



УНИВЕРСИТЕТ ИТМО

**Системное и прикладное программное обеспечение.
Программная инженерия.**

**Лабораторная работа №4.
Выполнение комплекса программ.**

Предмет: ОПД.

Преподаватель: Блохина Елена Николаевна

Выполнил: Бусыгин Иван.

Группа: Р3112.

Вариант: 2122.

Санкт-Петербург
2021 год

Цель работы.

Изучение способов связи между программными модулями, команды обращения к подпрограмме и исследование порядка функционирования БЭВМ при выполнении комплекса взаимосвязанных программ.

Задание.

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

0CB: + 0200	0D9: 0740	0E7: XXXX	6A3: EC01
0CC: EE1B	0DA: 4E0D	0E8: FA8A	6A4: 0A00
0CD: AE19	0DB: EE0C	-----	6A5: 028B
0CE: 0C00	0DC: AE08	698: AC01	6A6: 0065
0CF: D698	0DD: 0700	699: F208	
0D0: 0800	0DE: 0C00	69A: F007	
0D1: 0700	0DF: D698	69B: 7E09	
0D2: 6E15	0E0: 0800	69C: F905	
0D3: EE14	0E1: 0700	69D: 4C01	
0D4: AE11	0E2: 6E05	69E: 4C01	
0D5: 0740	0E3: EE04	69F: 4C01	
0D6: 0C00	0E4: 0100	6A0: 6E05	
0D7: D698	0E5: ZZZZ	6A1: CE01	
0D8: 0800	0E6: YYYY	6A2: AE02	

Описание программы.

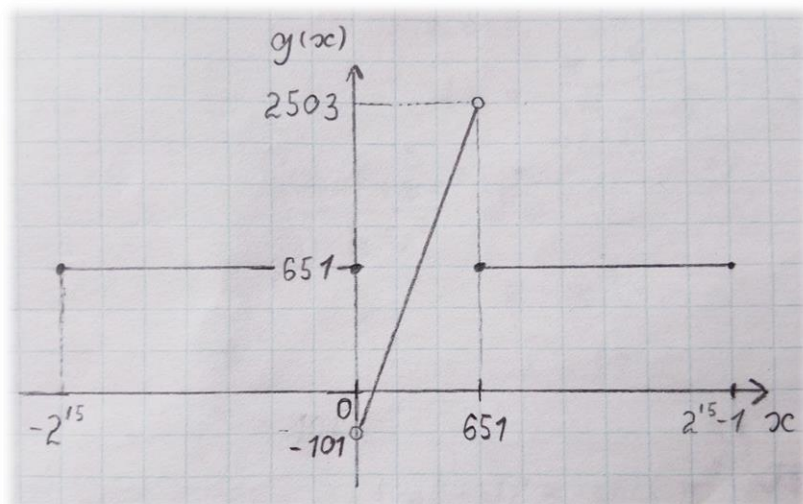
Комплекс программ предназначен для расчёта по следующей формуле:

$$f(x, y, z) = g(z + 1) - g(x) - g(y - 1) + 1$$

$$g(x) = \begin{cases} a, & x \leq 0 \vee x \geq a \\ 4x - b, & x \in (0; a) \end{cases}$$

В нашем случае $a = 651$, $b = 101$.

График функции $g(x)$:



Расположение в памяти:

Основная программа расположена в ячейках с 0СВ (первая исполняемая команда) по 0Е4 (последняя).

Подпрограмма расположена в ячейках с 698 (первая исполняемая команда) по 6А4 (последняя).

Входные переменные хранятся в ячейках 0Е5, 0Е6, 0Е7.

Ячейка с результатом – 0Е8

Параметры а и b хранятся в ячейках 6А5, 6А6.

Вычисление ОДЗ.**Ограничения.**

Для $g(x)$:

$$\left\{ \begin{array}{l} x \in [-2^{15}; 2^{15} - 1] \\ x \leq 0 \vee x \geq a \\ \left\{ \begin{array}{l} x \in [1; a - 1] \\ 4x \in [-2^{15}; 2^{15} - 1] \\ 4x - b \in [-2^{15}; 2^{15} - 1] \end{array} \right. \end{array} \right.$$

Для $f(x)$:

$$\left\{ \begin{array}{l} x, y, z \in [-2^{15}; 2^{15} - 1] \\ z + 1, x, y - 1 \in \text{ОДЗ}(g) \\ g(z + 1) - g(x) \in [-2^{15}; 2^{15} - 1] \\ f(x, y, z) \in [1 - 2^{15}; 2^{15} - 1] \end{array} \right.$$

ОДЗ.

Для $g(x)$:

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

Для $f(x)$:

$$\left\{ \begin{array}{l} x \in [-2^{15}; 2^{15} - 1] \\ y \in [1 - 2^{15}; 2^{15} - 1] \\ z \in [-2^{15}; 2^{15} - 2] \end{array} \right.$$

Область представления.

$$g(x) \in [-97; 2499]$$

$$f(x, y, z) \in [1 - 2^{15}; 2^{15} - 1]$$

Значения переменных, полученные у преподавателя:

X: 000D Y: 0F0D Z: FF0D

Текст основной программы.

Адрес	Код команды	Мнемоника	Комментарий
0CB	0200	CLA	Очистка аккумулятора. AC = 0
0CC	EE1B	ST IP + 1B	Сохранение, относительная адресация. t = AC = 0
0CD	AE19	LD IP + 19	Загрузка, относительная адресация. AC = X
0CE	0C00	PUSH	Положить в стек. arg = AC
0CF	D698	CALL 698	Вызов подпрограммы. arg = g(arg)
0D0	0800	POP	Взять из стека. AC = arg
0D1	0700	INC	Инкремент. AC++
0D2	6E15	SUB IP + 15	Вычитание, относительная адресация. AC -= t
0D3	EE14	ST IP + 14	Сохранение, относительная адресация. t = AC
0D4	AE11	LD IP + 11	Загрузка, относительная адресация. AC = Y
0D5	0740	DEC	Декремент. AC--
0D6	0C00	PUSH	Положить в стек. arg = AC
0D7	D698	CALL 698	Вызов подпрограммы. arg = g(arg)
0D8	0800	POP	Взять из стека. AC = arg
0D9	0740	DEC	Декремент. AC--
0DA	4E0D	ADD IP + D	Сложение, относительная адресация. AC += t
0DB	EE0C	ST IP + C	Сохранение, относительная адресация. t = AC
0DC	AE08	LD IP + 8	Загрузка, относительная адресация. AC = Z
0DD	0700	INC	Инкремент. AC++
0DE	0C00	PUSH	Положить в стек. arg = AC
0DF	D698	CALL 698	Вызов подпрограммы. arg = g(arg)
0E0	0800	POP	Взять из стека. AC = arg
0E1	0700	INC	Инкремент. AC++
0E2	6E05	SUB IP + 5	Вычитание, относительная адресация. AC -= t
0E3	EE04	ST IP + 4	Сохранение, относительная адресация. t = AC
0E4	0100	HLT	Отключение ТГ, переход в пультовый режим.

Текст подпрограммы.

Адрес	Код команды	Мнемоника	Комментарий
698	AC01	LD &1	Загрузка аргумента из стека. AC = arg
699	F208	BMI IP + 8	Переход, если меньше нуля. if AC < 0 then IP = 6A2
69A	F007	BEQ IP + 7	Переход, если равно нулю. if AC = 0 then IP = 6A2
69B	7E09	CMP IP + 9	Сравнение, относительная адресация.
69C	F905	BGE IP + 5	Переход, если больше или равно. if AC ≥ a then IP = 6A2
69D	4C01	ADD &1	Сложение с аргументом из стека. AC += arg
69E	4C01	ADD &1	Сложение с аргументом из стека. AC += arg
69F	4C01	ADD &1	Сложение с аргументом из стека. AC += arg
GA0	6E05	SUB IP + 5	Вычитание, относительная адресация. AC -= b
6A1	CE01	JUMP IP + 1	Перемещение по относительному адресу. IP = 6A3
6A2	AE02	LD IP + 2	Загрузка, относительная адресация. AC = a
6A3	EC01	ST &1	Сохранение в ячейку с аргументом. arg = AC
6A4	0A00	RET	Возврат из подпрограммы.

Выполняемая команда.		Содержимое регистров процессора после выполнения команды.								Ячейка, содержимое которой изменилось после выполнения команды.	
Адрес	Код	IP	CR	AR	DR	SP	BR	AC	NZVC	Адрес	Новый код
0CB	0200	0CC	0200	0CB	0200	000	00CB	0000	0100		
0CC	EE1B	0CD	EE1B	0E8	0000	000	001B	0000	0100	0E8	0000
0CD	AE19	0CE	AE19	0E7	000D	000	0019	000D	0000		
0CE	0C00	0CF	0C00	7FF	000D	7FF	00CE	000D	0000	7FF	000D
0CF	D698	698	D698	7FE	00D0	7FE	D698	000D	0000	7FE	00D0
698	AC01	699	AC01	7FF	000D	7FE	0001	000D	0000		
699	F208	69A	F208	699	F208	7FE	0699	000D	0000		
69A	F007	69B	F007	69A	F007	7FE	069A	000D	0000		
69B	7E09	69C	7E09	6A5	028B	7FE	0009	000D	1000		
69C	F905	69D	F905	69C	F905	7FE	069C	000D	1000		
69D	4C01	69E	4C01	7FF	000D	7FE	0001	001A	0000		
69E	4C01	69F	4C01	7FF	000D	7FE	0001	0027	0000		
69F	4C01	6A0	4C01	7FF	000D	7FE	0001	0034	0000		
6A0	6E05	6A1	6E05	6A6	0065	7FE	0005	FFCF	1000		
6A1	CE01	6A3	CE01	6A1	06A3	7FE	0001	FFCF	1000		
6A3	EC01	6A4	EC01	7FF	FFCF	7FE	0001	FFCF	1000	7FF	FFCF
6A4	0A00	0D0	0A00	7FE	00D0	7FF	06A4	FFCF	1000		
0D0	0800	0D1	0800	7FF	FFCF	000	00D0	FFCF	1000		
0D1	0700	0D2	0700	0D1	0700	000	00D1	FFD0	1000		
0D2	6E15	0D3	6E15	0E8	0000	000	0015	FFD0	1001		
0D3	EE14	0D4	EE14	0E8	FFD0	000	0014	FFD0	1001	0E8	FFD0
0D4	AE11	0D5	AE11	0E6	0F0D	000	0011	0F0D	0001		
0D5	0740	0D6	0740	0D5	0740	000	00D5	0F0C	0001		
0D6	0C00	0D7	0C00	7FF	0F0C	7FF	00D6	0F0C	0001	7FF	0F0C
0D7	D698	698	D698	7FE	00D8	7FE	D698	0F0C	0001	7FE	00D8
698	AC01	699	AC01	7FF	0F0C	7FE	0001	0F0C	0001		
699	F208	69A	F208	699	F208	7FE	0699	0F0C	0001		
69A	F007	69B	F007	69A	F007	7FE	069A	0F0C	0001		
69B	7E09	69C	7E09	6A5	028B	7FE	0009	0F0C	0001		
69C	F905	6A2	F905	69C	F905	7FE	0005	0F0C	0001		
6A2	AE02	6A3	AE02	6A5	028B	7FE	0002	028B	0001		
6A3	EC01	6A4	EC01	7FF	028B	7FE	0001	028B	0001	7FF	028B
6A4	0A00	0D8	0A00	7FE	00D8	7FF	06A4	028B	0001		
0D8	0800	0D9	0800	7FF	028B	000	00D8	028B	0001		
0D9	0740	0DA	0740	0D9	0740	000	00D9	028A	0001		
0DA	4E0D	0DB	4E0D	0E8	FFD0	000	000D	025A	0001		
0DB	EE0C	0DC	EE0C	0E8	025A	000	000C	025A	0001	0E8	025A
0DC	AE08	0DD	AE08	0E5	FF0D	000	0008	FF0D	1001		
0DD	0700	0DE	0700	0DD	0700	000	00DD	FF0E	1000		
0DE	0C00	0DF	0C00	7FF	FF0E	7FF	00DE	FF0E	1000	7FF	FF0E
0DF	D698	698	D698	7FE	00E0	7FE	D698	FF0E	1000	7FE	00E0
698	AC01	699	AC01	7FF	FF0E	7FE	0001	FF0E	1000		
699	F208	6A2	F208	699	F208	7FE	0008	FF0E	1000		
6A2	AE02	6A3	AE02	6A5	028B	7FE	0002	028B	0000		
6A3	EC01	6A4	EC01	7FF	028B	7FE	0001	028B	0000	7FF	028B
6A4	0A00	0E0	0A00	7FE	00E0	7FF	06A4	028B	0000		
0E0	0800	0E1	0800	7FF	028B	000	00E0	028B	0000		
0E1	0700	0E2	0700	0E1	0700	000	00E1	028C	0000		
0E2	6E05	0E3	6E05	0E8	025A	000	0005	0032	0001		
0E3	EE04	0E4	EE04	0E8	0032	000	0004	0032	0001	0E8	0032
0E4	0100	0E5	0100	0E4	0100	000	00E4	0032	0001		

Вывод.

Я освоил методы реализации подпрограмм на БВМ: ознакомился с необходимыми командами, узнал способы передачи данных между основной и вложенной программами.