

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

Факультет программной инженерии и компьютерной техники

Лабораторная работа №4

Вариант №43322

Выполнил

Макогон Ярослав Вадимович

Номер группы: Р3118

Проверил

Ермаков М.К.

Содержание

Задание.....	3
Решение.....	4
Доп.....	8
Вывод.....	9

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.

293: + 0200		2A1: 0740		-----		68C: 0A00
294: EE19		2A2: 4E0B		67F: AC01		68D: FC29
295: AE15		2A3: EE0A		680: F001		68E: 0012
296: 0740		2A4: AE07		681: F304		
297: 0C00		2A5: 0C00		682: 6E0A		
298: D67F		2A6: D67F		683: F201		
299: 0800		2A7: 0800		684: CE05		
29A: 4E13		2A8: 6E05		685: 4E07		
29B: EE12		2A9: EE04		686: 4C01		
29C: AE10		2AA: 0100		687: 4C01		
29D: 0740		2AB: ZZZZ		688: 4E05		
29E: 0C00		2AC: YYY Y		689: CE01		
29F: D67F		2AD: XXXX		68A: AE02		
2A0: 0800		2AE: 03D8		68B: EC01		

Решение

Адрес	Код команды	Мнемоника	Комментарий	Вид адресации
293	0200+	CLA	0 -> AC	Безадресная
294	EE19	ST (IP+25)	AC -> (2AE)	Прямая относительная
295	AE15	LD (IP+21)	(2AB) -> AC	Прямая относительная
296	0740	DEC	AC-1 -> AC	Безадресная
297	0C00	PUSH	AC -> -(SP)	Безадресная
298	D67F	CALL (67F)	SP-1 -> SP; IP -> (SP); (67F) -> IP	Абсолютная
299	0800	POP	(SP)+ -> AC	Безадресная
29A	4E13	ADD (IP+19)	(2AE) + AC -> AC	Прямая относительная
29B	EE12	ST (IP+18)	AC -> (2AE)	Прямая относительная
29C	AE10	LD (IP+16)	(2AD) -> AC	Прямая относительная
29D	0740	DEC	AC-1 -> AC	Безадресная
29E	0C00	PUSH	AC -> -(SP)	Безадресная
29F	D67F	CALL (67F)	SP-1 -> SP; IP -> (SP); (67F) -> IP	Абсолютная
2A0	0800	POP	(SP)+ -> AC	Безадресная
2A1	0740	DEC	AC-1 -> AC	Безадресная
2A2	4E0B	ADD (IP+11)	(2AE) + AC -> AC	Прямая относительная
2A3	EE0A	ST (IP+10)	AC -> (2AE)	Прямая относительная
2A4	AE07	LD (IP+7)	(2AC) -> AC	Прямая относительная
2A5	0C00	PUSH	AC -> -(SP)	Безадресная
2A6	D67F	CALL (67F)	SP-1 -> SP; IP -> (SP); (67F) -> IP	Абсолютная
2A7	0800	POP	(SP)+ -> AC	Безадресная
2A8	6E05	SUB (IP+5)	AC-(2AE) -> AC	Прямая относительная
2A9	EE04	ST (IP+4)	AC -> (2AE)	Прямая относительная
2AA	0100	HLT	Останов	Безадресная
2AB	ZZZZ		Z	-
2AC	YYYY		Y	-
2AD	XXXX		X	-
2AE	03D8		Результат (RES)	-
67F	AC01	LD (SP+1)	(SP+1)->AC	Со смещением относительно SP
680	F001	BEQ	Z==1: IP+1 -> IP	Ветвление
681	F304	BPL	N==0: IP+4 -> IP	Ветвление
682	6E0A	SUB (IP+10)	AC-(68D) -> AC	Прямая относительная
683	F201	BMI	N==1: IP+1 -> IP	Ветвление
684	CE05	JUMP (IP+5)	(68A) -> IP	Прямая относительная
685	4E07	ADD (IP+7)	AC + (68D) -> AC	Прямая относительная
686	4C01	ADD (SP+1)	AC + (SP+1) -> AC	Со смещением относительно SP
687	4C01	ADD (SP+1)	AC + (SP+1) -> AC	Со смещением относительно SP
688	4E05	ADD (IP+5)	AC + (68E) -> AC	Прямая относительная

689	CE01	JUMP (IP+1)	(68B) -> IP	Прямая относительная
68A	AE02	LD (IP+2)	(68D) -> AC	Прямая относительная
68B	EC01	ST (SP+1)	AC -> (SP+1)	Со смещением относительно SP
68C	0A00	RET	(ST)+ -> IP	Безадресная
68D	FC29		-983 (A)	-
68E	0012		18 (B)	-

Назначение программы:

2AE = 0

Z-1 -> STACK (arg)

(2AE) += F();

X-1 -> STACK (arg)

(2AE) += F()-1;

Y -> STACK (arg)

(2AE) = F()-(2AE)

F():

ac = arg

if(ac <= 0)

 ac -= (-983);

 if(ac >= 0) {

 goto(flag)

 }

 else {

 ac += (-983);

 }

}

ac += 2*arg;

ac += 18;

if(false) {

 flag: (-983) -> ac;

}

ac -> (заменить arg в stack)

return;

Фактическое назначение программы:

```

F(x) {
    Else If(x >= -983 && x <= 0) return (-983);
    Else return return (3*x + 18)
}

Main() {
    x,y,z;
    res = F(y) – (F(z-1) + F(x-1)-1);
}

```

Описание программы

Область представления данных	X, Y, Z, A, B, RES - 16-ти разрядные знаковые числа
Область допустимых значений	-10927 <= X, Z <= 10917 -10928 <= Y <= 10916 -2 ¹⁵ <= RES <= 2 ¹⁵ - 1
Расположение в памяти исходных данных и результатов	Z - 2AB; Y - 2AC; X - 2AD – исходные данные (переменные) A – 68D; B – 68E – исходные данные (константы подпрограммы) RES – 2AE - результат
Адреса первой и последней выполняемой команды	Основная программа: 293 – первая команда 2AA – последняя команда Подпрограмма: 67F – первая команда 68C – последняя команда

ОДЗ (пояснение):

Результат подпрограммы:

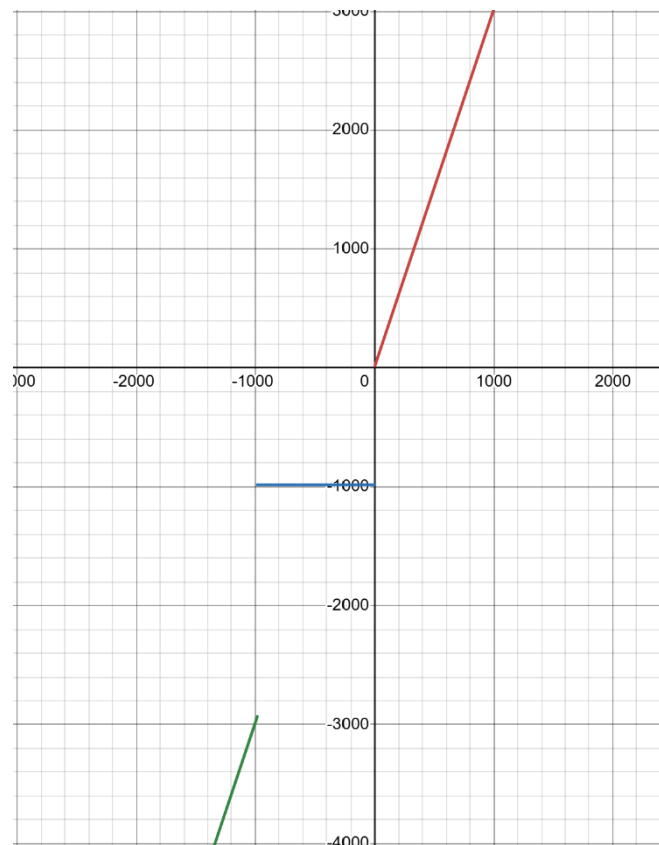
$[-983, 0] \Rightarrow -983$

$[-2^{15}, -984] \vee [1, 2^{15} - 1] \Rightarrow 3a + 18$

Нужно: $F_{\min} \geq -2^{15}$; $F_{\max} \leq 2^{15} - 1 \Rightarrow -10928 \leq a \leq 10916$ (Иначе - переполнение)

Значит: $-10927 \leq X, Z \leq 10917$; $-10928 \leq Y \leq 10916$

Т.к. $F(\dots) \in [-2^{15}, 2^{15} - 1] \Rightarrow \text{RES} = F(\dots) - (F(\dots) + F(\dots) - 1) \in [-2^{15}, 2^{15} - 1]$



Доп

$x = 128(10) = 0080(16)$, $y = -10 = \text{FFF6}(16)$, $z = 1024 = 0400(16)$

$\text{Res} = -983 - (1023 \cdot 3 + 18 + 127 \cdot 3 + 18 - 1) = -4468(10) = \text{EE8C}(16)$

Трассировка (☹)

Выполняемая команда		Содержимое регистров процессора после выполнения команды									Ячейка, содержащее которой изменилось после выполнения команды	
Адрес	Код	IP	CR	AR	DR	SP	BR	AC	PS	NZVC	Адрес	Новый код
293	0200	294	0200	293	0200	000	0293	0000	004	0100		
294	EE19	295	EE19	2AE	0000	000	0019	0000	004	0100	2AE	0000
295	AE15	296	AE15	2AB	0400	000	0015	0400	000	0000		
296	0740	297	0740	296	0740	000	0296	03FF	001	0001		
297	0C00	298	0C00	7FF	03FF	7FF	0297	03FF	001	0001	7FF	03FF
298	D67F	67F	D67F	7FE	0299	7FE	D67F	03FF	001	0001	7FE	0299
67F	AC01	680	AC01	7FF	03FF	7FE	0001	03FF	001	0001		
680	F001	681	F001	680	F001	7FE	0680	03FF	001	0001		
681	F304	686	F304	681	F304	7FE	0004	03FF	001	0001		
686	4C01	687	4C01	7FF	03FF	7FE	0001	07FE	000	0000		
687	4C01	688	4C01	7FF	03FF	7FE	0001	0BFD	000	0000		
688	4E05	689	4E05	68E	0012	7FE	0005	0C0F	000	0000		
689	CE01	68B	CE01	689	068B	7FE	0001	0C0F	000	0000		
68B	EC01	68C	EC01	7FF	0C0F	7FE	0001	0C0F	000	0000	7FF	0C0F
68C	0A00	299	0A00	7FE	0299	7FF	068C	0C0F	000	0000		
299	0800	29A	0800	7FF	0C0F	000	0299	0C0F	000	0000		
29A	4E13	29B	4E13	2AE	0000	000	0013	0C0F	000	0000		
29B	EE12	29C	EE12	2AE	0C0F	000	0012	0C0F	000	0000	2AE	0C0F
29C	AE10	29D	AE10	2AD	0080	000	0010	0080	000	0000		
29D	0740	29E	0740	29D	0740	000	029D	007F	001	0001		
29E	0C00	29F	0C00	7FF	007F	7FF	029E	007F	001	0001	7FF	007F
29F	D67F	67F	D67F	7FE	02A0	7FE	D67F	007F	001	0001	7FE	02A0
67F	AC01	680	AC01	7FF	007F	7FE	0001	007F	001	0001		
680	F001	681	F001	680	F001	7FE	0680	007F	001	0001		
681	F304	686	F304	681	F304	7FE	0004	007F	001	0001		
686	4C01	687	4C01	7FF	007F	7FE	0001	00FE	000	0000		
687	4C01	688	4C01	7FF	007F	7FE	0001	017D	000	0000		
688	4E05	689	4E05	68E	0012	7FE	0005	018F	000	0000		
689	CE01	68B	CE01	689	068B	7FE	0001	018F	000	0000		
68B	EC01	68C	EC01	7FF	018F	7FE	0001	018F	000	0000	7FF	018F
68C	0A00	2A0	0A00	7FE	02A0	7FF	068C	018F	000	0000		
2A0	0800	2A1	0800	7FF	018F	000	02A0	018F	000	0000		
2A1	0740	2A2	0740	2A1	0740	000	02A1	018E	001	0001		
2A2	4E0B	2A3	4E0B	2AE	0C0F	000	000B	0D9D	000	0000		
2A3	EE0A	2A4	EE0A	2AE	0D9D	000	000A	0D9D	000	0000	2AE	0D9D
2A4	AE07	2A5	AE07	2AC	FFF6	000	0007	FFF6	008	1000		
2A5	0C00	2A6	0C00	7FF	FFF6	7FF	02A5	FFF6	008	1000	7FF	FFF6
2A6	D67F	67F	D67F	7FE	02A7	7FE	D67F	FFF6	008	1000	7FE	02A7

67F	AC01	680	AC01	7FF	FFF6	7FE	0001	FFF6	008	1000		
680	F001	681	F001	680	F001	7FE	0680	FFF6	008	1000		
681	F304	682	F304	681	F304	7FE	0681	FFF6	008	1000		
682	6E0A	683	6E0A	68D	FC29	7FE	000A	03CD	001	0001		
683	F201	684	F201	683	F201	7FE	0683	03CD	001	0001		
684	CE05	68A	CE05	684	068A	7FE	0005	03CD	001	0001		
68A	AE02	68B	AE02	68D	FC29	7FE	0002	FC29	009	1001		
68B	EC01	68C	EC01	7FF	FC29	7FE	0001	FC29	009	1001	7FF	FC29
68C	0A00	2A7	0A00	7FE	02A7	7FF	068C	FC29	009	1001		
2A7	0800	2A8	0800	7FF	FC29	000	02A7	FC29	009	1001		
2A8	6E05	2A9	6E05	2AE	0D9D	000	0005	EE8C	009	1001		
2A9	EE04	2AA	EE04	2AE	EE8C	000	0004	EE8C	009	1001	2AE	EE8C
2AA	0100	2AB	0100	2AA	0100	000	02AA	EE8C	009	1001		

Вывод

- Изучил организацию подпрограмм в БЭВМ.
- Узнал о командах вызова подпрограммы и возврата.
- Узнал, как организована передача аргументов в подпрограммы БЭВМ.
- Изучил организацию рекурсивных подпрограмм в БЭВМ.