

Федеральное государственное автономное образовательное учреждение высшего
образования «Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Лабораторная работа №4

Выполнение комплекса программ

Вариант №45067

Выполнил

Лабин Макар Андреевич

группа Р3131

Проверил

Обляшевский Севастьян Александрович

г. Санкт-Петербург, 2025

Содержание

Задание.....	3
Выполнение работы.....	4
Заключение.....	7

Задание

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

45067

2DD: + 0200		2EB: 4E0D		2F9: F649		691: 0A00
2DE: EE1A		2EC: EE0C		-----		692: FCC3
2DF: AE18		2ED: AE08		685: AC01		693: 0047
2E0: 0C00		2EE: 0740		686: F001		
2E1: D685		2EF: 0C00		687: F304		
2E2: 0800		2F0: D685		688: 6E09		
2E3: 4E15		2F1: 0800		689: F201		
2E4: EE14		2F2: 0740		68A: CE04		
2E5: AE11		2F3: 4E05		68B: 4E06		
2E6: 0740		2F4: EE04		68C: 4C01		
2E7: 0C00		2F5: 0100		68D: 4E05		
2E8: D685		2F6: ZZZZ		68E: CE01		
2E9: 0800		2F7: YYYY		68F: AE02		
2EA: 0700		2F8: XXXX		690: EC01		

Выполнение работы

1. Текст исходной программы:

Адрес	Код	Мнемоника	Комментарий
2DD	0200	CLA	
2DE	EE1A	ST 1A	$A = 0;$
2DF	AE18	LD 18	$AC = X;$
2E0	0C00	PUSH	
2E1	D685	CALL \$685	$AC = f(X);$
2E2	0800	POP	
2E3	4E15	ADD 15	
2E4	EE14	ST 14	$A += f(X);$
2E5	AE11	LD 11	
2E6	0740	DEC	$AC = Y - 1;$
2E7	0C00	PUSH	
2E8	D685	CALL \$685	$AC = f(Y - 1);$
2E9	0800	POP	
2EA	0700	INC	
2EB	4E0D	ADD D	$A += f(Y - 1) + 1$
2EC	EE0C	ST C	
2ED	AE08	LD 8	
2EE	0740	DEC	$AC = Z - 1;$
2EF	0C00	PUSH	
2F0	D685	CALL \$685	$AC = f(Z - 1);$
2F1	0800	POP	
2F2	0740	DEC	
2F3	4E05	ADD 5	$A += f(Z - 1) - 1;$
2F4	EE04	ST 4	
2F5	0100	HLT	Конец программы.
2F6	ZZZZ	;Z	Константа Z.
2F7	YYYY	;Y	Константа Y.
2F8	XXXX	;X	Константа X.
2F9	F649	;A	Переменная A — результат программы.

Функция f(k)			
685	AC01	LD &1	Загрузить в AC последнюю переменную стека (сразу после кода возврата): обозначим её за <i>k</i> .
686	F001	BZS 1	void f(*k) {
687	F304	BNC 4	AC = *k;
688	6E09	SUB 9	if (AC == 0 AC < 0) {
689	F201	BNS 1	AC -= S;
68A	CE04	JUMP 4	if (AC >= 0) {
68B	4E06	ADD 6	AC = S;
68C	4C01	ADD &1	*k = AC;
68D	4E05	ADD 5	return;
68E	CE01	JUMP 1	}
68F	AE02	LD 2	AC += k;
690	EC01	ST &1	AC += T;
691	0A00	RET	*k = AC;
692	FCC3	;S	}
693	0047	;T	Константа S = -829
			Константа T = 71.

2. Описание программы:

1) *Назначение программы и реализуемые ей функции.* Программа предназначена для расчёта формулы:

$$f(k) = \begin{cases} 2k + T, & k > 0 \\ 2k - S + T, & k - S < 0 \\ S, & k - S \geq 0 \end{cases}$$

$$A = f(X) + f(Y - 1) + 1 + f(Z - 1) - 1$$

$$A = f(X) + f(Y - 1) + f(Z - 1)$$

2) *Область представления и область допустимых значений исходных данных и результата.*

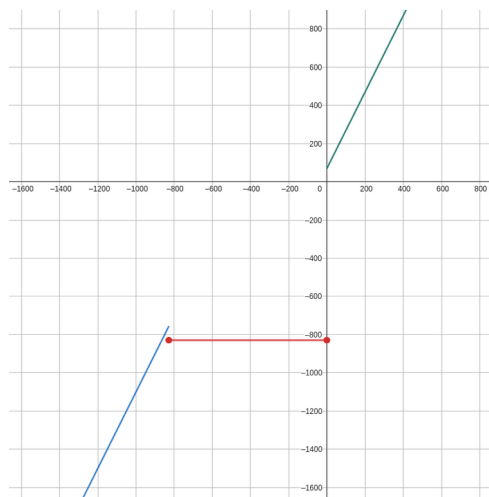
A, X, Y, Z, T, S — знаковые, 16-разрядные числа.

Область допустимых значений:

$$S = -829$$

$$T = 71$$

График функции $f(k)$ (обрезан, $-2^{15} \leq k \leq 2^{15}-1$)



$$k \in [-829; 0] \Rightarrow f(k) = -829 \Rightarrow \text{OF} = 0$$

$$k \in (0, 2^{15} - 1] \Rightarrow f \text{ возрастает, } f \in [73; 2^{16} + 69] \text{ с шагом } 2 \Rightarrow \text{OF} = 1, k \geq 2^{14} - 35$$

$$k \in [-2^{15}, -829) \Rightarrow f \text{ возрастает, } f \in [-2^{16} + 900; -760] \text{ с шагом } 2 \Rightarrow \text{OF} = 1, k \leq -2^{14} - 451$$

Значит, при $-2^{14}-450 \leq k \leq 2^{14}-36$ переполнения нет.

Осталось подобрать такие комбинации X, Y, Z, чтобы A не переполнялось.

3) *Расположение в памяти ЭВМ программы, исходных данных и результатов.* Исходные данные располагаются по адресам 2F6, 2F7, 2F8, 692, 293. Промежуточные значения хранятся в стеке, а результат — по адресу 2F9.

4) *Адреса первой и последней выполняемой команд программы.* Набор исполняемых команд основной программы располагается в диапазоне [2DD; 2F5], при этом в программе есть команды ветвления, поэтому не все операции могут быть задействованы в ходе работы программы. Также в коде вызов подпрограммы, команды которой расположены по адресу [685; 691] (в ней также есть команды ветвления).

3. Таблица трассировки для исходных данных X=0xA6, Y=0xFFD3, Z=0x5:

Выполняемая команда		Содержимое регистров процессора после выполнения команды							Ячейка, содержащее которой изменилось после выполнения команды	
Адрес	Код	IP	AR	CR	DR	AC	SP	NZVC	Адрес	Новый код
2DD	0200	2DD	000	0000	0000	0000	000	0100		
2DD	0200	2DE	2DD	0200	0200	0000	000	0100		
2DE	EE1A	2DF	2F9	EE1A	0000	0000	000	0100	2F9	0000
2DF	AE18	2E0	2F8	AE18	00A6	00A6	000	0000		
2E0	0C00	2E1	7FF	0C00	00A6	00A6	7FF	0000	7FF	00A6
2E1	D685	685	7FE	D685	02E2	00A6	7FE	0000	7FE	02E2
685	AC01	686	7FF	AC01	00A6	00A6	7FE	0000		
686	F001	687	686	F001	F001	00A6	7FE	0000		
687	F304	68C	687	F304	F304	00A6	7FE	0000		
68C	4C01	68D	7FF	4C01	00A6	014C	7FE	0000		
68D	4E05	68E	693	4E05	0047	0193	7FE	0000		
68E	CE01	690	68E	CE01	0690	0193	7FE	0000		
690	EC01	691	7FF	EC01	0193	0193	7FE	0000	7FF	0193
691	0A00	2E2	7FE	0A00	02E2	0193	7FF	0000		
2E2	0800	2E3	7FF	0800	0193	0193	000	0000		
2E3	4E15	2E4	2F9	4E15	0000	0193	000	0000		
2E4	EE14	2E5	2F9	EE14	0193	0193	000	0000	2F9	0193
2E5	AE11	2E6	2F7	AE11	FFD3	FFD3	000	1000		
2E6	0740	2E7	2E6	0740	0740	FFD2	000	1001		
2E7	0C00	2E8	7FF	0C00	FFD2	FFD2	7FF	1001	7FF	FFD2
2E8	D685	685	7FE	D685	02E9	FFD2	7FE	1001	7FE	02E9
685	AC01	686	7FF	AC01	FFD2	FFD2	7FE	1001		
686	F001	687	686	F001	F001	FFD2	7FE	1001		
687	F304	688	687	F304	F304	FFD2	7FE	1001		
688	6E09	689	692	6E09	FCC3	030F	7FE	0001		
689	F201	68A	689	F201	F201	030F	7FE	0001		
68A	CE04	68F	68A	CE04	068F	030F	7FE	0001		
68F	AE02	690	692	AE02	FCC3	FCC3	7FE	1001		
690	EC01	691	7FF	EC01	FCC3	FCC3	7FE	1001	7FF	FCC3
691	0A00	2E9	7FE	0A00	02E9	FCC3	7FF	1001		
2E9	0800	2EA	7FF	0800	FCC3	FCC3	000	1001		
2EA	0700	2EB	2EA	0700	0700	FCC4	000	1000		
2EB	4E0D	2EC	2F9	4E0D	0193	FE57	000	1000		
2EC	EE0C	2ED	2F9	EE0C	FE57	FE57	000	1000	2F9	FE57
2ED	AE08	2EE	2F6	AE08	0005	0005	000	0000		
2EE	0740	2EF	2EE	0740	0740	0004	000	0001		
2EF	0C00	2F0	7FF	0C00	0004	0004	7FF	0001	7FF	0004
2F0	D685	685	7FE	D685	02F1	0004	7FE	0001	7FE	02F1
685	AC01	686	7FF	AC01	0004	0004	7FE	0001		
686	F001	687	686	F001	F001	0004	7FE	0001		
687	F304	68C	687	F304	F304	0004	7FE	0001		
68C	4C01	68D	7FF	4C01	0004	0008	7FE	0000		
68D	4E05	68E	693	4E05	0047	004F	7FE	0000		

68E	CE01	690	68E	CE01	0690	004F	7FE	0000		
690	EC01	691	7FF	EC01	004F	004F	7FE	0000	7FF	004F
691	0A00	2F1	7FE	0A00	02F1	004F	7FF	0000		
2F1	0800	2F2	7FF	0800	004F	004F	000	0000		
2F2	0740	2F3	2F2	0740	0740	004E	000	0001		
2F3	4E05	2F4	2F9	4E05	FE57	FEA5	000	1000		
2F4	EE04	2F5	2F9	EE04	FEA5	FEA5	000	1000	2F9	FEA5
2F5	0100	2F6	2F5	0100	0100	FEA5	000	1000		

Заключение

В ходе выполнения лабораторной работы я изучил способы связи между программными модулями, команды для обращения к подпрограмме и взаимодействия со стеком и применил их на практике в процессе анализа работы программы, которая попала мне в выданном варианте.