Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и технологий

Высшая школа интеллектуальных систем и суперкомпьютерных технологий

**Отчёт по лабораторной работе № 2**

Дисциплина: Низкоуровневое программирование

Тема: EDSAC

Вариант №1

Выполнила

студентка гр. 3530901/10003 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.Р. Саушкина

(подпись)

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д.А. Коренев

(подпись)

“ ” 2022 г.

Санкт-Петербург

2022

Оглавление

Оглавление

[Техническое задание 3](#_Toc118378728)

[Метод решения 4](#_Toc118378729)

[Программа Orders1 5](#_Toc118378730)

[Работа программы Orders1 9](#_Toc118378731)

[Программа Orders2 10](#_Toc118378732)

[Работа программы Orders2 13](#_Toc118378733)

# Техническое задание

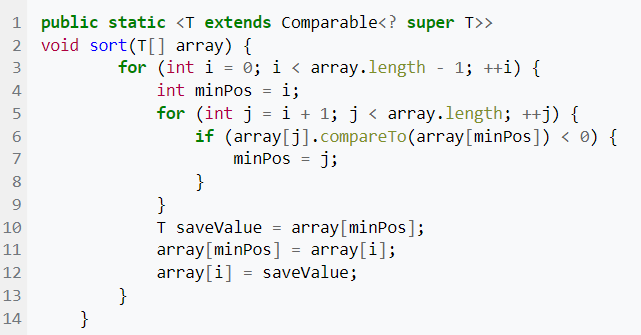
Написать программу для EDSAC, реализующую сортировку выбором массива чисел in-place.

# Метод решения

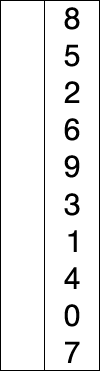
Шаги алгоритма:

1. находим номер минимального значения в текущем списке
2. производим обмен этого значения со значением первой неотсортированной позиции (обмен не нужен, если минимальный элемент уже находится на данной позиции)
3. теперь сортируем хвост списка, исключив из рассмотрения уже отсортированные элементы

Пример реализации сортировки выбором на Java:



Наглядный пример сортировки массива выбором.



# Программа Orders1

[Сортировка выбором массива чисел in-place, ответ в ячейках 158-165]

[31] T 166 S [Завершение программы]

[32] Z 0 S

[33] T 0 S [Обнуление аккумулятора]

[34] A 148 S [Помещаем в аккумулятор единицу]

[35] T 1 S [Записываем начальное значение внешнего счетчика]

[36] T 6 S [Обнуляем значение ячейки для дальнейшего использования]

[37] A 97 S [Помещаем в аккумулятор изначальный вид команды l1]

[38] T 8 S [Сохраняем изначальные виды команд l1]

[39] A 109 S [Помещаем в аккумулятор изначальный вид команды k1]

[40] T 7 S [Сохраняем изначальный вид команды k1]

[41] [Начало внешнего цикла] A 1 S [Загрузка в аккумулятор счетчика внешнего цикла]

[42] U 2 S [Выгрузка значения внешнего счетчика в ячейку внутреннего счетчика]

[43] S 149 S [Вычитаем длину массива, проверяем не подошли ли к последнему элементу]

[44] E 131 S [Если аккумулятор после вычитания >=0, то выходим из программы]

[45] T 0 S [Обнуление аккумулятора]

[46] X 0 S

[47] X 0 S

[48] A 1 S [Загрузка в аккумулятор счетчика внешнего цикла]

[49] S 148 S [Вычитание из значения счетчика единицы - получение смещения адреса текущего заменяемого элемента]

[50] L 0 L [Сдвиг влево на 1]

[51] A 150 S [В аккумуляторе адрес того элемента, на место которого придет наименьший, найденный во внутреннем цикле]

[52] A 144 S [Добавляем в аккумулятор опкод команды S, теперь в аккумуляторе полноценная команда r2]

[53] U 63 S [Помещаем сформированную команду r2 в свою ячейку]

[54] S 144 S [Убираем из аккумулятора опкод команды S чтобы заменить его на опкод команды A]

[55] A 147 S [Добавляем к адресу команды в аккумуляторе единицу]

[56] A 145 S [Добавляем к аккумулятор опкод команды A]

[57] T 62 S [Помещаем сформированную команду r2 в свою ячейку]

[58] [Начало внутреннего цикла]A 2 S [Добавляем в аккумулятор внутренний счетчик]

[59] S 149 S [Вычитаем из аккумулятора длину массива]

[60] E 91 S [Если результат >=0, то переходим к концу внутреннего цикла]

[61] T 0 S [Очищаем аккумулятор]

[62] [r1:] A 0 S [Добавляем в акк. el2 чтобы сравнить его с el1]

[63] [r2:] S 0 S [Вычитаем из акк. el1]

[64] E 82 S [Если (el2-el1) >= 0, то они стоят в правильном порядке, пропускаем смену текущего наименьшего элемента]

[65] T 0 S [Если нет, то начинаем смену текущего наименьшего элемента, обнуляем аккумулятор]

[66] A 6 S [Добавляем в аккумулятор флаг смены наименьшего элемента]

[67] S 148 S [Вычитаем из аккумулятора единицу, делая его отрицательным, была смена элементов]

[68] T 6 S [Кладем флаг обратно в свою ячейку]

[69] X 0 S

[70] X 0 S

[71] X 0 S

[72] A 62 S [Добавляем в аккумулятор команду r1 чтобы достать из нее адрес el2]

[73] S 145 S [Вычитаем из кода r1 опкод команды A, оставляя в аккумуляторе только адрес el2]

[74] T 5 S [Кладем адрес el2 в специальную ячейку]

[75] A 63 S [Аналогично добавляем в аккумулятор команду r2 чтобы достать из нее адрес el1]

[76] S 144 S [Убираем из кода r2 опкод команды S]

[77] T 4 S [Сохраняем адрес el1 в соответствующей ячейке]

[78] A 63 S [Помещаем в аккумулятор команду r2]

[79] S 4 S [Вычитаем из r2 находящийся в ней адрес el1]

[80] A 5 S [Добавляем в r2 адрес el2]

[81] T 63 S [Помещаем измененную r2 на ее место]

[82] T 0 S [Обнуляем аккумулятор]

[83] A 62 S [Переходим сюда чтобы увеличить на единицу адрес r1]

[84] A 147 S [Увеличиваем адрес в r1 на 1]

[85] T 62 S [Кладем измененную r1 обратно]

[86] A 2 S [Добавляем в аккумулятор внутренний счетчик чтобы увеличить его]

[87] A 148 S [Увеличиваем на единицу внутренний счетчик]

[88] T 2 S [Кладем измененный счетчик обратно]

[89] E 58 S [Повторение внутреннего цикла]

[90] X 0 S

[91] T 0 S [Внутренний цикл закончен, обнуление аккумулятора]

[92] A 6 S [Добавляем в аккумулятор флаг чтобы проверить необходимость смены местами элементов]

[93] E 112 S [Если аккумулятор >=0, то el1 уже был отсортирован]

[94] X 0 S

[95] X 0 S

[96] T 0 S [Если смена элементов местами необходима, то попадаем сюда, обнуляем аккумулятор]

[97] [l1:] A 158 S [Добавляем в аккумулятор элемент, на место которого придет найденный наименьший]

[98] T 3 S [Временно сохраняем элемент]

[99] A 5 S [Добавляем в аккумулятор адрес найденного наименьшего el1]

[100] A 145 S [Добавляем в аккумулятор опкод команды А]

[101] U 108 S [Кладем сформированную команду в l2]

[102] S 145 S [Убираем из аккумулятора уже лишний опкод А]

[103] A 146 S [Добавлем в аккумулятор опкод команды Т]

[104] T 111 S [Кладем сформированную команду в k2]

[105] X 0 S

[106] X 0 S

[107] X 0 S

[108] [l2:] A 0 S [Добавляем в аккумулятор найденный наименьший элемент]

[109] [k1:] T 158 S [Кладем наименьший элемент на место элемента, сохраненного в 3 ячейке]

[110] A 3 S [Добавляем в акк. элемент, сохраненный в swap]

[111] [k2:] T 0 S [Кладем сохраненный элемент на место наименьшего]

[112] T 0 S [Попадаем сюда, если пропускали замену элементов, обнуляем аккумулятор]

[113] T 6 S [Очищаем 6 ячейку, помещая в нее ячейку 0]

[114] X 0 S

[115] X 0 S

[116] A 97 S [Добавляем в аккумулятор команду l1 чтобы увеличить в ней адрес]

[117] A 147 S [Увеличиваем на единицу адрес в команде l1]

[118] T 97 S [Кладем l1 обратно в свою ячейку]

[119] X 0 S

[120] X 0 S

[121] A 109 S [Добавляем в аккумулятор команду k1 чтобы увеличить в ней адрес]

[122] A 147 S [Увеличиваем на единицу адрес в команде k1]

[123] T 109 S [Кладем измененную k1 обратно]

[124] X 0 S

[125] X 0 S

[126] A 1 S [Помещаем в аккумулятор внешний счетчик]

[127] A 148 S [Уыеличиваем на единицу внешний счетчик]

[128] T 1 S [Кладем его обратно]

[129] E 41 S [Повторение внещнего цикла]

[130] X 0 S [Конец внешнего цикла]

[131] Z 0 S [Переходим сюда если итерации внешних циклов закончены, массив отсортирован]

[132] T 0 S

[133] X 0 S

[134] A 8 S [Загружаем в аккумулятор изначальный вид команды l1]

[135] T 97 S [Возвращаем изначальный вид l1]

[136] A 7 S [Загружаем в аккумулятор изначальный вид команды k1]

[137] T 109 S [Возвращаем изначальный вид k1]

[138] X 0 S

[139] Z 0 S [Конец программы]

[140] Z 0 S

[141] Z 0 S

[142] Z 0 S

[143] X 0 S

[144] S 0 S [Опкод команды S]

[145] A 0 S [Опкод команды А]

[146] T 0 S [Опкод команды T]

[147] P 1 S [const 2 = 2]

[148] P 0 L [const 1 = 1]

[149] P 4 S [Длина массива]

[150] P 158 S [316] [Умноженный на 2 адрес первого элемента массива]

[151] P 0 S

[152] P 0 S

[153] P 0 S

[154] P 0 S

[155] P 0 S

[156] P 0 S

[157] P 0 S

[158] P 7 L [15]

[159] P 1 L [3]

[160] P 0 L [1]

[161] P 15 L [31]

[162] P 3 L [7]

[163] P 63 L [127]

[164] P 127 L [253]

[165] P 31 L [63]

[166] [Конец]

# Работа программы Orders1











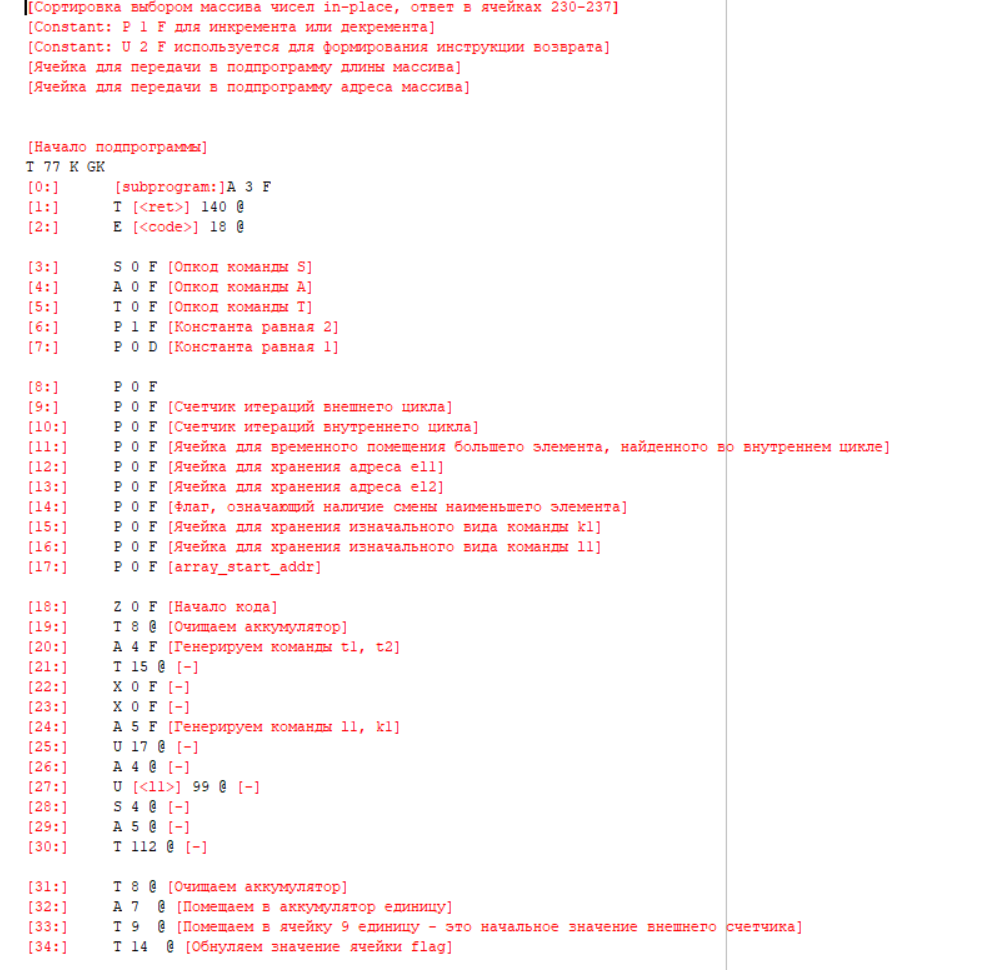


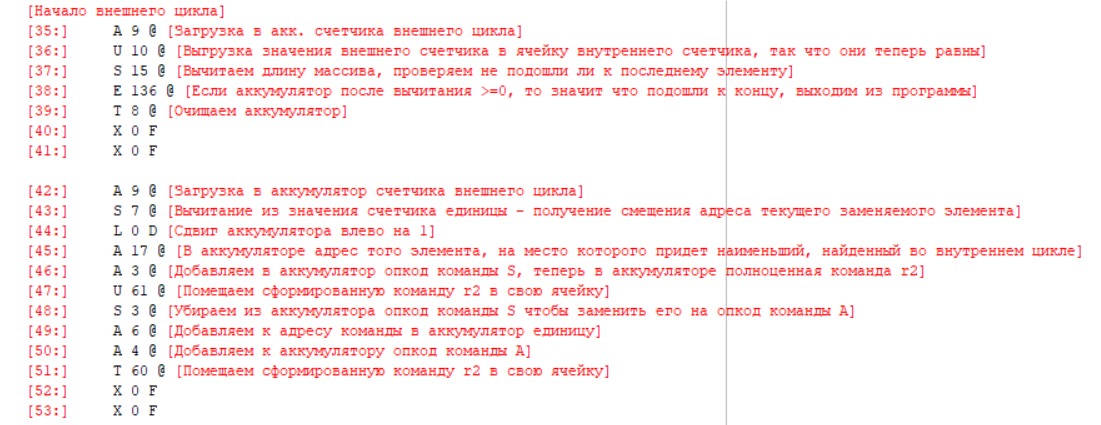


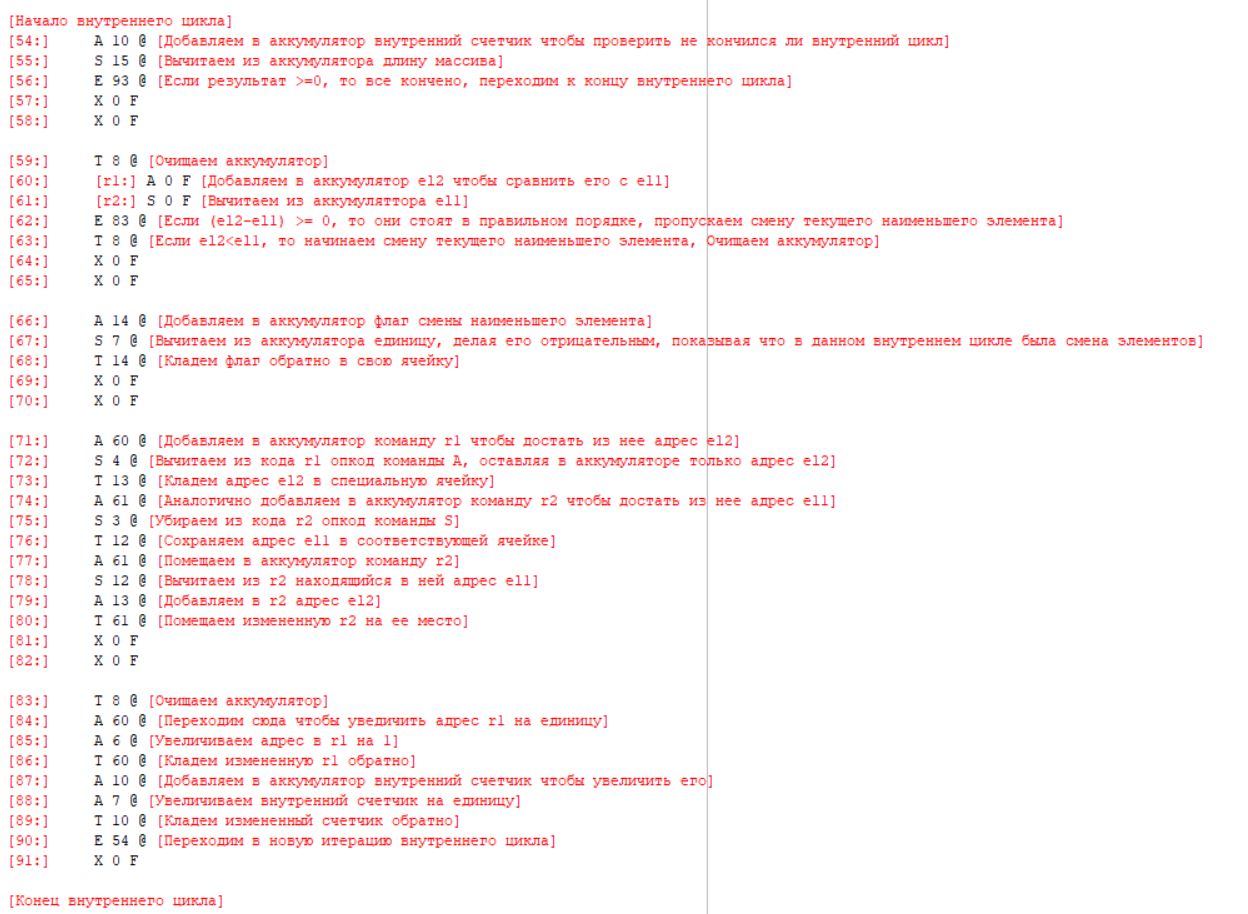


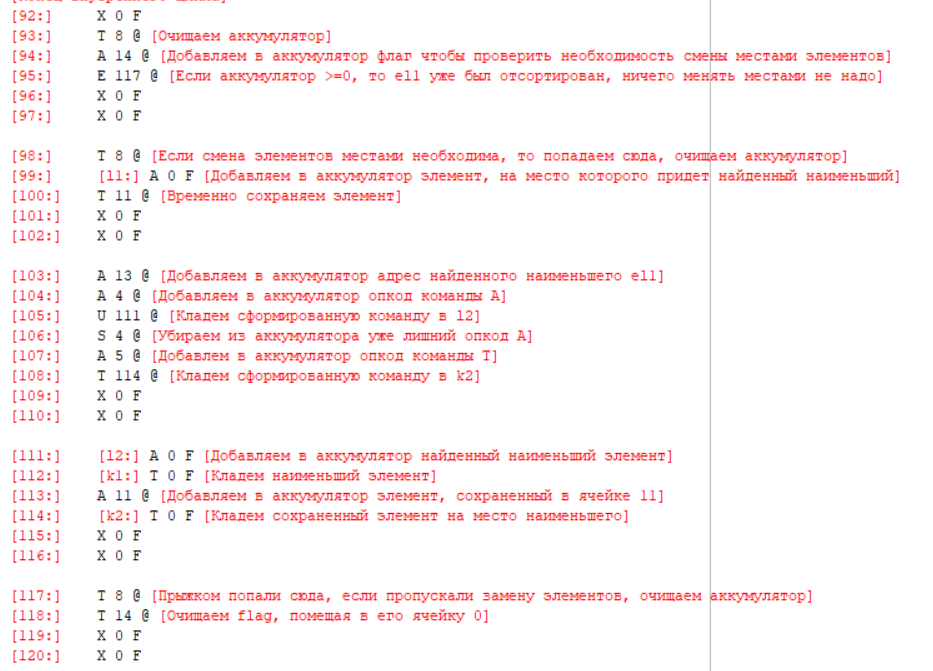
# Программа Orders2

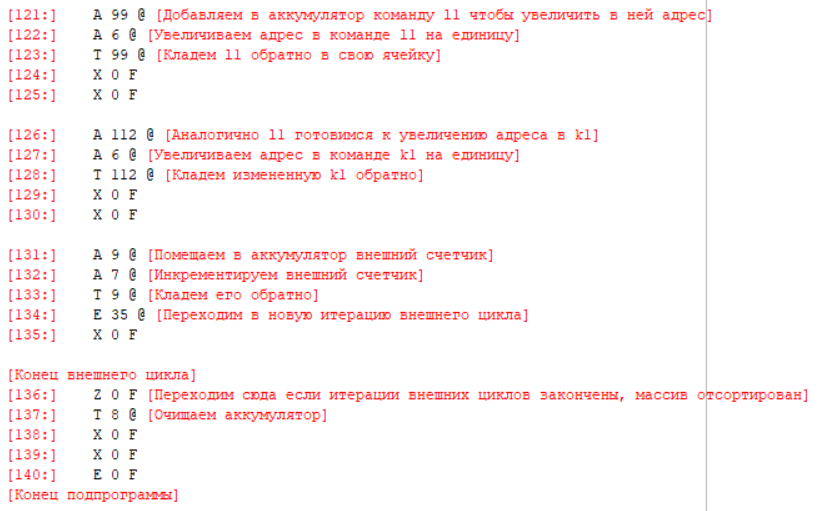
Результат работы программы – отсортированный массив на месте старого (ячейки 230-238)

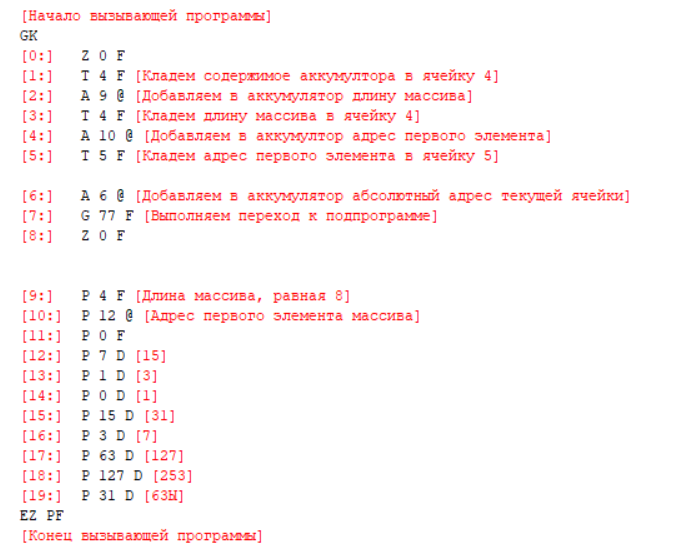












# Работа программы Orders2















