

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ
ТЕХНИКИ

НАПРАВЛЕНИЕ ПРОГРАММНАЯ ИНЖЕНЕРИЯ
ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА СИСТЕМНОЕ И ПРИКЛАДНОЕ
ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 4
курса «Основы профессиональной деятельности»
по теме: «Выполнение комплексных программ»
Вариант № 9409

Выполнил студент:

Шубин Егор Вячеславович

группа: Р3109

Преподаватель:

Лектор: Клименков С. В.,

Практик: Ткешелашвили Н. М.

Санкт-Петербург, 2025 г.

Содержание

Лабораторная работа № 4. Выполнение комплексных программ	3
1. Задание варианта № 9409	3
2. Выполнение задания	4
1. Текст исходной программы:	4
2. Описание программы:	6
3. Таблица трассировки:	8
4. Получение новых чисел	9
3. Вывод	10

Лабораторная работа № 4

Выполнение комплексных программ

1. Задание варианта № 9409

Лабораторная работа №4

По выданному преподавателем варианту восстановить текст заданного варианта программы и подпрограммы (программного комплекса), определить предназначение и составить его описание, определить область представления и область допустимых значений исходных данных и результата, выполнить трассировку программного комплекса.

Введите номер варианта

5AB: + 0200	5B9: 0800	5C7: YYYY	68E: 0EE2
5AC: EE1C	5BA: 0740	5C8: XXXX	68F: 0064
5AD: AE1A	5BB: 6E0D	5C9: 0EE1	
5AE: 0700	5BC: EE0C	-----	
5AF: 0C00	5BD: AE09	684: AC01	
5B0: D684	5BE: 0700	685: F205	
5B1: 0800	5BF: 0C00	686: 7E07	
5B2: 0740	5C0: D684	687: F903	
5B3: 4E15	5C1: 0800	688: 0500	
5B4: EE14	5C2: 0740	689: 4E05	
5B5: AE10	5C3: 4E05	68A: CE01	
5B6: 0740	5C4: EE04	68B: AE02	
5B7: 0C00	5C5: 0100	68C: EC01	
5B8: D684	5C6: ZZZZ	68D: 0A00	

Рис. 1.1: Задание

2. Выполнение задания

2. 1. Текст исходной программы:

Адрес	Код команды	Мнемоника	Комментарии
5AB	0200	CLA	Очистить аккумулятор
5AC	EE1C	ST IP+1D(5C9)	Прямая относительная выгрузка AC -> R
5AD	AE1A	LD IP+1B(5C8)	Прямая относительная загрузка X -> AC
5AE	0700	INC	Инкремент аккумулятора ++AC -> AC
5AF	0C00	PUSH	Положить аккумулятор в стек AC -> -(SP)
5B0	D684	CALL(684)	Вызов функции F(X+1) SP-1 -> SP, IP -> (SP), 684 -> IP
5B1	0800	POP	Снять со стека +(SP) -> AC
5B2	0740	DEC	Декремент аккумулятора --AC -> AC
5B3	4E15	ADD IP+16(5C9)	Сложить аккумулятор с ячейкой R AC+MEM(R) -> AC
5B4	EE14	ST IP+15(5C9)	Прямая относительная выгрузка AC -> R
5B5	AE10	LD IP+11(5C6)	Прямая относительная загрузка Z -> AC
5B6	0740	DEC	Декремент аккумулятора --AC -> AC
5B7	0C00	PUSH	Положить аккумулятор в стек AC -> -(SP)
5B8	D684	CALL(684)	Вызов функции F(Z-1) SP-1 -> SP, IP -> (SP), 684 -> IP
5B9	0800	POP	Снять со стека +(SP) -> AC
5BA	0740	DEC	Декремент аккумулятора --AC -> AC
5BB	6E0D	SUB IP+E(5C9)	Вычесть значение R из аккумулятора AC-R -> AC

Таблица 1.1: Текст исходной программы

Адрес	Код ко-манды	Мнемоника	Комментарии
5BC	EE0C	ST IP+D(5C9)	Прямая относительная выгрузка AC -> R
5BD	AE09	LD IP+A(5C7)	Прямая относительная загрузка Y -> AC
5BE	0700	INC	Инкремент аккумулятора ++AC -> AC
5BF	0C00	PUSH	Положить аккумулятор в стек AC -> -(SP)
5C0	D684	CALL(684)	Вызов функции F(Y+1) SP-1 -> SP, IP -> (SP), 684 -> IP
5C1	0800	POP	Снять со стека +(SP) -> AC
5C2	0740	DEC	Декремент аккумулятора -AC -> AC
5C3	4E05	ADD IP+6(5C9)	Сложить аккумулятор с ячейкой R AC+MEM(R) -> AC
5C4	EE04	ST IP+5(5C9)	Прямая относительная выгрузка AC -> R
5C5	0100	HLT	Остановка программы
5C6	ZZZZ	-	Значение Z
5C7	YYYY	-	Значение Y
5C8	XXXX	-	Значение X
5C9	0E01	-	Результат R

Таблица 1.2: Текст исходной программы

Адрес	Код ко-манды	Мнемоника	Комментарии
684	AC01	LD (SP+1)	Загрузка аргумента из стека MEM(SP+1) -> AC
685	F205	BNS IP+6(68B)	Если ($N == 1$) IP+6 -> IP
686	7E07	CMP IP+8(68E)	Установить флаги по результату AC - A
687	F903	BGE IP+4(68B)	Если ($N == V \vee N \oplus V == 0$) IP+4 -> IP
688	0500	ASL	Сдвиг аккумулятора влево AC*2 -> AC
689	4E05	ADD IP+6(68F)	Сложить аккумулятор с ячейкой B AC+B -> AC
68A	CE01	JUMP IP+2(68C)	Прямой относительный прыжок IP+2 -> IP
68B	AE02	LD IP+3(68E)	Прямая относительная загрузка A -> AC
68C	EC01	ST (SP+1)	Прямая относительная выгрузка AC -> (SP+1)
68D	0A00	RET	Возврат (SP)+ -> IP
68E	0EE2	-	Значение A
68F	0064	-	Значение B

Таблица 1.3: Текст исходной подпрограммы

2. 2. Описание программы:

- Предназначение программы:

Вычисление значения функции:

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

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

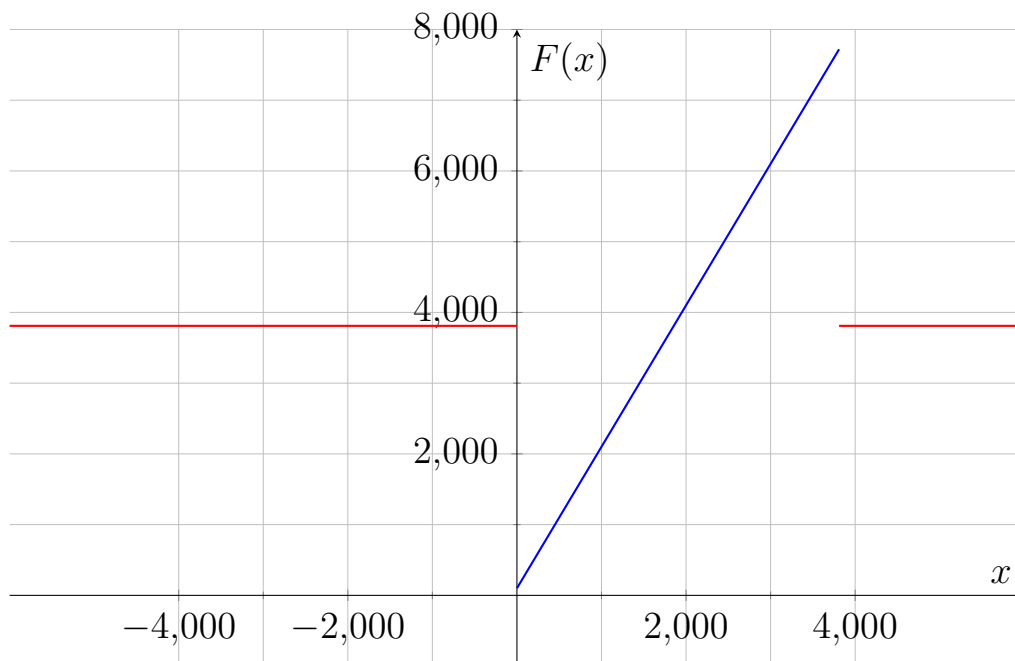
$$R = F(Z - 1) - F(X + 1) + F(Y + 1) - 1$$

$$A = EE2_{16} = 3810_{10}$$

$$B = 64_{16} = 100_{10}$$

$$F(x) = \begin{cases} x \in (-\infty; 0) \vee [A; +\infty); x = A \\ x \in [0; A); x = x * 2 + B \end{cases}$$

График:



- Область представления программы:
X, Y, Z, A, B - Знаковые 15-разрядные числа.
- Область допустимых значений программы:
 $A = EE2_{16} = 3810_{10}$
 $B = 64_{16} = 100_{10}$

На промежутке $(-\infty; 0) \cup [3810; +\infty)$ функция принимает значение 3810. Переполнения не возникает.

На промежутке $[0; 3810)$ функция принимает значение $2x+100$.

Минимальное значение: $f(0) = 100$

Максимальное значение: $f(3809) = 7718$.

На всем промежутке значений аргумента, функция принимает значение на промежутке $[0; 7718]$. Переполнения не возникает.

Минимальное значение, которое может принимать результат:

$$R = F(Z-1) - F(X+1) + F(Y+1) - 1 = 100 - 7718 + 100 - 1 = -7518$$

Максимальное значение, которое может принимать результат:

$$R = F(Z-1) - F(X+1) + F(Y+1) - 1 = 7718 - 100 + 7718 - 1 = 15335$$

В обоих случаях переполнения не возникает.

Значит область допустимых значений следующая:

$$Z \in [-2^{15} + 1; 2^{15} - 1]$$

$$X, Y \in [-2^{15}; 2^{15} - 2]$$

$$R \in [-7515; 15333], \text{ при } A = 3810 \text{ и } B = 100$$

2. 3. Таблица трассировки:

Адр	Знач	IP	CR	AR	DR	SP	BR	AC	NZVC	Адр	Знач
xxx	xxxx	xxx	xxxx	xxx	xxxx	xxx	xxxx	xxxx	xxxx	xxx	xxxx
5AB	0200	5AB	0000	000	0000	000	0000	0000	0100		
5AB	0200	5AC	0200	5AB	0200	000	05AB	0000	0100		
5AC	EE1C	5AD	EE1C	5C9	0000	000	001C	0000	0100	5C9	0000
5AD	AE1A	5AE	AE1A	5C8	3901	000	001A	3901	0000		
5AE	0700	5AF	0700	5AE	0700	000	05AE	3902	0000		
5AF	0C00	5B0	0C00	7FF	3902	7FF	05AF	3902	0000	7FF	3902
5B0	D684	684	D684	7FE	05B1	7FE	D684	3902	0000	7FE	05B1
684	AC01	685	AC01	7FF	3902	7FE	0001	3902	0000		
685	F205	686	F205	685	F205	7FE	0685	3902	0000		
686	7E07	687	7E07	68E	0EE2	7FE	0007	3902	0001		
687	F903	68B	F903	687	F903	7FE	0003	3902	0001		
68B	AE02	68C	AE02	68E	0EE2	7FE	0002	0EE2	0001		
68C	EC01	68D	EC01	7FF	0EE2	7FE	0001	0EE2	0001	7FF	0EE2
68D	0A00	5B1	0A00	7FE	05B1	7FF	068D	0EE2	0001		
5B1	0800	5B2	0800	7FF	0EE2	000	05B1	0EE2	0001		
5B2	0740	5B3	0740	5B2	0740	000	05B2	0EE1	0001		
5B3	4E15	5B4	4E15	5C9	0000	000	0015	0EE1	0000		
5B4	EE14	5B5	EE14	5C9	0EE1	000	0014	0EE1	0000	5C9	0EE1
5B5	AE10	5B6	AE10	5C6	FFFB	000	0010	FFFB	1000		
5B6	0740	5B7	0740	5B6	0740	000	05B6	FFFA	1001		
5B7	0C00	5B8	0C00	7FF	FFFA	7FF	05B7	FFFA	1001	7FF	FFFA
5B8	D684	684	D684	7FE	05B9	7FE	D684	FFFA	1001	7FE	05B9
684	AC01	685	AC01	7FF	FFFA	7FE	0001	FFFA	1001		
685	F205	68B	F205	685	F205	7FE	0005	FFFA	1001		
68B	AE02	68C	AE02	68E	0EE2	7FE	0002	0EE2	0001		
68C	EC01	68D	EC01	7FF	0EE2	7FE	0001	0EE2	0001	7FF	0EE2
68D	0A00	5B9	0A00	7FE	05B9	7FF	068D	0EE2	0001		
5B9	0800	5BA	0800	7FF	0EE2	000	05B9	0EE2	0001		
5BA	0740	5BB	0740	5BA	0740	000	05BA	0EE1	0001		
5BB	6E0D	5BC	6E0D	5C9	0EE1	000	000D	0000	0101		
5BC	EE0C	5BD	EE0C	5C9	0000	000	000C	0000	0101	5C9	0000
5BD	AE09	5BE	AE09	5C7	0001	000	0009	0001	0001		
5BE	0700	5BF	0700	5BE	0700	000	05BE	0002	0000		
5BF	0C00	5C0	0C00	7FF	0002	7FF	05BF	0002	0000	7FF	0002

Таблица 1.4: Трассировка программы

Адр	Знач	IP	CR	AR	DR	SP	BR	AC	NZVC	Адр	Знач
xxx	xxxx	xxx	xxxx	xxx	xxxx	xxx	xxxx	xxxx	xxxx	xxx	xxxx
5C0	D684	684	D684	7FE	05C1	7FE	D684	0002	0000	7FE	05C1
684	AC01	685	AC01	7FF	0002	7FE	0001	0002	0000		
685	F205	686	F205	685	F205	7FE	0685	0002	0000		
686	7E07	687	7E07	68E	0EE2	7FE	0007	0002	1000		
687	F903	688	F903	687	F903	7FE	0687	0002	1000		
688	0500	689	0500	688	0002	7FE	0688	0004	0000		
689	4E05	68A	4E05	68F	0064	7FE	0005	0068	0000		
68A	CE01	68C	CE01	68A	068C	7FE	0001	0068	0000		
68C	EC01	68D	EC01	7FF	0068	7FE	0001	0068	0000	7FF	0068
68D	0A00	5C1	0A00	7FE	05C1	7FF	068D	0068	0000		
5C1	0800	5C2	0800	7FF	0068	000	05C1	0068	0000		
5C2	0740	5C3	0740	5C2	0740	000	05C2	0067	0001		
5C3	4E05	5C4	4E05	5C9	0000	000	0005	0067	0000		
5C4	EE04	5C5	EE04	5C9	0067	000	0004	0067	0000	5C9	0067
5C5	0100	5C6	0100	5C5	0100	000	05C5	0067	0000		
684	AC01	685	AC01	001	0000	000	0001	0000	0100		
685	F205	686	F205	685	F205	000	0685	0000	0100		
686	7E07	687	7E07	68E	0EE2	000	0007	0000	1000		
687	F903	688	F903	687	F903	000	0687	0000	1000		
688	0500	689	0500	688	0000	000	0688	0000	0100		
689	4E05	68A	4E05	68F	0064	000	0005	0064	0000		
68A	CE01	68C	CE01	68A	068C	000	0001	0064	0000		
68C	EC01	68D	EC01	001	0064	000	0001	0064	0000	001	0064
68D	0A00	000	0A00	000	0000	001	068D	0064	0000		

Таблица 1.5: Трассировка программы

2. 4. Получение новых чисел

- $Adr = 100$
- $N = 4$
- $Arr = [-5, 16, 0, -182]$
- $R = -187 = FF45$

3. Вывод

Во время выполнения данной лабораторной работы я научился работать со стеком и понял, как вызывать и писать подпрограммы.