Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

**Отчёт**

**По лабораторной работе №4**

**«Выполнение комплекса программ»**

по дисциплине «Основы профессиональной деятельности»

**Вариант: 11003**

Работу выполнил:

Поленов Кирилл Александрович

Группа Р3113

Работу приняла:

Ткешелашвили Нино Мерабиевна

Оглавление

[Задание 3](#_Toc159367232)

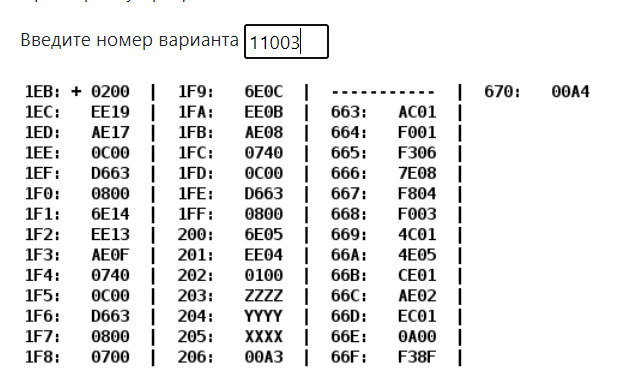
[Определение функции, вычисляемой программой 3](#_Toc159367233)

[ОП и ОДЗ исходных данных и результата 6](#_Toc159367234)

[Трассировка программы 7](#_Toc159367235)

[Выводы 8](#_Toc159367236)

# Задание



# Определение функции, вычисляемой программой

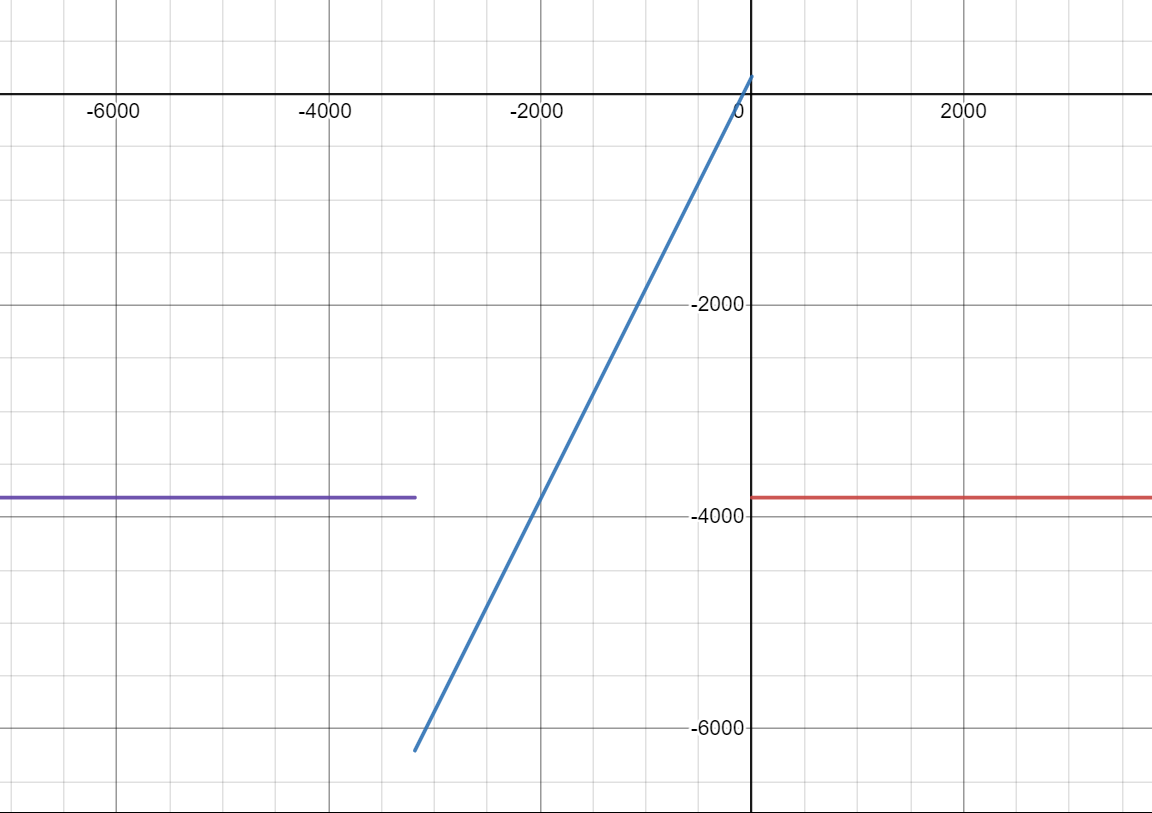
**Описание программы**

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Содержимое | Мнемоника | Описание |
| 1EB | 0200 | CLA | Очистка аккумулятора |
| 1EC | EE19 | ST (IP+19) | AC -> MEM(IP+19 = 206) Загрузили 0000 в ячейку 206 (обнулили R) |
| 1ED | AE17 | LD (IP+17) | MEM(IP+17 = 205) -> AC  Загрузили число X в аккумулятор |
| 1EE | 0C00 | PUSH | AC -> -(SP)  Положили содержимое AC на вершину стека |
| 1EF | D663 | CALL 663 | SP-1 -> SP  IP -> SP  663 -> IP  Вызов подпрограммы |
| 1F0 | 0800 | POP | (SP)+ -> AC  Взяли результат со стека |
| 1F1 | 6E14 | SUB (IP+14) | AC – MEM(IP+14=205) -> AC  F(X) - R |
| 1F2 | EE13 | ST (IP+13) | AC + MEM(IP+13=205) -> AC |
| 1F3 | AE0F | LD (IP+15) | MEM(IP+15=203) -> AC  Закинули Z |
| 1F4 | 0740 | DEC | AC-1 -> AC  Z-1 |
| 1F5 | 0C00 | PUSH | AC -> -(SP)  Положили содержимое AC на вершину стека |
| 1F6 | D663 | CALL 663 | SP-1 -> SP  IP -> SP  663 -> IP  Вызов подпрограммы |
| 1F7 | 0800 | POP | (SP)+ -> AC  Взяли результат со стека |
| 1F8 | 0700 | DEC | AC-1 -> AC  F(Z-1)-1 |
| 1F9 | 6E0C | SUB (IP+12) | AC - MEM(IP+12=206) - >  F(Z-1)-1 – F(X) |
| 1FA | EE0B | ST (IP+11) | AC -> MEM(IP+11=206) |
| 1FB | AE08 | LD (IP+8) | MEM(IP+8=204) -> AC  Взяли Y |
| 1FC | 0740 | DEC | AC-1 -> AC  Y-1 |
| 1FD | 0C00 | PUSH | AC -> -(SP)  Положили содержимое AC на вершину стека |
| 1FE | D663 | CALL 663 | SP-1 -> SP  IP -> SP  663 -> IP  Вызов подпрограммы |
| 1FF | 0800 | POP | (SP)+ -> AC  Взяли результат со стека |
| 200 | 6E05 | SUB (IP+5) | AC-MEM(IP+5=206) -> AC  F(Z-1)-1 – F(X) – F(Y-1) |
| 201 | EE04 | ST (IP+4) | AC -> MEM(IP+4=206) |
| 202 | 0100 | HLT | Остановить выполнение программы |
| 203 | ZZZZ | Z | Число |
| 204 | YYYY | Y | Число |
| 205 | XXXX | X | Число |
| 206 | 00A3 | R | Число |
| Подпрограмма | | | |
| 663 | AC01 | LD (SP+1) | MEM(SP+1 = 7FF) -> AC  Загрузили число NUM из стека |
| 664 | F001 | BEQ (IP+1) | Переход в (IP+1 = 666), если Z==1 (NUM == 0) |
| 665 | F306 | BPL (IP+6) | Переход в (IP+6 = 66C), если N==0 (NUM >= 0) |
| 666 | 7E08 | CMP (IP+8) | AC – MEM(IP+8 = 66F) -> N, Z, V, C  Сравниваем с F38F = -3185 |
| 667 | F804 | BLT (IP+4) | Переход в (IP+4 = 66C), если N != V (NUM < -3185­­) |
| 668 | F003 | BEQ(IP+3) | Переход в (IP+3= 66C), если Z==1 (NUM == 0) |
| 669 | 4C01 | ADD (SP+1) | AC + MEM(SP+1) -> AC  NUM + NUM = 2\*NUM |
| 66A | 4E05 | ADD (IP+5) | AC + MEM(IP+5=670) -> AC  2\*NUM + 164 |
| 66B | CE01 | JUMP (IP+1) | Переход в (IP+1= 66E) |
| 66C | AE02 | LD (IP+2) | MEM(IP+2 = 66F) ->  Return -3185 |
| 66D | EC01 | ST (SP+1) | AC -> MEM(SP+1)  Положили преобразованный NUM на вершину стека |
| 66E | 0A00 | RET | (SP)+ -> IP  Выход из подпрограммы |
| 66F | F38F | A | Константа для сравнения |
| 670 | 00A4 | B | Константа для вычитания |

*Таблица 1*

Программа выполняет следующую функцию:

R = F(Y-1) - F (Z-1)-1 – F(X), где



# ОП и ОДЗ исходных данных и результата

**Область представления:**

* X, Y, Z, R, A, B – 16-ричные знаковые числа

**Область определения:**

A = F38F16 = -318510 (константа)

B = 00A416 = 16410 (константа)

Основная программа вычисляет выражение

R = F(Y-1) - F (Z-1)-1 – F(X)

При значении n в промежутке функция вернёт -3185. То есть при вводе любого значения из этого промежутка не произойдет переполнения.

В остальных случаях, а именно , она вернёт . Найдем ОДЗ для аргумента функции в этом случае:

Min R = -6204 – 1 – 164 – 164 = -6533

Max R = 164 – 1 – (-6204) – (-6204) = 12571

В обоих случаях переполнения нету.

Итого:

# Трассировка программы

**Таблица трассировки**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Выполняемая команда | | Содержимое регистров после выполнения команды | | | | | | | | | Ячейка, содержимое которой изменилось после выполнения команды | | |
| Адрес | Содержимое | IP | CR | AR | DR | SP | BR | AC | PS | NZVC | Адрес | Содержимое |
| xxx | xxxx | xxx | xxxx | xxx | xxxx | xxx | xxxx | xxxx | xxx | xxxx | xxx | xxxx |
| 1EB | 0200 | 1EB | 0000 | 000 | 0000 | 000 | 0000 | 0000 | 004 | 0100 |  |  |
| 1EB | 0200 | 1EC | 0200 | 1EB | 0200 | 000 | 01EB | 0000 | 004 | 0100 |  |  |
| 1EC | EE19 | 1ED | EE19 | 206 | 0000 | 000 | 0019 | 0000 | 004 | 0100 | 206 | 0000 |
| 1ED | AE17 | 1EE | AE17 | 205 | F380 | 000 | 0017 | F380 | 008 | 1000 |  |  |
| 1EE | 0C00 | 1EF | 0C00 | 7FF | F380 | 7FF | 01EE | F380 | 008 | 1000 | 7FF | F380 |
| 1EF | D663 | 663 | D663 | 7FE | 01F0 | 7FE | D663 | F380 | 008 | 1000 | 7FE | 01F0 |
| 663 | AC01 | 664 | AC01 | 7FF | F380 | 7FE | 0001 | F380 | 008 | 1000 |  |  |
| 664 | F001 | 665 | F001 | 664 | F001 | 7FE | 0664 | F380 | 008 | 1000 |  |  |
| 665 | F306 | 666 | F306 | 665 | F306 | 7FE | 0665 | F380 | 008 | 1000 |  |  |
| 666 | 7E08 | 667 | 7E08 | 66F | F38F | 7FE | 0008 | F380 | 008 | 1000 |  |  |
| 667 | F804 | 66C | F804 | 667 | F804 | 7FE | 0004 | F380 | 008 | 1000 |  |  |
| 66C | AE02 | 66D | AE02 | 66F | F38F | 7FE | 0002 | F38F | 008 | 1000 |  |  |
| 66D | EC01 | 66E | EC01 | 7FF | F38F | 7FE | 0001 | F38F | 008 | 1000 | 7FF | F38F |
| 66E | 0A00 | 1F0 | 0A00 | 7FE | 01F0 | 7FF | 066E | F38F | 008 | 1000 |  |  |
| 1F0 | 0800 | 1F1 | 0800 | 7FF | F38F | 000 | 01F0 | F38F | 008 | 1000 |  |  |
| 1F1 | 6E14 | 1F2 | 6E14 | 206 | 0000 | 000 | 0014 | F38F | 009 | 1001 |  |  |
| 1F2 | EE13 | 1F3 | EE13 | 206 | F38F | 000 | 0013 | F38F | 009 | 1001 | 206 | F38F |
| 1F3 | AE0F | 1F4 | AE0F | 203 | 0016 | 000 | 000F | 0016 | 001 | 0001 |  |  |
| 1F4 | 0740 | 1F5 | 0740 | 1F4 | 0740 | 000 | 01F4 | 0015 | 001 | 0001 |  |  |
| 1F5 | 0C00 | 1F6 | 0C00 | 7FF | 0015 | 7FF | 01F5 | 0015 | 001 | 0001 | 7FF | 0015 |
| 1F6 | D663 | 663 | D663 | 7FE | 01F7 | 7FE | D663 | 0015 | 001 | 0001 | 7FE | 01F7 |
| 663 | AC01 | 664 | AC01 | 7FF | 15 | 7FE | 0001 | 0015 | 001 | 0001 |  |  |
| 664 | F001 | 665 | F001 | 664 | F001 | 7FE | 0664 | 0015 | 001 | 0001 |  |  |
| 665 | F306 | 66C | F306 | 665 | F306 | 7FE | 0006 | 0015 | 001 | 0001 |  |  |
| 66C | AE02 | 66D | AE02 | 66F | F38F | 7FE | 0002 | F38F | 009 | 1001 |  |  |
| 66D | EC01 | 66E | EC01 | 7FF | F38F | 7FE | 0001 | F38F | 009 | 1001 | 7FF | F38F |
| 66E | 0A00 | 1F7 | 0A00 | 7FE | 01F7 | 7FF | 066E | F38F | 009 | 1001 |  |  |
| 1F7 | 0800 | 1F8 | 0800 | 7FF | F38F | 000 | 01F7 | F38F | 009 | 1001 |  |  |
| 1F8 | 0740 | 1F9 | 0740 | 1F8 | 0740 | 000 | 01F8 | F38E | 009 | 1001 |  |  |
| 1F9 | 6E0C | 1FA | 6E0C | 206 | F38F | 000 | 000C | FFFF | 008 | 1000 |  |  |
| 1FA | EE0B | 1FB | EE0B | 206 | FFFF | 000 | 000B | FFFF | 008 | 1000 | 206 | FFFF |
| 1FB | AE08 | 1FC | AE08 | 204 | FFAE | 000 | 0008 | FFAE | 008 | 1000 |  |  |
| 1FC | 0740 | 1FD | 0740 | 1FC | 0740 | 000 | 01FC | FFAD | 009 | 1001 |  |  |
| 1FD | 0C00 | 1FE | 0C00 | 7FF | FFAD | 7FF | 01FD | FFAD | 009 | 1001 | 7FF | FFAD |
| 1FE | D663 | 663 | D663 | 7FE | 01FF | 7FE | D663 | FFAD | 009 | 1001 | 7FE | 01FF |
| 663 | AC01 | 664 | AC01 | 7FF | FFAD | 7FE | 0001 | FFAD | 009 | 1001 |  |  |
| 664 | F001 | 665 | F001 | 664 | F001 | 7FE | 0664 | FFAD | 009 | 1001 |  |  |
| 665 | F306 | 666 | F306 | 665 | F306 | 7FE | 0665 | FFAD | 009 | 1001 |  |  |
| 666 | 7E08 | 667 | 7E08 | 66F | F38F | 7FE | 0008 | FFAD | 001 | 0001 |  |  |
| 667 | F804 | 668 | F804 | 667 | F804 | 7FE | 0667 | FFAD | 001 | 0001 |  |  |
| 668 | F003 | 669 | F003 | 668 | F003 | 7FE | 0668 | FFAD | 001 | 0001 |  |  |
| 669 | 4C01 | 66A | 4C01 | 7FF | FFAD | 7FE | 0001 | FF5A | 009 | 1001 |  |  |
| 66A | 4E05 | 66B | 4E05 | 670 | 00A4 | 7FE | 0005 | FFFE | 008 | 1000 |  |  |
| 66B | CE01 | 66D | CE01 | 66B | 066D | 7FE | 0001 | FFFE | 008 | 1000 |  |  |
| 66D | EC01 | 66E | EC01 | 7FF | FFFE | 7FE | 0001 | FFFE | 008 | 1000 | 7FF | FFFE |
| 66E | 0A00 | 1FF | 0A00 | 7FE | 01FF | 7FF | 066E | FFFE | 008 | 1000 |  |  |
| 1FF | 0800 | 200 | 0800 | 7FF | FFFE | 000 | 01FF | FFFE | 008 | 1000 |  |  |
| 200 | 6E05 | 201 | 6E05 | 206 | FFFF | 000 | 0005 | FFFF | 008 | 1000 |  |  |
| 201 | EE04 | 202 | EE04 | 206 | FFFF | 000 | 0004 | FFFF | 008 | 1000 | 206 | FFFF |
| 202 | 0100 | 203 | 0100 | 202 | 0100 | 000 | 0202 | FFFF | 008 | 1000 |  |  |

*Таблица 2*

# Выводы

В ходе данной лабораторной работы я:

- Познакомился с реализацией подпрограмм в БЭВМ

- Познакомился с такой структурой данных, как стек

- Закрепил знания о режимах адресации в БЭВМ