

Министерство образования и науки Украины
Национальный аэрокосмический университет
им. Н.Е. Жуковского

Кафедра 503

Лабораторная работа № 7

по курсу

«Архитектура ЭВМ»

Тема: «Изучение одномерных массивов в МП x86»

Вариант № 37

Выполнил: ст. гр. 525д

Петренко Р.Е.

Проверил: ст. преподаватель

Дужий В.И.

Харьков 2005

1. **Задание.** Выполнить сортировку массива простым выбором, используя команды управления циклами и режим адресации с масштабированием. Длина массива и сам массив вводятся из файла `in.txt`. Признак конца ввода – длина массива, равная нулю. Файл с исходными данными содержит корректные данные, которые проверять не нужно. Результаты работы программы выводятся на терминал.

Написать программу на С и на ассемблере, которая читает из файла массив, выполняет его сортировку и выводит результаты на дисплей.

2. **Исходные данные.**

`in.txt` – текстовый файл с исходными данными;

`MAX_LENGTH` – максимальная длина исходного массива – константа;

`arr, arr_a` – исходный массив длинных целых чисел.

3. **Требуемый результат.**

`arr, arr_a` – результирующий массив длинных целых чисел.

4. **Описание алгоритма на псевдокоде.**

Открыть файл с именем `in.txt`;

Повторять в цикле

Прочитать длину массива;

Если длина массива недопустима, завершить цикл;

Копировать содержимое файла в исходный массив;

Сделать копию массива для ассемблера;

Вывод исходного массива;

Сортировать массив на С;

Сортировать массив на ассемблере;

Вывод исходного массива после сортировки на С;

Вывод исходного массива после сортировки на ассемблере;

Закрыть файл с именем `in.txt`;

Детальный алгоритм решения задачи представлен ниже (рис.1).

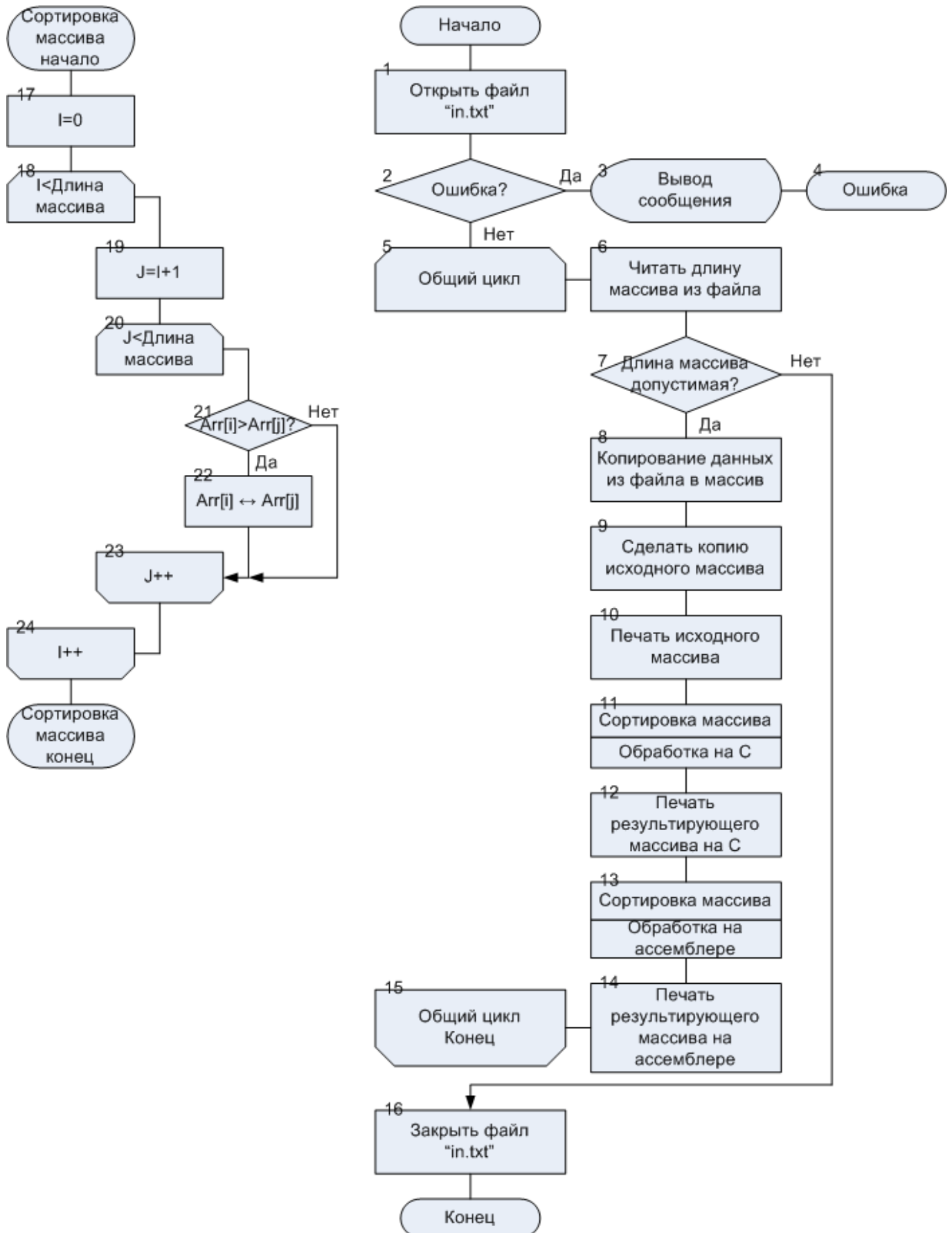


Рис.1. Схема алгоритма сортировки простым выбором и решения задачи в целом.

5. Распределение переменных по регистрам в программе lab5.c

edx		tmp	eax
ecx	len		ebx
esi	i	j	edi

6. Текст программы с комментариями.

```
//+=====
// File lab_5.c
// Циклическая программа, работающая с массивом
// Вариант 37
// Программа выполняет сортировку простым выбором
// Длина масива и сам массив вводятся из файла in.txt
//
// (C) Дужий В.И., 2005
//-----
#include <stdio.h>
#define MAX_LENGTH 100
long int arr[MAX_LENGTH], arr_a[MAX_LENGTH];
int len;

int main()
{
    int i, j, tmp;
    FILE *fin;

    printf("\n\t\t(C) Дужий В.И., 2005\n");
    // Открыть файл с исходными данными
    if((fin = fopen("in.txt","r"))==NULL)
    {
        printf("\nCan't open file\n"); exit(1);
    }
    for (;;)
    {
        // Читать длину масива
        fscanf(fin,"%i\n",&len);
        // Если массив имеет ненулевую и допустимую длину, читать его
        if (len==0 || MAX_LENGTH>100) break;
        // Копировать файл в массив
        for (i=0;i<len;i++) fscanf(fin,"%i",&arr[i]);
        // Сделать копию для ассемблера
        for (i=0;i<len;i++) arr_a[i]=arr[i];
        printf("\n\tСортировка массива простым выбором");
        //===== С =====
        // Вывод исходного массива
        printf("\nИсходный масив имеет длину: %i\n",len);
        for (i=0; i<len; i++) printf("%8i",arr[i]);
        // Сортировка массива
        for (i=0; i<len; i++)
            for (j=i+1; j<len; j++)
            {
                if(arr[i]>arr[j])
                {
                    tmp = arr[i];
                    arr[i] = arr[j];
                    arr[j] = tmp;
                }
            }
    }
}
```

```

    }
// Вывод результирующего массива
    printf("\nРезультирующий массив ( C )\n");
    for (i=0; i<len; i++) printf("%8i",arr[i]);
//===== Assembler =====
// Разделить исходные переменные на знаменатель DENOM
asm(
//     for (i=0; i<len; i++)
"        xorl    %esi,%esi        ;"
"        movl    len,%ecx         ;"
//     for (j=i+1; j<len; j++) {
"L0:
"        movl    %esi,%edi        ;"
"        incl    %edi             ;"
"L1:
//         tmp = arr[i];
"        movl    arr_a(,%esi,4),%eax ;"
//         if(arr[i]>arr[j]) {
"        cmpl    arr_a(,%edi,4),%eax ;"
"        jl L2                    ;"
//         arr[i] = arr[j];
"        xchgl    %eax,arr_a(,%edi,4) ;"
//         arr[j] = tmp;}
"        xchgl    %eax,arr_a(,%esi,4) ;"
"L2:
"        incl    %edi             ;"
"        cmpl    %ecx,%edi        ;"
"        jl      L1              ;"
"        incl    %esi            ;"
"        cmpl    %ecx,%esi        ;"
"        jl      L0              ;");
// Вывод результатов
    printf("\nРезультирующий массив (Asm)\n");
    for (i=0; i<len; i++) printf("%8i",arr[i]);
    printf("\n");
}
fclose(fin); // Закрыть исходный файл
return 0;
}

```

7. Тестовые примеры.

Но- мер	Дли на	Исходный массив	Ожидаемый результат	Полу- ченный результат	Цель теста
1	0				Признак конца ввода
2	101				Длина больше максимальной
3	5	-20, -10, 0, 30, 50	-20, -10, 0, 30, 50		Элементы по возрастанию
4	7	90,80,70,50,-30,-50,-70	-70,-50,-30,50,70,80,90		Элементы по убыванию
5	6	100,-4, 8,-30,-99,200	-99, -30, -4, 8, 100,200		Элементы в случайном порядке
6	1	1234	1234		Массив из одного элемента
7	100	0,-1,-2,-3,-4,-5,-6, -7,-8,-9,-10,-11...	-99,-98,-97,-96, -95,-94,-93,...		Массив из 100 элементов