Министерство науки и высшего образования Российской Федерации

федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

*Факультет Программной инженерии и компьютерной техники*

**Лабораторная работа №4**

Выполнение комплекса программ

Вариант №3107

Группа: P3131

Выполнил: Хайкин О. И.

Проверил:

Перцев Тимофей Сергеевич

Санкт-Петербург

2022г

Оглавление

[Задание 3](#_Toc96339222)

[Выполнение работы 3](#_Toc96339223)

[Исходная программа 3](#_Toc96339224)

[Назначение программы и реализуемые ею функции 4](#_Toc96339225)

[Область представления исходных данных и результата 4](#_Toc96339226)

[Область допустимых значений исходных данных и результата 4](#_Toc96339227)

[Расположение в памяти ЭВМ программы, исходных данных и результата 5](#_Toc96339228)

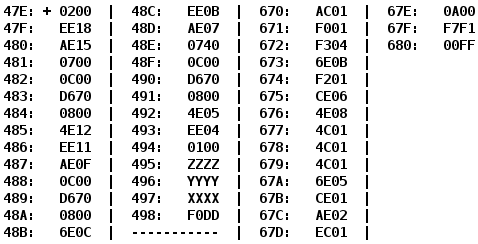
[Адреса первой и последней выполняемых команд 5](#_Toc96339229)

[Таблица трассировки 6](#_Toc96339230)

[Диапазон ячеек памяти для размещения массива 7](#_Toc96339231)

[Вывод 7](#_Toc96339232)

# 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.

# Выполнение работы

## Исходная программа

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Код команды | Мнемоника | Комментарии |
| 47E | 0200 | CLA | Очистка аккумулятора |
| 47F | EE18 | ST (IP+18) | Сохранение текущего значения аккумулятора (0) в ячейку 498 |
| 480 | AE15 | LD (IP+15) | Загрузка значения из ячейки 496 в аккумулятор |
| 481 | 0700 | INC | Инкремент |
| 482 | 0C00 | PUSH | Кладём значение на аккумуляторе в стек |
| 483 | D670 | CALL 670 | Вызываем подпрограмму по адресу 670 |
| 484 | 0800 | POP | Достаём значение с вершины стека в аккумулятор |
| 485 | 4E12 | ADD (IP+12) | Сложить значение на аккумуляторе со значением из ячейки 498 |
| 486 | EE11 | ST (IP+11) | Сохранение текущего значения аккумулятора в ячейку 498 |
| 487 | AE0F | LD (IP+F) | Загрузка значения из ячейки 497 в аккумулятор |
| 488 | 0C00 | PUSH | Кладём значение на аккумуляторе в стек |
| 489 | D670 | CALL 670 | Вызываем подпрограмму по адресу 670 |
| 48A | 0800 | POP | Достаём значение с вершины стека в аккумулятор |
| 48B | 6E0C | SUB (IP+C) | Вычесть из значения на аккумуляторе значение из ячейки 498 |
| 48C | EE0B | ST (IP+B) | Сохранение текущего значения аккумулятора в ячейку 498 |
| 48D | AE07 | LD (IP+7) | Загрузка значения из ячейки 495 в аккумулятор |
| 48E | 0740 | DEC | Декремент |
| 48F | 0C00 | PUSH | Кладём значение на аккумуляторе в стек |
| 490 | D670 | CALL 670 | Вызываем подпрограмму по адресу 670 |
| 491 | 0800 | POP | Достаём значение с вершины стека в аккумулятор |
| 492 | 4E05 | ADD (IP+5) | Сложить значение на аккумуляторе со значением из ячейки 498 |
| 493 | EE04 | ST (IP+4) | Сохранение текущего значения аккумулятора в ячейку 498 |
| 494 | 0100 | HLT | Останов |
| 495 | …. | Число (a) |  |
| 496 | …. | Число (b) |  |
| 497 | …. | Число (c) |  |
| 498 | …. | Число (R) | результат |

|  |  |  |  |
| --- | --- | --- | --- |
| Адрес | Код команды | Мнемоника | Комментарии |
| 670 | AC01 | LD (SP+1) | Загрузка значения из предпоследней ячейки стека в аккумулятор |
| 671 | F001 | BEQ IP+1 | Операция ветвления, совершающая переход на ячейку 673, если (Z==1) |
| 672 | F304 | BPL IP+4 | Операция ветвления, совершающая переход на ячейку 677, если (N==0) |
| 673 | 6E0B | SUB (IP+B) | Вычесть из значения на аккумуляторе значение из ячейки 67F |
| 674 | F201 | BMI IP+1 | Операция ветвления, совершающая переход на ячейку 676, если (N==1) |
| 675 | CE06 | JUMP (IP+6) | Прыжок, переходит к выполнению команды по адресу 67C |
| 676 | 4E08 | ADD (IP+8) | Сложить значение на аккумуляторе со значением из ячейки 67F |
| 677 | 4C01 | ADD (SP+1) | Сложить значение на аккумуляторе со значением из предпоследней ячейки стека |
| 678 | 4C01 | ADD (SP+1) | Сложить значение на аккумуляторе со значением из предпоследней ячейки стека |
| 679 | 4C01 | ADD (SP+1) | Сложить значение на аккумуляторе со значением из предпоследней ячейки стека |
| 67A | 6E05 | SUB (IP+5) | Вычесть из значения на аккумуляторе значение из ячейки 680 |
| 67B | CE01 | JUMP (IP+1) | Прыжок, переходит к выполнению команды по адресу 67D |
| 67C | AE02 | LD (IP+2) | Загрузка значения из ячейки 67F в аккумулятор |
| 67D | EC01 | ST (SP+1) | Сохранение текущего значения аккумулятора в предпоследнюю ячейку стека |
| 67E | 0A00 | RET | Возвращение из подпрограммы к основной программе |
| 67F | F7F1 | число (d) | -30705 |
| 680 | 00FF | Число (e) | 255 |

## Назначение программы и реализуемые ею функции

Назначение: высчитывание результата по формуле R=f(c)-f(b+1)+f(a-1), где a, b и с – числа, а f(x) определена как:

, где d и e – определённые числа.

## Область представления исходных данных и результата

Результат (498): знаковое 16-разрядное число

Исходные числа (495-497): знаковые 16-разрядные числа

Константы (67F-680): знаковые 16-разрядные числа

## Область допустимых значений исходных данных и результата

Результат: -215 ≤ R ≤ 215-1

ОДЗ исходных данных:

1) если c,(b+1),(a-1)≠0:

R=4(c+b+1+a-1)-3e

R=4(a+b+c)-3e

R=4(a+b+c)-765

-215 ≤ 4(a+b+c)-765 ≤ 215-1

-8000 ≤ (a+b+c) ≤ 8383

-2666 ≤ ≤ 2794

a,b,c∊[-2666, 2794] (гарантирует отсутствие переполнения у результата)

2) если c=0:

R=4(a+b)-2e+d

R=4(a+b)-510-30705

R=4(a+b)-31215

-215 ≤ 4(a+b)-31215 ≤ 215-1

-388 ≤ (a+b) ≤ 15995

-194 ≤ ≤ 7997

a,b∊[-194, 7997] (a≠1, b≠-1)

3) если b=-1:

R=4(a+c-1)-2e+d

...

-388 ≤ (a+c-1) ≤ 15995

-387 ≤ (a+c) ≤ 15996

-193 ≤ ≤ 7998

a,c∊[-193, 7998] (a≠1, c≠0)

4) если a=1:

R=4(b+c+1)-2e+s

...

-388 ≤ (b+c+1) ≤ 15995

-389 ≤ (b+c) ≤ 15994

-194 ≤ ≤ 7997

b,c∊[-194, 7997] (b≠-1, c≠0)

5) a=1; b=-1:

R=4c-e+2d

R=4c-255-30705\*2

R=4c-61665

…

7225 ≤ c ≤ 23608

НО: при вычислении f(c) не должно случиться переполнения, т.е. -215 ≤ 4c ≤ 215-1, т.е. в итоге: 7225 ≤ c ≤ 8191; c≠0

6) a=1; c=0

R=4(b+1)-e+2d

R=4(b+1)-255-30705\*2

R=4(b+1)-61665

…

7225 ≤ (b+1) ≤ 23608

НО: при вычислении f(b+1) не должно случиться переполнения, т.е. -215 ≤ 4(b+1) ≤ 215-1, т.е: 7225 ≤ (b+1) ≤ 8191;

7224 ≤ b ≤ 8190; b≠-1

7) b=-1; c=0

R=4(a-1)-e+2d

R=4(a-1)-255-30705\*2

R=4(a-1)-61665

…

7225 ≤ (a-1) ≤ 23608

НО: при вычислении f(a-1) не должно случиться переполнения, т.е. -215 ≤ 4(a-1) ≤ 215-1, т.е: 7225 ≤ (a-1) ≤ 8191;

7226 ≤ a ≤ 8192; a≠1

8) a=1; b=-1; c=0

R=3d

R=-92115; переполнение.

Итоговые варианты ОДЗ:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | c,(b+1),(a-1)≠0 | c=0 | b=-1 | a=1 |
| a: | [-2666, 1)(1, 2794] | [-194, 1)(1, 7997] | [-193, 1)(1, 7998] | {1} |
| b: | [-2666, -1)(-1, 2794] | [-194, -1)(-1, 7997] | {-1} | [-194, -1)(-1, 7997] |
| c: | [-2666, 0)(0, 2794] | {0} | [-193, 0)(0, 7998] | [-194, 0)(0, 7997] |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | a=1; b=-1 | a=1; c=0 | b=-1; c=0 | a=1; b=-1; c=0 |
| a: | {1} | {1} | [7226, 8192] | Не подходит |
| b: | {-1} | [7224, 8190] | {-1} |
| c: | [7225, 8191] | {0} | {0} |

## Расположение в памяти ЭВМ программы, исходных данных и результата

**Исходные данные:**

495-497 – исходные числа a,b,c

67F, 680 – константы d,e

**Результат:**

498

**Команды:**

47E-494; 670-67E

## Адреса первой и последней выполняемых команд

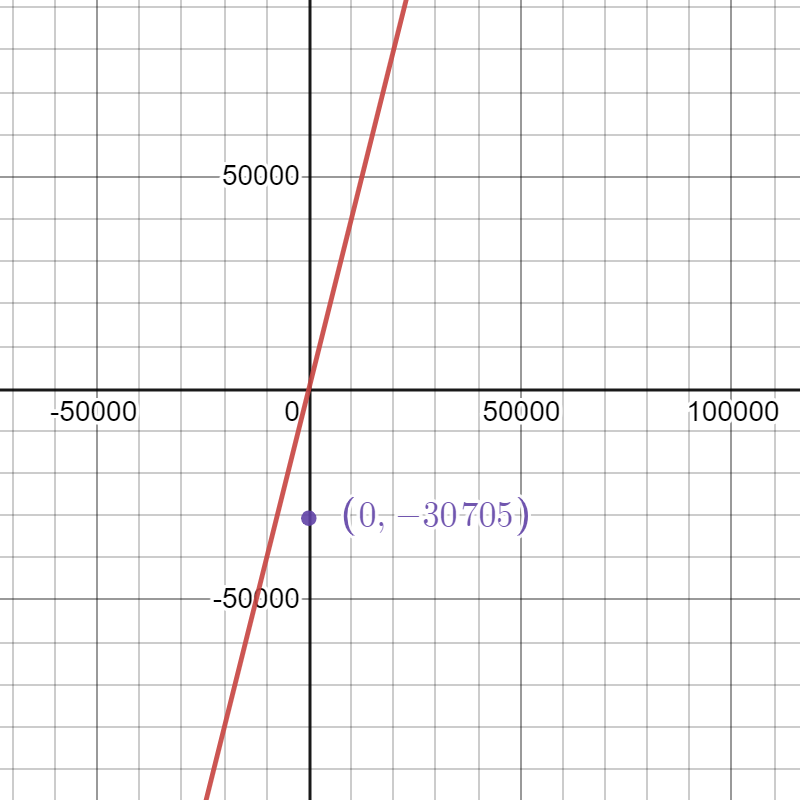
47E – первая

494 – последняя

## Таблица трассировки

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Выполняемая команда | | Содержимое регистров после выполнения команды | | | | | | | | Ячейка, содержимое которой изменилось после выполнения команды | |
| Адрес | Код | IP | CR | AR | DR | SP | BR | AC | NZVC | Адрес | Новый код |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |

## График функции



# Вывод

В ходе выполнения четвёртой лабораторной работы я изучил способы связей между командными модулями и команды обращения к подпрограмме в БЭВМ