**Отчет по лабораторной работе №12** по курсу \_\_Фундаментальная информатика\_\_\_\_\_\_\_\_\_\_\_\_

Студент группы М80-101Б-21 Постнов Александр Вячеславович, № по списку 17

Контакты www, e-mail:

Работа выполнена: «» 202 1г.

Преподаватель: каф. 806 \_\_\_\_\_Титов В.К.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Входной контроль знаний с оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Отчет сдан « » \_\_\_\_\_\_\_\_\_2021\_ г., итоговая оценка \_\_\_\_\_

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Тема:** Программирование на языке СИ\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Цель работы:** составить и отладить программу на СИ. \_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Задание** (*вариант: 4*)**:**

Вводится натуральное число, образовать два новых числа: первое состоит из цифр исходного числа, каждая их которых меньше, чем следующая, идущая за ней в исходном числе цифра, а второе из остальных цифр исходного числа. Распечатать два полученных числа и их сумму. Например: 11233455 -> 1234, 1355, 2589.

1. **Оборудование(лабораторное)**:

ЭВМ -, процессор -, имя узла сети - с ОП - ГБ,

НМД - ГБ, терминал- адрес -, принтер -

Другие устройства -

*Оборудование ПЭВМ студента, если использовалось:*

Процессор AMD Ryzen 5 4500U, с ОП 8 ГБ

Другие устройства -

1. **Программное обеспечение:**

Операционная система семейства -, наименование - версия - интерпретатор команд - версия

Система программирования - версия -

Редактор текстов - версия -

Утилиты операционной системы -

Прикладные системы и программы -

Местонахождение и имена файлов программ и данных -

*Программное обеспечение ЭВМ студента, если использовалось:*

Операционная система семейства GNU/Linux, наименование Pop!\_OS версия 21.04

интерпретатор команд GNOME Terminal версия 3.38.2

Система программирования \_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_версия\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Редактор текстов emacs версия 3.27.20

Утилиты операционной системы cat

Прикладные системы и программы -

Местонахождение и имена файлов программ и данных -

1. **Идея, метод, алгоритм**  решения задачи (в формах: словесной, псевдокода, графической [блок-схема, диаграмма, рисунок, таблица] или формальные спецификации с пред- и постусловиями)

Словесный алгоритм:

1. Так как последнюю цифру нельзя сравнить с последующей цифрой, она попадает во второе число, поэтому первое сравнение будет производиться с -1;
2. Проход по числу будет производиться с конца: последняя цифра числа будет получаться с помощью остатка от деления на 10, а отсекаться она будет с помощью целочисленного деления на 10
3. Сравнение текущей цифры будет происходить с последующей(в моем случае с предыдущей, так как я рассматриваю число с конца), если она будет меньше, то она попадает в первое число, а если нет, то во второе
4. Сборка числа будет производиться с помощью степеней 10, изначально переменным будет присвоено десять в 0 степени(1), когда цифра попадает в условие, текущая цифра умножается на степень десятки и прибавляется к ответу.

**7. Сценарий выполнения работы** [план работы, первоначальный текст программы в черновике (можно на отдельном листе) и тесты либо соображения по тестированию].

#include <stdio.h>

int main(void){

int num;

scanf("%d", &num);

int last\_num = -1; //так как последнюю цифру числа не с чем сравнивать => берем его в любом случае во 2 число,

//поэтому сравнение идет с -1

int num1 = 0;

int num2 = 0;

int t1 = 1; //степени десятки для первого и второго числа

int t2 = 1;

while (num > 0){

int e = num % 10; //текущая цифра числа

num /= 10; //отсечение последней цифры числа

if (e < last\_num){

num1 += t1 \* e;

t1 \*= 10;

}

else {

num2 += t2 \* e;

t2 \*= 10;

}

last\_num = e;

}

printf("%d, %d, %d \n", num1, num2, num1 + num2);

}

Тесты:

11233455 -> 1234, 1355, 2589;

123 -> 12, 3, 15;

132678 -> 1267, 38, 1305;

20 -> 0, 20, 20;

5 -> 0, 5, 5;

12345 -> 1234, 5, 1239;

*Допущен к выполнению работы.*  **Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**8. Распечатка протокола**  (подклеить листинг окончательного варианта программы с тестовыми примерами, подписанный преподавателем).

alex@alex-pc:~/mai\_labs/12c(?)$ cat head.txt

-----------------------------------------------------

| Лабораторная работа №12 |

| Программирование на ЯП СИ |

| Выполнил: студент группы М8О-101Б-21 |

| Постнов Александр Вячеславович |

-----------------------------------------------------

alex@alex-pc:~/mai\_labs/12c(?)$ g++ main.c

main.c: In function ‘int main()’:

main.c:17:21: error: expected ‘;’ before ‘}’ token

17 | t1 \*= 10

| ^

| ;

18 | }

| ~

main.c:21:21: error: expected ‘;’ before ‘}’ token

21 | t2 \*= 10

| ^

| ;

22 | }

| ~

alex@alex-pc:~/mai\_labs/12c(?)$ g++ main.c

alex@alex-pc:~/mai\_labs/12c(?)$ ./a.out

11233455

1234, 1355, 2589

alex@alex-pc:~/mai\_labs/12c(?)$ ./a.out

123

12, 3, 15

alex@alex-pc:~/mai\_labs/12c(?)$ ./a.out

132678

1267, 38, 1305

alex@alex-pc:~/mai\_labs/12c(?)$ ./a.out

20

0, 20, 20

alex@alex-pc:~/mai\_labs/12c(?)$ ./a.out

5

0, 5, 5

alex@alex-pc:~/mai\_labs/12c(?)$ ./a.out

12345

1234, 5, 1239

alex@alex-pc:~/mai\_labs/12c(?)$ cat main.c

#include <stdio.h>

int main(void){

int num;

scanf("%d", &num);

int last\_num = -1; //так как последнюю цифру числа не с чем сравнивать => берем его в любом случае во 2 число,

//поэтому сравнение идет с -1

int num1 = 0;

int num2 = 0;

int t1 = 1; //степени десятки для первого и второго числа

int t2 = 1;

while (num > 0){

int e = num % 10; //текущая цифра числа

num /= 10; //отсечение последней цифры числа

if (e < last\_num){

num1 += t1 \* e;

t1 \*= 10;

}

else {

num2 += t2 \* e;

t2 \*= 10;

}

last\_num = e;

}

printf("%d, %d, %d \n", num1, num2, num1 + num2);

**9. Дневник отладки** должен содержать дату и время сеансов отладки и основные события (ошибки в сценарии и программе, нестандартные ситуации) и краткие комментарии к ним. В дневнике отладки приводятся сведения об использовании других ЭВМ, существенном участии преподавателя и других лиц в написании и отладке программы.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Лаб. или дом. | Дата | Время | Событие | Действие по исправлению | Примечание |
| 1 | дом | 05.11 | 13:51 | Программа не скомпилировалась из-за синтаксической ошибки | Добавил “;” в строках 17, 21. |  |

**10. Замечания автора :**

11.**Выводы**

\_\_В ходе лабораторной работе я составил и отладил программу на СИ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Недочёты при выполнении задания могут быть устранены следующим образом: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись студента \_Постнов\_\_\_\_\_\_\_\_\_\_\_\_\_\_