Санкт-Петербургский Национальный Исследовательский

Университет ИТМО

Факультет программной инженерии и компьютерной техники

Основы профессиональной деятельности

Лабораторная работа №4

Вариант 1234

Выполнил:

студент группы P3231

Воробьев Кирилл Олегович

Преподаватель:

Блохина Елена Николаевна

г. Санкт-Петербург

2021 г.

**Задание:**

Изображение выглядит как стол

Автоматически созданное описаниеПо 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.

**Текст исходной программы:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Адрес** | **Код команды** | **Мнемоника** | **Комментарии** |
| 287 | + 0200 | CLA | Очистка аккумулятора |
| 288 | EE18 | ST IP+24 | Сохранение AC → 2A1  Относительная адресация (IP+24) |
| 289 | AE14 | LD IP+20 | Загрузка Z → AC  Относительная адресация (IP+20) |
| 28A | 0700 | INC | Инкремент AC + 1 → AC |
| 28B | 0C00 | PUSH | Запись в стек AC → -(SP) |
| 28C | D682 | CALL 682 | Вызов подпрограммы  SP – 1 → SP, IP → (SP), 682 → IP |
| 28D | 0800 | POP | Чтение из стека (SP)+ → AC |
| 28E | 0740 | DEC | Декремент AC - 1 → AC |
| 28F | 6E11 | SUB IP+17 | Вычитание AC – 2A1 → AC  Относительная адресация (IP+17) |
| 290 | EE10 | ST IP+16 | Сохранение AC → 2A1  Относительная адресация (IP+16) |
| 291 | AE0D | LD IP+13 | Загрузка Y → AC  Относительная адресация (IP+13) |
| 292 | 0C00 | PUSH | Запись в стек AC → -(SP) |
| 293 | D682 | CALL 682 | Вызов подпрограммы  SP – 1 → SP, IP → (SP), 682 → IP |
| 294 | 0800 | POP | Чтение из стека (SP)+ → AC |
| 295 | 6E0B | SUB IP+11 | Вычитание AC – 2A1 → AC  Относительная адресация (IP+11) |
| 296 | EE0A | ST IP+10 | Сохранение AC → 2A1  Относительная адресация (IP+10) |
| 297 | AE08 | LD IP+8 | Загрузка X → AC  Относительная адресация (IP+8) |
| 298 | 0C00 | PUSH | Запись в стек AC → -(SP) |
| 299 | D682 | CALL 682 | Вызов подпрограммы  SP – 1 → SP, IP → (SP), 682 → IP |
| 29A | 0800 | POP | Чтение из стека (SP)+ → AC |
| 29B | 6E05 | SUB IP+5 | Вычитание AC – 2A1 → AC  Относительная адресация (IP+5) |
| 29C | EE04 | ST IP+4 | Сохранение AC → 2A1  Относительная адресация (IP+4) |
| 29D | 0100 | HLT | Отключение ТГ, переход в пультовый режим |

**Текст подпрограммы:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Адрес** | **Код команды** | **Мнемоника** | **Комментарии** |
| 682 | AC01 | LD &1 | Загрузка в аккумулятор 1-го элемента из стека |
| 683 | F303 | BPL +3 | Переход, если плюс IF N==0 THEN IP+3+1 → IP |
| 684 | 7E08 | CMP IP+8 | Сравнение. Установить флаги по результату АС-68D |
| 685 | F201 | BMI IP+1 | Переход, если плюс IF N==1 THEN IP+1+1 → IP |
| 686 | CE03 | JUMP IP+3 | Переход 68A → IP  Относительная адресация (IP+3) |
| 687 | 4C01 | ADD &1 | Сложение аккумулятора и 1-го элемента стека |
| 688 | 6E05 | SUB IP+5 | Вычитание AC – 68E → AC  Относительная адресация (IP+5) |
| 689 | CE01 | JUMP IP+1 | Переход 68B → IP  Относительная адресация (IP+1) |
| 68A | AE02 | LD IP+2 | Загрузка 68D → AC  Относительная адресация (IP+2) |
| 68B | EC01 | ST &1 | Сохранение значения из аккумулятора в 1-й элемент стека |
| 68C | 0A00 | RET | Возврат из подпрограммы (SP)+ → IP |

**Описание программы:**

Программа вычисляет значение по формуле:

**Описание подпрограммы:**

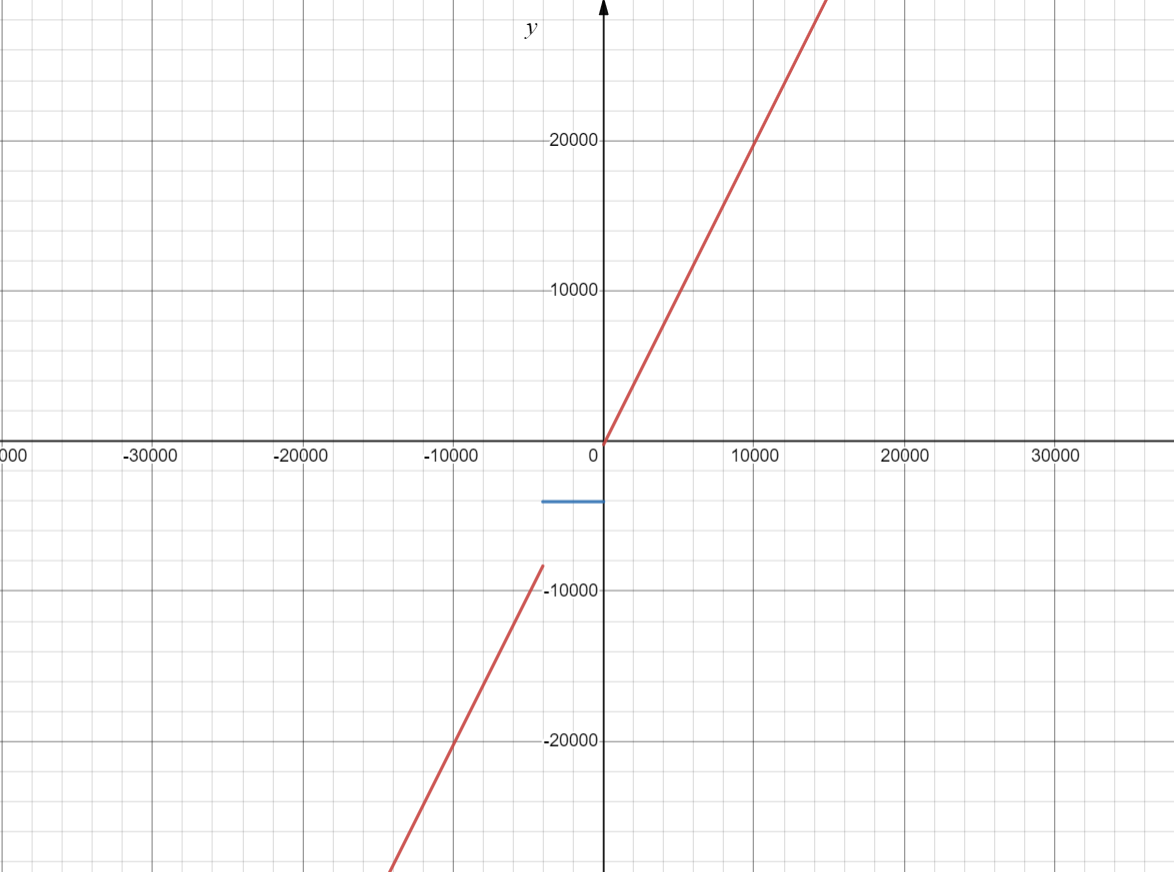
Программа предназначена для вычисления функции:

, где

**Описание программного комплекса:**

Программный комплекс вычисляет значение по формуле:

**График функции:**

****

**Область представления:**

X, Y, Z – знаковые 16-ти разрядные числа

Результат R – знаковое 16-ти разрядное число

Константы A и B – знаковые 16-ти разрядные числа

**Область допустимых значений:**

1. Найдем значение
2. Найдем значение

Т. к. у нас сначала в аккумуляторе окажется сумма k+k, а только потом 2k-243, то:

1. Т. к. функции передается инкрементированное значение Z, а результат функции декрементируется, то:

.

1. Так как изначально в R мы помещаем (F(Z+1)-1), то

Если , то . Т. к. далее в ячейку будет сохраняться F(Y) – (F(Z+1)-1), то => .

Если , то . Т. к. далее в ячейку будет сохраняться F(Y) – (F(Z+1)-1), то => .

Если , то =>

.

Если , то =>

.

**Расположение в памяти ЭВМ программы, исходных данных и результатов**

Расположение программы : 287-29D

Расположение аргументов вычисляемой функции : 29E-2A0

Ячейка для хранения результата : 2A1

Расположение подпрограммы : 682 – 68C

Расположение константы A, используемой подпрограммой : 68D

Расположение константы B, используемой подпрограммой : 68E

**Адрес первой выполняемой команды программы:** 287

**Адрес последней выполняемой команды программы:** 29D

**Адрес первой выполняемой команды подпрограммы:** 682

**Адрес последней выполняемой команды подпрограммы:** 68С

**Данные для трассировки:**

X = 8500 = =

Y = -10900 = =

Z = -3999= =

Числа представлены в 10-ой системе счисления.

**Трассировка:**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Выполняемая команда** | | **Содержание 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 |
| 287 | 0200 | 288 | 0200 | 287 | 0200 | 000 | 0287 | 0000 | 0100 |  |  |
| 288 | EE18 | 289 | EE18 | 2A1 | 0000 | 000 | 0018 | 0000 | 0100 | 2A1 | 0000 |
| 289 | AE14 | 28A | AE14 | 29E | F061 | 000 | 0014 | F061 | 1000 |  |  |
| 28A | 0700 | 28B | 0700 | 28A | 0700 | 000 | 028A | F062 | 1000 |  |  |
| 28B | 0C00 | 28C | 0C00 | 7FF | F062 | 7FF | 028B | F062 | 1000 | 7FF | F062 |
| 28C | D682 | 682 | D682 | 7FE | 028D | 7FE | D682 | F062 | 1000 | 7FE | 028D |
| 682 | AC01 | 683 | AC01 | 7FF | F062 | 7FE | 0001 | F062 | 1000 |  |  |
| 683 | F303 | 684 | F303 | 683 | F303 | 7FE | 0683 | F062 | 1000 |  |  |
| 684 | 7E08 | 685 | 7E08 | 68D | F036 | 7FE | 0008 | F062 | 0001 |  |  |
| 685 | F201 | 686 | F201 | 685 | F201 | 7FE | 0685 | F062 | 0001 |  |  |
| 686 | CE03 | 68A | CE03 | 686 | 068A | 7FE | 0003 | F062 | 0001 |  |  |
| 68A | AE02 | 68B | AE02 | 68D | F036 | 7FE | 0002 | F036 | 1001 |  |  |
| 68B | EC01 | 68C | EC01 | 7FF | F036 | 7FE | 0001 | F036 | 1001 | 7FF | F036 |
| 68C | 0A00 | 28D | 0A00 | 7FE | 028D | 7FF | 068C | F036 | 1001 |  |  |
| 28D | 0800 | 28E | 0800 | 7FF | F036 | 000 | 028D | F036 | 1001 |  |  |
| 28E | 0740 | 28F | 0740 | 28E | 0740 | 000 | 028E | F035 | 1001 |  |  |
| 28F | 6E11 | 290 | 6E11 | 2A1 | 0000 | 000 | 0011 | F035 | 1001 |  |  |
| 290 | EE10 | 291 | EE10 | 2A1 | F035 | 000 | 0010 | F035 | 1001 | 2A1 | F035 |
| 291 | AE0D | 292 | AE0D | 29F | D56C | 000 | 000D | D56C | 1001 |  |  |
| 292 | 0C00 | 293 | 0C00 | 7FF | D56C | 7FF | 0292 | D56C | 1001 | 7FF | D56C |
| 293 | D682 | 682 | D682 | 7FE | 0294 | 7FE | D682 | D56C | 1001 | 7FE | 0294 |
| 682 | AC01 | 683 | AC01 | 7FF | D56C | 7FE | 0001 | D56C | 1001 |  |  |
| 683 | F303 | 684 | F303 | 683 | F303 | 7FE | 0683 | D56C | 1001 |  |  |
| 684 | 7E08 | 685 | 7E08 | 68D | F036 | 7FE | 0008 | D56C | 1000 |  |  |
| 685 | F201 | 687 | F201 | 685 | F201 | 7FE | 0001 | D56C | 1000 |  |  |
| 687 | 4C01 | 688 | 4C01 | 7FF | D56C | 7FE | 0001 | AAD8 | 1001 |  |  |
| 688 | 6E05 | 689 | 6E05 | 68E | 00F3 | 7FE | 0005 | A9E5 | 1001 |  |  |
| 689 | CE01 | 68B | CE01 | 689 | 068B | 7FE | 0001 | A9E5 | 1001 |  |  |
| 68B | EC01 | 68C | EC01 | 7FF | A9E5 | 7FE | 0001 | A9E5 | 1001 | 7FF | A9E5 |
| 68C | 0A00 | 294 | 0A00 | 7FE | 0294 | 7FF | 068C | A9E5 | 1001 |  |  |
| 294 | 0800 | 295 | 0800 | 7FF | A9E5 | 000 | 0294 | A9E5 | 1001 |  |  |
| 295 | 6E0B | 296 | 6E0B | 2A1 | F035 | 000 | 000B | B9B0 | 1000 |  |  |
| 296 | EE0A | 297 | EE0A | 2A1 | B9B0 | 000 | 000A | B9B0 | 1000 | 2A1 | B9B0 |
| 297 | AE08 | 298 | AE08 | 2A0 | 2134 | 000 | 0008 | 2134 | 0000 |  |  |
| 298 | 0C00 | 299 | 0C00 | 7FF | 2134 | 7FF | 0298 | 2134 | 0000 | 7FF | 2134 |
| 299 | D682 | 682 | D682 | 7FE | 029A | 7FE | D682 | 2134 | 0000 | 7FE | 029A |
| 682 | AC01 | 683 | AC01 | 7FF | 2134 | 7FE | 0001 | 2134 | 0000 |  |  |
| 683 | F303 | 687 | F303 | 683 | F303 | 7FE | 0003 | 2134 | 0000 |  |  |
| 687 | 4C01 | 688 | 4C01 | 7FF | 2134 | 7FE | 0001 | 4268 | 0000 |  |  |
| 688 | 6E05 | 689 | 6E05 | 68E | 00F3 | 7FE | 0005 | 4175 | 0001 |  |  |
| 689 | CE01 | 68B | CE01 | 689 | 068B | 7FE | 0001 | 4175 | 0001 |  |  |
| 68B | EC01 | 68C | EC01 | 7FF | 4175 | 7FE | 0001 | 4175 | 0001 | 7FF | 4175 |
| 68C | 0A00 | 29A | 0A00 | 7FE | 029A | 7FF | 068C | 4175 | 0001 |  |  |
| 29A | 0800 | 29B | 0800 | 7FF | 4175 | 000 | 029A | 4175 | 0001 |  |  |
| 29B | 6E05 | 29C | 6E05 | 2A1 | B9B0 | 000 | 0005 | 87C5 | 1010 |  |  |
| 29C | EE04 | 29D | EE04 | 2A1 | 87C5 | 000 | 0004 | 87C5 | 1010 | 2A1 | 87C5 |
| 29D | 0100 | 29E | 0100 | 29D | 0100 | 000 | 029D | 87C5 | 1010 |  |  |

**Результат работы программы:**

F(Z+1)-1 = F(-3998)-1 = -4042 – 1 = -4043 =

F(Y) = F(-10900) = -10900\*2 – 243 = -22043 =

F(X) = F(8500) = 8500\*2 – 243 = 16757 =

*,* соответственно у нас произошло переполнение, а оно произошло из-за того, что введенные данные (данные для трассировки) не соответствовали условиям ОДЗ (п.6: «Если , то => », в нашем же случае F(Y) – (F(Z+1) – 1) = -18000 => X не может принимать значение равное 8500), поэтому результатом программы будет являться число 87С5 (но т.к. у нас – знаковое число, то ответ неверный).

**Вывод:** При выполнении данной лабораторной работы я познакомился с подпрограммами в БЭВМ, узнал про стек, понял принцип его работы, а также ознакомился с новым типом команд с адресацией относительно указателя стека.