1 | AE1B |
| 5AE | AE11 | 5AF | AE11 | 5C0 | 1770 | 000 | 0011 | 1770 | 0000 |  |  |
| 5AF | 0700 | 5B0 | 0700 | 5AF | 0700 | 000 | 05AF | 1771 | 0000 |  |  |
| 5B0 | 0C00 | 5B1 | 0C00 | 7FF | 1771 | 7FF | 05B0 | 1771 | 0000 | 7FF | 1771 |
| 5B1 | D6BB | 6BB | D6BB | 7FE | 05B2 | 7FE | D6BB | 1771 | 0000 | 7FE | 05B2 |
| 6BB | AC01 | 6BC | AC01 | 7FF | 1771 | 7FE | 0001 | 1771 | 0000 |  |  |
| 6BC | F001 | 6BD | F001 | 6BC | F001 | 7FE | 06BC | 1771 | 0000 |  |  |
| 6BD | F304 | 6C2 | F304 | 6BD | F304 | 7FE | 0004 | 1771 | 0000 |  |  |
| 6C2 | 0500 | 6C3 | 0500 | 6C2 | 1771 | 7FE | 06C2 | 2EE2 | 0000 |  |  |
| 6C3 | 4C01 | 6C4 | 4C01 | 7FF | 1771 | 7FE | 0001 | 4653 | 0000 |  |  |
| 6C4 | 4E05 | 6C5 | 4E05 | 6CA | 0025 | 7FE | 0005 | 4678 | 0000 |  |  |
| 6C5 | CE01 | 6C7 | CE01 | 6C5 | 06C7 | 7FE | 0001 | 4678 | 0000 |  |  |
| 6C7 | EC01 | 6C8 | EC01 | 7FF | 4678 | 7FE | 0001 | 4678 | 0000 | 7FF | 4678 |
| 6C8 | 0A00 | 5B2 | 0A00 | 7FE | 05B2 | 7FF | 06C8 | 4678 | 0000 |  |  |
| 5B2 | 0800 | 5B3 | 0800 | 7FF | 4678 | 000 | 05B2 | 4678 | 0000 |  |  |
| 5B3 | 4E0D | 5B4 | 4E0D | 5C1 | AE1B | 000 | 000D | F493 | 1000 |  |  |
| 5B4 | EE0C | 5B5 | EE0C | 5C1 | F493 | 000 | 000C | F493 | 1000 | 5C1 | F493 |
| 5B5 | AE08 | 5B6 | AE08 | 5BE | 01F4 | 000 | 0008 | 01F4 | 0000 |  |  |
| 5B6 | 0700 | 5B7 | 0700 | 5B6 | 0700 | 000 | 05B6 | 01F5 | 0000 |  |  |
| 5B7 | 0C00 | 5B8 | 0C00 | 7FF | 01F5 | 7FF | 05B7 | 01F5 | 0000 | 7FF | 01F5 |
| 5B8 | D6BB | 6BB | D6BB | 7FE | 05B9 | 7FE | D6BB | 01F5 | 0000 | 7FE | 05B9 |
| 6BB | AC01 | 6BC | AC01 | 7FF | 01F5 | 7FE | 0001 | 01F5 | 0000 |  |  |
| 6BC | F001 | 6BD | F001 | 6BC | F001 | 7FE | 06BC | 01F5 | 0000 |  |  |
| 6BD | F304 | 6C2 | F304 | 6BD | F304 | 7FE | 0004 | 01F5 | 0000 |  |  |
| 6C2 | 0500 | 6C3 | 0500 | 6C2 | 01F5 | 7FE | 06C2 | 03EA | 0000 |  |  |
| 6C3 | 4C01 | 6C4 | 4C01 | 7FF | 01F5 | 7FE | 0001 | 05DF | 0000 |  |  |
| 6C4 | 4E05 | 6C5 | 4E05 | 6CA | 0025 | 7FE | 0005 | 0604 | 0000 |  |  |
| 6C5 | CE01 | 6C7 | CE01 | 6C5 | 06C7 | 7FE | 0001 | 0604 | 0000 |  |  |
| 6C7 | EC01 | 6C8 | EC01 | 7FF | 0604 | 7FE | 0001 | 0604 | 0000 | 7FF | 0604 |
| 6C8 | 0A00 | 5B9 | 0A00 | 7FE | 05B9 | 7FF | 06C8 | 0604 | 0000 |  |  |
| 5B9 | 0800 | 5BA | 0800 | 7FF | 0604 | 000 | 05B9 | 0604 | 0000 |  |  |
| 5BA | 0700 | 5BB | 0700 | 5BA | 0700 | 000 | 05BA | 0605 | 0000 |  |  |
| 5BB | 4E05 | 5BC | 4E05 | 5C1 | F493 | 000 | 0005 | FA98 | 1000 |  |  |
| 5BC | EE04 | 5BD | EE04 | 5C1 | FA98 | 000 | 0004 | FA98 | 1000 | 5C1 | FA98 |
| 5BD | 0100 | 5BE | 0100 | 5BD | 0100 | 000 | 05BD | FA98 | 1000 |  |  |

# Выводы

В ходе работы я узнал про такую структуру данных, как стек и разобрался в его действии в БЭВМ. Узнал про работу команд PUSH, POP и в комплексе программ, использующих стек при работе.