Отчёт по лабораторной работе № 12 по курсу «Архитектура компьютера и информационных систем»

Студент группы М8О-112Б-22

Епифанов Евгений Валерьевич, № по списку: 9

# E-mail: [epi\_abitur](mailto:epi_abitur)@mail.ru

# Телефон: 8(916)783-95-69

# Работа выполнена: 9 ноября 2022 г.

Преподаватель: Никулин Сергей Петрович, каф.806

**1. Тема:** Техника работы с целыми числами. Системы счисления.

**2. Цель работы:** Составить программу на языке Си в целом типе данных, которая для любых допустимых и корректно записанных чисел этого типа в десятичном отображении, поступающих на стандартный ввод программы, выполняет указанное вариантом действие над их значениями.

**3. Задание:** составить программу на Си для решения полученной задачи (Вариант 30: “Упорядочить цифры числа попарно по возрастанию.”).

Изображение выглядит как текст

Автоматически созданное описание

**4. Оборудование (ПЭВМ студента):**

Ноутбук с процессором Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz, 1992 МГц, ядер: 4, логических процессоров: 8, ОП 8 ГБ. SSD 256 ГБ.

**5. Программное обеспечение ЭВМ студента:**

* Операционная система семейства UNIX, наименование: Linux Ubuntu (22.04) 5.15.0-48-generic #54-Ubuntu SMP Fri Aug 26 13:26:29 UTC 2022 x86\_64 x86\_64 x86\_64 GNU/Linux.
* Интерпретатор команд: GNU bash, version 5.1.16(1)-release (x86\_64-pc-linux-gnu).
* Компилятор gcc Linux.
* Редактор текстов: GNU Emacs 28.2
* Утилиты и прочие программы.

**6. Идея, метод, алгоритм решения задачи:**

1. Взаимодействие с программой.
   1. emacs \_\_\_.c – запуск emacs с последующим вводом кода.
   2. cc \_\_.c – компиляции программы.
   3. ./a.out – тестирование программы.
2. Программа:

#include <stdio.h>

#include <math.h>

int main()

{

long n, m, new = 0, k, rzr1, rzr2, ogr = 2147483647, flag = 0;

while(scanf("%li", &n)!=EOF)

{

if (fabs(n) >= ogr)

{

printf("%s\n","Error: Perepolnenie!");

continue;

}

if (n < 0)

{

n = fabs(n);

flag = 1;

}

m = n;

new = 0;

k = 1;

while (m > 0)

{

rzr1 = m % 10;

m /= 10;

rzr2 = m % 10;

m /= 10;

if (rzr1 <= rzr2)

{

new += k \* rzr2;

k \*= 10;

new += k \* rzr1;

}

else

{

new += k \* rzr1;

k \*= 10;

new += k \* rzr2;

}

k \*= 10;

}

if (flag == 1)

{

new \*= (-1);

n \*= (-1);

}

printf("%s%li%s%li\n","Input: ", n, " Output: ", new);

}

}

Данная программа получает на вход некоторое десятичное число. В случае, если число проходит ограничение по размеру int, в нем попарно начинаются сортироваться цифры по возрастанию, числа обрабатываются справа налево. Если входное число состоит из нечетного кол-ва цифр, самая левая цифра останется на своей позиции. Незначащие нули числа-результата стираются. Пары чисел записываются в переменные rzr1 и rzr2 для сравнения. Число k отображает позицию цифры в новом числе, является степенью 10. Переменная flag нужна для обработки отрицательных чисел.

После обработки числа программа выведет ответ: “Input: \_\_\_\_ Output: \_\_\_\_”. Если же число не прошло проверку на размер, будет выведена ошибка. Программа прекратит работу, когда будет нажата комбинация клавиш: Ctrl + D.

1. Тесты для программы:

Ввод: 1

Вывод: 1

Ввод: 123

Вывод: 123

Ввод: 1457

Вывод: 1457

Ввод: 7954

Вывод: 7945

Ввод: 987533

Вывод: 895733

Ввод: 954

Вывод: 945

Ввод: 120067

Вывод: 120067

Ввод: 540870

Вывод: 450807

Ввод: 10000000007

Вывод: Error: Perepolnenie!

Ввод: 1000007

Вывод: 1000007

Ввод: 100007

Вывод: 10007

Ввод: 90

Вывод: 9

Ввод: -1011

Вывод: -111

**7. Сценарий выполнения работы:**

1. Проанализировать полученную задачу и разработать для её решения корректно функционирующую программу.
2. Написать программу в текстовом редакторе Emacs.
3. Скомпилировать программу и проверить её работоспособность на заранее заготовленных тестах разной сложности.

**8. Распечатка протокола:**

evgeniy2@evgeniy2:~$ emacs laba12.c

Изображение выглядит как текст

Автоматически созданное описание

evgeniy2@evgeniy2:~$ emacs laba12.c

evgeniy2@evgeniy2:~$ cc laba12.c

evgeniy2@evgeniy2:~$ ./a.out

1

Input: 1 Output: 1

123

Input: 123 Output: 123

1457

Input: 1457 Output: 1457

7954

Input: 7954 Output: 7945

987533

Input: 987533 Output: 895733

954

Input: 954 Output: 945

120067

Input: 120067 Output: 120067

540870

Input: 540870 Output: 450807

10000000007

Error: Perepolnenie!

1000007

Input: 1000007 Output: 1000007

100007

Input: 100007 Output: 10007

90

Input: 90 Output: 9

-1011

Input: -1011 Output: -111

evgeniy2@evgeniy2:~$

9. Дневник отладки:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Лаб. или дом. | Дата | Время | Событие | Действие по исправлению | Примечание |
|  |  |  |  |  |  |  |

10. Замечания по существу работы: -

11. Выводы: Я научился решать задачи, связанные с целочисленным типом данных, на языке программирования Си. Я усовершенствовал свои навыки программирования на Си в Linux.

Недочёты при выполнении задания могут быть устранены следующим образом: -