

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

Факультет Программной Инженерии и Компьютерной Техники

Лабораторная работа №3

По дисциплине

«Основы профессиональной деятельности»

Вариант 49331

Выполнил:

Петров Вячеслав Маркович

Группа Р3108

Принял:

Вербовой Александр Александрович

Содержание

<i>Задание</i>	3
<i>Описание программы</i>	4
Назначение программы	4
Область представления	4
Область допустимых значений	4
Расположение данных в памяти	4
Адреса первой и последней выполняемой команды	4
<i>Трассировка программы</i>	5
<i>Вывод</i>	6

Задание

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

4C2:	04D6	4D0:	7EF4
4C3:	0200	4D1:	F901
4C4:	E000	4D2:	EEF2
4C5:	0200	4D3:	84C4
4C6:	+ AF80	4D4:	CEF9
4C7:	0740	4D5:	0100
4C8:	0680	4D6:	F300
4C9:	EEFB	4D7:	F000
4CA:	AF04	4D8:	F500
4CB:	EEF8	4D9:	FA00
4CC:	AEF5		
4CD:	EEF5		
4CE:	AAF4		
4CF:	F203		

Адрес	Код команды	Мнемоника	Комментарии
4C2	04D6	addr_first_elem	Адрес первого элемента
4C3	0200	current_elem	Адрес текущего элемента (начиная с первого) //04D6
4C4	E000	arr_length	Количество элементов массива //0004
4C5	0200	result	Результат //7FFF по базе
4C6	AF80	LD #80	Прямая загрузка FF80 -> AC
4C7	0740	DEC	Вычитание 1: AC – 1 -> AC
4C8	0680	SWAB	Обмен ст. и мл. байта AC7...AC0 <-> AC15...AC8
4C9	EEFB	ST (IP-5)	Прямое относительное сохранение AC -> M (4C5)
4CA	AF04	LD #04	Прямая загрузка 0004 -> AC // устанавливаем длину массива [0; 127]
4CB	EEF8	ST (IP-8)	Прямое относительное сохранение AC -> M (4C4)
4CC	AEF5	LD (IP-11)	Прямая относительная загрузка M(4C2) -> AC //04D6
4CD	EEF5	ST (IP-11)	Прямое относительное сохранение AC -> M (4C3)
4CE	AAF4	LD (IP-12) +	Косвенная автоинкрементная загрузка: M(3н(4C3)) -> AC; M(4C3) += 1 Точка начала цикла
4CF	F203	BMI (IP+3)	Если N==1, то IP + 3 + 1 -> IP
4D0	7EF4	CMP (IP-12)	Прямое относительное установление флагов по результату AC – M(4C5)
4D1	F901	BGE (IP+1)	Если $N \oplus V = 0$, то IP + 1 + 1 -> IP
4D2	EEF2	ST (IP-14)	Прямое относительное сохранение AC -> M (4C5)
4D3	84C4	LOOP 4C4	M(4C4) – 1 -> M(4C4); Если M(4C4) <= 0, то IP + 1 -> IP
4D4	CEF9	JUMP (IP-7)	Прямой относительный прыжок IP - 7 + 1 -> IP
4D5	0100	HLT	Останов
4D6	F300		Элементы массива
4D7	F000		
4D8	F500		
4D9	FA00		

Описание программы

Назначение программы

Находит минимальное неотрицательное число в массиве. Если все числа отрицательные, то $result = 2^{15} - 1$ (данное значение задается в программе).

Область представления

- `addr_first_elem`, `current_elem` – 11-ти разрядные, адрес БЭВМ
- `arr_length`, `result` – 16-ти разрядные целые числа, беззнаковое
- элементы массива – 16-ти разрядные знаковые целые числа

Область допустимых значений

- $0 \leq arr_length \leq 127$
- $$\begin{cases} 0 \leq addr_first_elem \leq 4C2 - arr_length \\ 4D6 \leq addr_first_elem \leq 7FF - arr_length \end{cases}$$
- $addr_first_elem \leq current_elem \leq addr_first_elem + arr_length - 1$
- $-2^{15} \leq \text{элементы массива} \leq 2^{15} - 1$
- $0 \leq result \leq 2^{15} - 1$

Расположение данных в памяти

- 4C2, 4C4, 4D6, 4D7, 4D8, 4D9 – исходные данные;
- 4C3 – промежуточный результат;
- 4C5 – итоговый результат;
- 4C6 – 4D5 – команды.

Адреса первой и последней выполняемой команды

- Адрес первой команды: 4C6
- Адрес последней команды: 4D5

Трассировка программы

Значения:

arr = [0x1313, 0xB00B, 0x3138]; arr_length = 3; first_elem = 0x04D6

(выдали новые числа)

Выполняемая команда		Содержимое регистров после выполнения команды									Ячейка, содержимое которой изменилось после выполнения команды	
Адрес	Код Команды	IP	CR	AR	DR	SP	BR	AC	PS	NZVC	Адрес	Новый код
4C6	AF80	4C6	0000	000	0000	000	0000	0000	004	0100		
4C6	AF80	4C7	AF80	4C6	FF80	000	FF80	FF80	008	1000		
4C7	0740	4C8	0740	4C7	0740	000	04C7	FF7F	009	1001		
4C8	0680	4C9	0680	4C8	0680	000	04C8	7FFF	001	0001		
4C9	EEFB	4CA	EEFB	4C5	7FFF	000	FFF B	7FFF	001	0001	4C5	7FFF
4CA	AF03	4CB	AF03	4CA	0003	000	0003	0003	001	0001		
4CB	EEF8	4CC	EEF8	4C4	0003	000	FFF8	0003	001	0001	4C4	0003
4CC	AEF5	4CD	AEF5	4C2	04D6	000	FFF5	04D6	001	0001		
4CD	EEF5	4CE	EEF5	4C3	04D6	000	FFF5	04D6	001	0001	4C3	04D6
4CE	AAF4	4CF	AAF4	4D6	1313	000	FFF4	1313	001	0001	4C3	04D7
4CF	F203	4D0	F203	4CF	F203	000	04CF	1313	001	0001		
4D0	7EF4	4D1	7EF4	4C5	7FFF	000	FFF4	1313	008	1000		
4D1	F901	4D2	F901	4D1	F901	000	04D1	1313	008	1000		
4D2	EEF2	4D3	EEF2	4C5	1313	000	FFF2	1313	008	1000	4C5	1313
4D3	84C4	4D4	84C4	4C4	0002	000	0001	1313	008	1000	4C4	0002
4D4	CEF9	4CE	CEF9	4D4	04C E	000	FFF9	1313	008	1000		
4CE	AAF4	4CF	AAF4	4D7	B00B	000	FFF4	B00B	008	1000	4C3	04D8
4CF	F203	4D3	F203	4CF	F203	000	0003	B00B	008	1000		
4D3	84C4	4D4	84C4	4C4	0001	000	0000	B00B	008	1000	4C4	0001
4D4	CEF9	4CE	CEF9	4D4	04C E	000	FFF9	B00B	008	1000		
4CE	AAF4	4CF	AAF4	4D8	3138	000	FFF4	3138	000	0000	4C3	04D9
4CF	F203	4D0	F203	4CF	F203	000	04CF	3138	000	0000		
4D0	7EF4	4D1	7EF4	4C5	1313	000	FFF4	3138	001	0001		
4D1	F901	4D3	F901	4D1	F901	000	0001	3138	001	0001		
4D3	84C4	4D5	84C4	4C4	0000	000	FFFF	3138	001	0001	4C4	0000
4D5	0100	4D6	0100	4D5	0100	000	04D5	3138	001	0001		

Вывод

Во время выполнения лабораторной работы я ознакомился с режимами адресации, работой циклических программ в БЭВМ и изучил цикл выполнения команд LOOP и JUMP. Также научился взаимодействовать с элементами одномерного массива.