МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

Кафедра компьютерной математики и программирования

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| доцент, канд.тех.наук |  |  |  | А.А. Попов |
| должность |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №5 |
| Условные и безусловные переходы. Обработка массивов. |
| по курсу: АРХИТЕКТУРА ЭВМ И СИСТЕМ |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4936 |  |  |  | Е.А.Цыганкова |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург

2021

Практическая часть.

Практическая часть работы включает выполнение следующих действий:

а) в соответствии с индивидуальным заданием составление двух программ обработки массивов, содержащих не менее 10 целых чисел; одна программа для обращения к элементам массивов должна использовать косвенные способы адресации, а другая - адресацию с индексированием; во второй программе для организации цикла необходимо использовать команду ACB, а в первой программе использование команды организации цикла ACB запрещено;

б) формирование и занесение в память исходных значений массивов, определение и занесение в память и РОНы необходимых вспомогательных данных;

в) запись программ обработки массивов данных, хранящихся в памяти;

г) выполнение программ;

д) контроль результатов работы программ.

Правильность разработки и выполнения программ арифметико-логической обработки данных контролируется путем ручной трассировки заданных алгоритмов с последующим сравнением результатов работы программ с результатами ручной трассировки.

Варианты заданий.

Тексты заданий приведены в табл.2.7.

Допустимые способы проверки конца массива (организации цикла) приведены в табл.2.8.

Способы адресации, используемые для вычисления базового адреса при адресации с индексированием, приведены в табл.2.9.

Размеры элементов массива определены в табл.2.10..

Исходные данные должны располагаться в памяти, начиная с адреса, определяемого выражением: Aдр = NВ \* NГ = 4\*6 = 24(10) = 18(16)

Начальный адрес размещения программ определяется выражением: Aдр = ( NВ \* NГ ) + 60 = (4\*6) + 60 = 84(10) = 54(16)

Вспомогательные значения, необходимые для выполнения программ, должны быть расположены в памяти, начиная с адреса, определяемого выражением: Aдр = ( NВ \* NГ ) + 300 = (4\*6) + 300 = 324(10) = 144(16)

Порядок выполнения работы.

А. В процессе самостоятельной работы

1. Выбрать исходные данные в соответствии с номером варианта.

2. Составить алгоритмы программ для решения поставленной задачи.

3. Составить программы вычислений в мнемонических и машинных кодах.

4. Составить карту распределения памяти под команды и данные.

5. Произвести ручную трассировку программ с использованием заданных исходных данных, при этом в таблице трассировки должны быть отражены значения информации в ячейках памяти и используемых регистрах.

6. Оформить отчет по лабораторной работе.

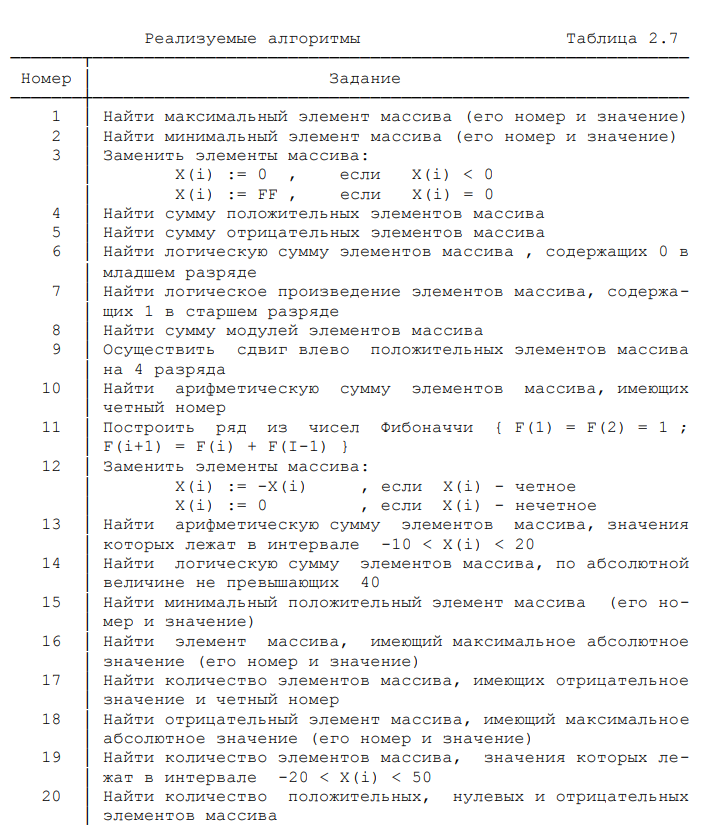
Б. В учебной лаборатории

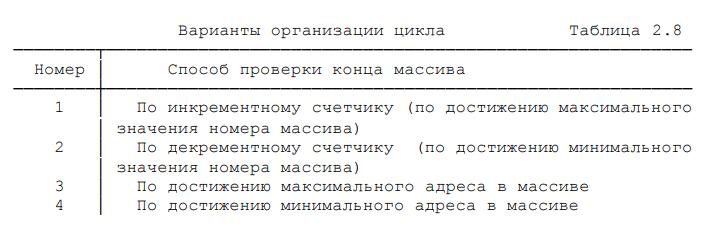
1. Записать в память и РОН исходные данные, программы и дополнительную информацию, необходимую для реализации заданных алгоритмов и различных способов адресации.

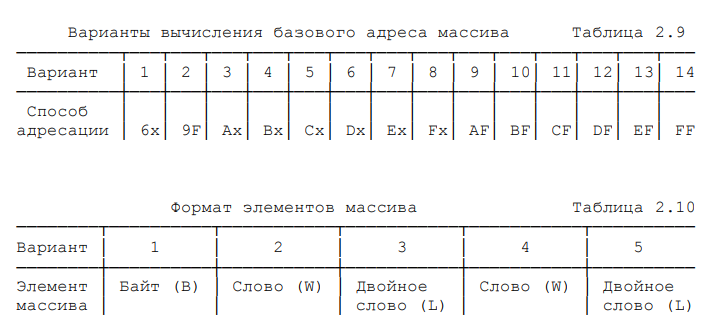
2. Выполнить программы, занесенные в память.

3. Проверить результаты выполнения программ, сравнивая их с результатами ручной трассировки алгоритма.

**NB = 4, NГ = 6.**





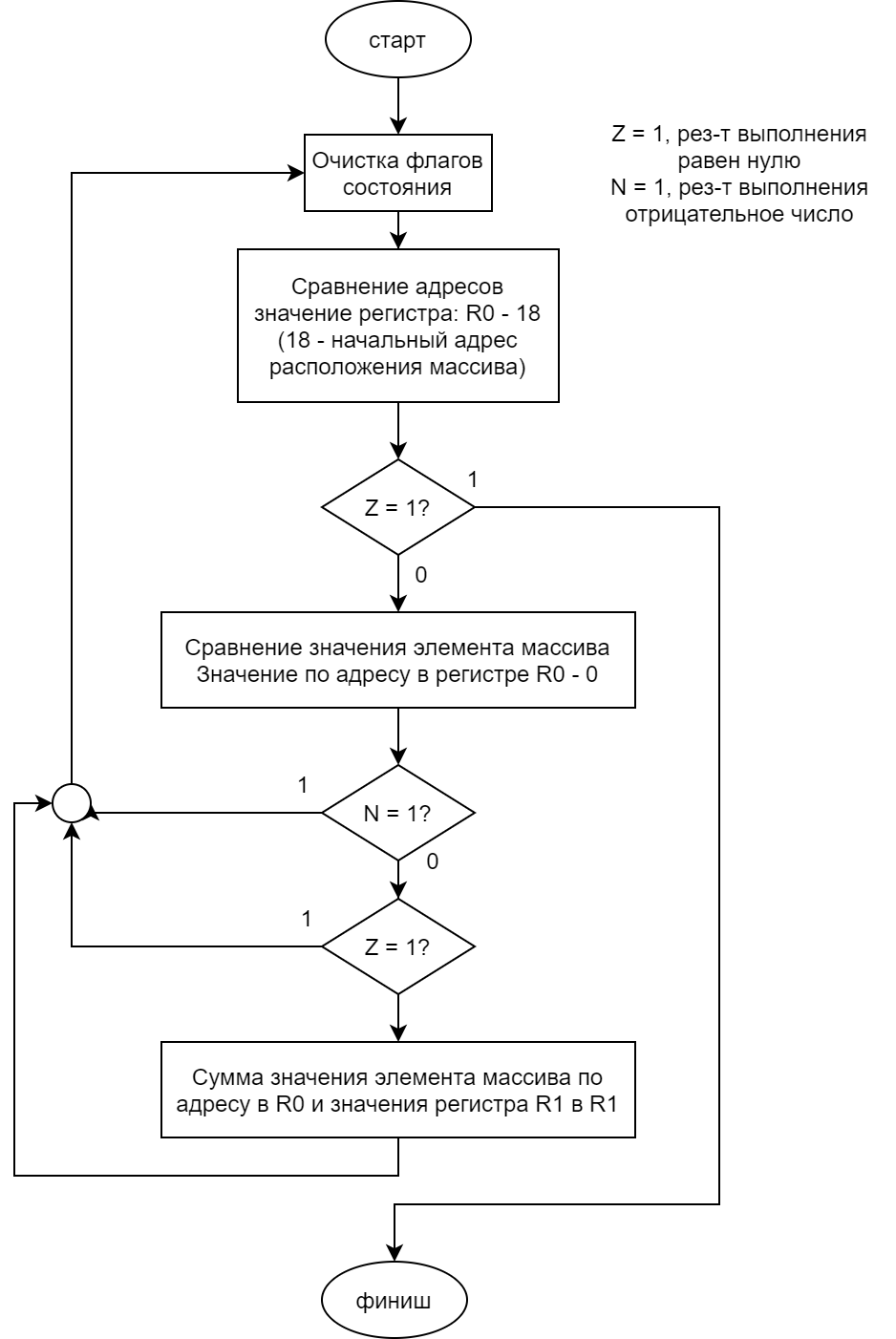


Карта распределения памяти

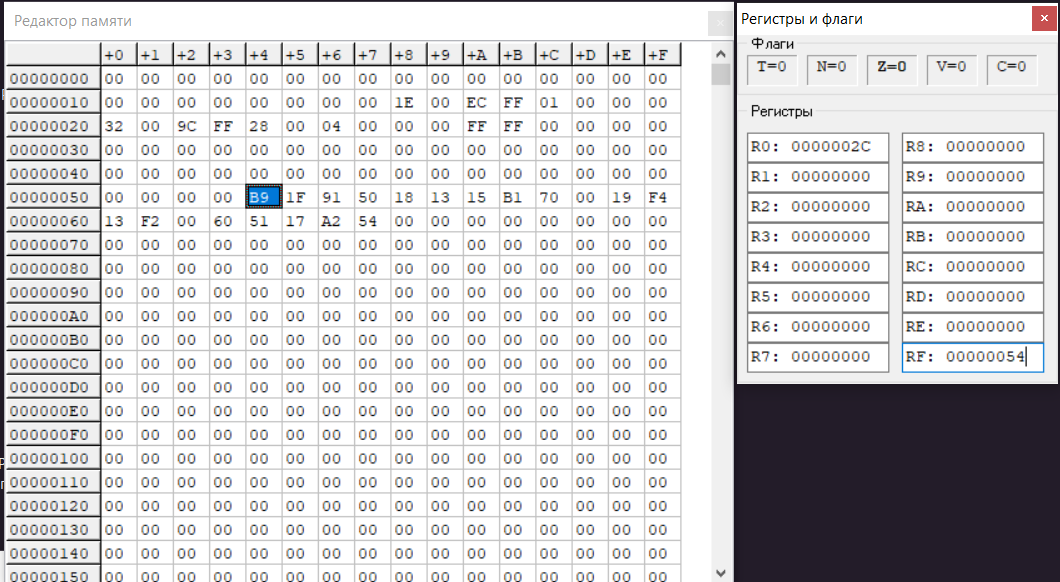
|  |  |  |  |
| --- | --- | --- | --- |
| Число | Десятичное число | Шестнадцатеричный код | Адрес загрузки |
| Х1  Х2  Х3  Х4  Х5  Х6  Х7  Х8  Х9  Х10 | 30  -20  1  0  50  -100  40  4  0  -1  Aдр = ( NВ \* NГ ) + 60 = (4\*6) + 60 = 84(10)  Aдр = ( NВ \* NГ ) + 300 = (4\*6) + 300 = 324(10) | 001E  FFEC  0001  0000  0032  FF9C  0028  0004  0000  FFFF  Текст программы 1  Текст программы 2  Вспомогательные значения | 18  1A  1C  1E  20  22  24  26  28  2A  54  54  144 |

А) Первая программа

Схема алгоритма



Начальное состояние памяти и регистров и флагов состояния:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Оператор | Адрес | Шестнадцатеричный код | Мнемокод | Комментарии |
| 0  1  2  3  4  5  6  7 | 54  56  59  5B  5E  60  62  65 | B9 1F  91 50 18  13 15  B1 70 00  19 F4  13 F2  A0 60 51  17 A2 54  00 | BICPSW  CMPB R0, K# 18  BEQL  CMPW -2(R0), K# 0  BLSS  BEQL  ADDW2 (R0), R1  JMP R0  HALT | Очистка флагов  R0 - 18  Завершение, если выход за границы адреса  Значение по адресу в R0 - 0  Переход на начало программы, если значение отрицательное  Переход на начало программы, если значение равно нулю  Значение по адресу в R0 + значение в R1  Переход по адресу 54  ОСТАНОВ |

Таблица трассировки

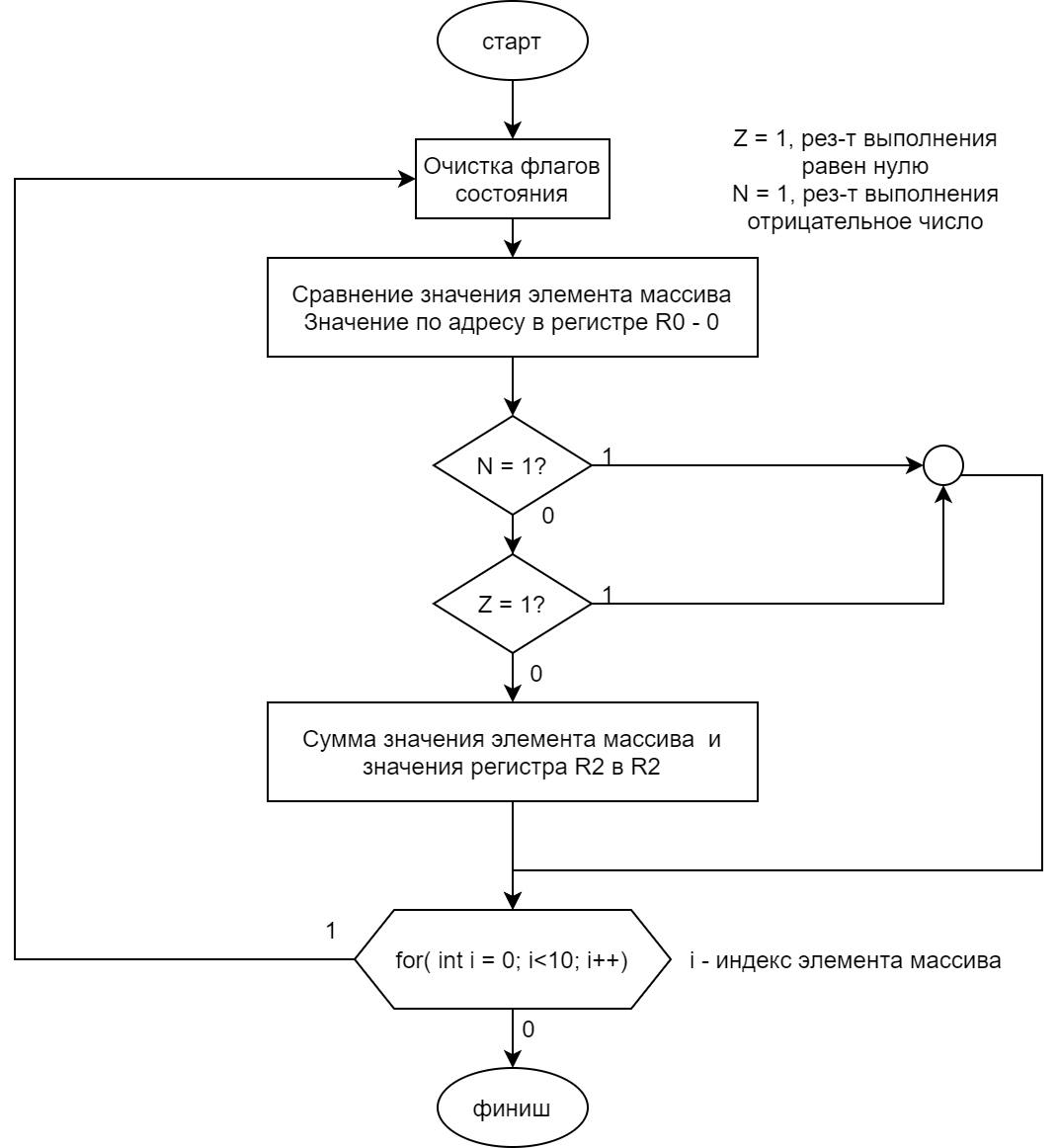
Массив: 30, -20, 1, 0, 50, -100, 40, 4, 0, -1 (ответ: 125(10) = 7D(16)) Результат в регистре R1



|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Номер шага | Номер регистра | Расчетные значения | | Значения, полученные в лаборатории | Флаги | Расчетные значения | | Значения, полученные в лаборатории |
| До выполнения команды | После выполнения команды | До | После |
| 0  1  2  3  4  0  1  2  3  4  5  0  1  2  3  4  5  6  7  …  0  1  2 | RF  R0  RF  RF  R0  RF  RF  RF  R0  RF  RF  R1  RF  …  RF  R0  R1 | 59  2C  5E  59  2A  5E  60  59  28  5E  60  0000 0000  65  …  59  18  5F | 5B  2A  54  5B  28  60  54  5B  26  60  62  0000 0004  54  …  70  18  7D | 5B(нет зав)  2A  54  5B  28  60  54  5B  26  60  62  0000 0004  54  …  70 (завершение)  18 (конечный адрес)  7D | Все  N  N  N  N  N  Z  Z  N  N  Z  …  Все  Z | 0  0  0  1  0  0  0  1  0  0  0  …  0  0 | 0  0  1  0  0  0  1  0  0  0  0  …  0  1 | 0  0  1  0  0  0  1  0  0  0  0  …  0  1 |

Б) Вторая программа

Схема алгоритма



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Оператор | Адрес | Шестнадцатеричный код | Мнемокод | Комментарии |
| 0  1  2  3  4  5 | 54  56  5B  5D  5F  64  6A | B9 1F  B1 40 B1 00 00  19 F7  13 F5  A0 40 B1 00 52  3D 53 01 50 EA FF  00 | BICPSW  CMPW @ 00 (R0)[R1]  BLSS  BEQL  ADDW2 @ 00 (R0)[R1], R2  ACBW  HALT | Очистка флагов  Значение элемента массива - 0  Переход на шаг 5 программы, если значение отрицательное  Переход на шаг 5 программы, если значение равно нулю  Значение элемента массива + значение в R2  Переход в начало программы, если индекс находится в пределах массива  ОСТАНОВ |

Начальное состояние памяти и регистров и флагов состояния:

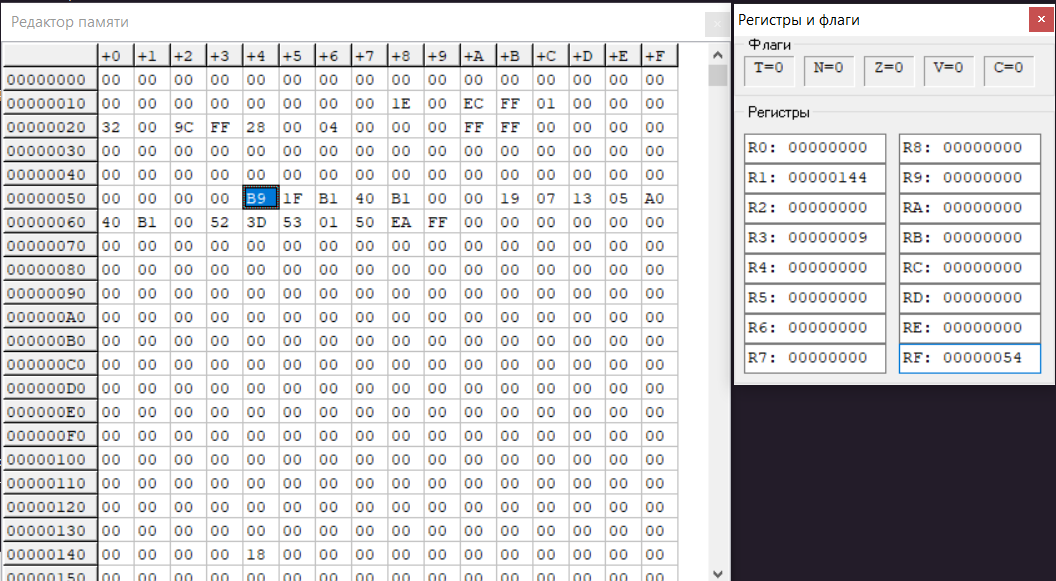


Таблица трассировки

Массив: 30, -20, 1, 0, 50, -100, 40, 4, 0, -1 (ответ: 125(10) = 7D(16)) Результат в регистре R2

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Номер шага | Номер регистра | Расчетные значения | | Значения, полученные в лаборатории | Флаги | Расчетные значения | | Значения, полученные в лаборатории |
| До выполнения команды | После выполнения команды | До | После |
| 0  1  2  3  4  5  0  1  2  5  0  1  2  3  4  5  0  1  2  3  …  5 | RF  RF  R2  R0  RF  RF  R0  RF  RF  RF  R2  R0  RF  RF  RF  …  RF  R0  R2 | 5B  5D  0000 0000  0000 0000  64  5B  0000 0001  64  5B  5D  0000 001E  0000 0002  64  5B  5D  …  64  0000 0009  0000 007D | 5D  5F  0000 001E  0000 0001  54  64  0000 0002  54  5D  5F  0000 001F  0000 0003  54  5D  54  …  6A  0000 000A  0000 007D | 5D(нет зав)  5F  0000 001E  0000 0001  54  64  0000 0002  54  5D  5F  0000 001E  0000 0003  54  5D  54  …  6A  0000 000A  0000 007D | Все  N  Все  N  Все  N,Z  Все  Z  … | 0  0  0  0  0  0  0  0  … | 0  0  0  1  0  0  0  1  … | 0  0  0  1  0  0  0  1  … |