

Отчет по лабораторной работе № 12 по курсу “Фундаментальная информатика”

Студент группы М80-109Б-22 Мозговой Никита Евгеньевич, № 10

Контакты email: MozgovoyNE@mail.ru
Telegram: @M1N8E

Работа выполнена: «14» декабря 2022г.

Преподаватель: каф. 806 Сысоев Максим Алексеевич

Отчет сдан «. »__12____2022 г., итоговая оценка_____

Подпись преподавателя_____

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 выполняет указанное действие.

3. **Задание:** Вариант 30. Упорядочить цифры числа попарно по убыванию.

4. **Оборудование (студента):**

Процессор 2,5 GHz 2-ядерный процессор Intel Core i5 ОП 4 Гб, Монитор 1280x800

5. **Программное обеспечение (студента):**

Операционная система семейства: *linux*, наименование: *ubuntu*, версия 22.04 *jammy*

интерпретатор команд: *bash* версия 5.1.16(1)-release

Система программирования CLion версия 2021 1.3, редактор текстов *nano* версия 6.2

Утилиты операционной системы WinRar, Microsoft Word

Прикладные системы и программы Ubuntu wsl, CLion, Google Chrome

Местонахождение и имена файлов программ и данных на домашнем компьютере ~/home

6. Идея, метод, алгоритм

Число можно реконструировать, если знать, на какой позиции стояла определённая цифра, тогда получится следующее:

$A * 10^0 + B * 10^1 + C * 10^2 + \dots +$
где A, B, C – цифры числа

7. Сценарий выполнения работы

Входные данные	Выходные данные	Описание тестируемого случая
1	1	Число длиной 1
1 2 3	1 2 3	Несколько чисел длиной 1
123456	214365	Обычный случай
1234567	2143657	Нечётное количество цифр

8. Распечатка протокола

```
#include <stdio.h>
#include <assert.h>
#include <ctype.h>

int powFunc(int num, int power) {
    if (power == 0) {
        return 1;
    }
    int res = 1;
    while (power > 0) {
        res *= num;
        --power;
    }
    return res;
}

int getNewPair(int digit1, int digit2, int pairsDone, int isReversed) {
    if (isReversed == 1 ? digit1 >= digit2 : digit1 <= digit2) {
        return digit1 * powFunc(10, pairsDone * 2) + digit2 * powFunc(10, pairsDone * 2 + 1);
    } else {
        return digit2 * powFunc(10, pairsDone * 2) + digit1 * powFunc(10, pairsDone * 2 + 1);
    }
}

int reverseNum(int num, int pairsDone, int isOdd) {
    int counter = isOdd == 1 ? 0 : -1;
    int res = 0;

    while (num > 0) {
        res += num % 10 * powFunc(10, pairsDone * 2 - counter++);
        num /= 10;
    }

    return res;
}

void test_powFunc() {
    assert(powFunc(10, 0) == 1);
    assert(powFunc(1, 100) == 1);
    assert(powFunc(10, 2) == 100);
    assert(powFunc(2, 9) == 512);
    assert(powFunc(0, 9) == 0);
}

void test_getNewPair() {
    assert(getNewPair('4' - '0', '1' - '0', 0, 0) == 41);
    assert(getNewPair('1' - '0', '4' - '0', 0, 0) == 41);
    assert(getNewPair('9' - '0', '9' - '0', 0, 0) == 99);
}
```

```

}

void test_reverseNum() {
    assert(reverseNum(123456, 2, 0) == 654321);
    assert(reverseNum(654321, 2, 0) == 123456);
    assert(reverseNum(5454545, 3, 1) == 5454545);
}

void execTests() {
    test_powFunc();
    test_getNewPair();
    test_reverseNum();
}

int main() {
    execTests();

    // попарно по возрастанию
    // 1234 =
    // = 1 * 10**3 + 2 * 10**2 + 3 * 10**1 + 4 * 10**0
    int c;
    int digit1 = -1;
    int digit2 = -1;
    int isOdd = 0;
    int resNum = 0;
    int pairsDone = 0;

    while ((c = getchar()) != 'a') {
        if (isdigit(c) == 0) {
            if (resNum == 0) {
                if (digit1 == -1 && digit2 == -1) {
                    continue;
                }

                if (digit1 != -1) {
                    printf("%d ", digit1);
                } else {
                    printf("%d ", digit2);
                }

                digit1 = -1;
                digit2 = -1;
                continue;
            }

            if (digit1 != -1 || digit2 != -1) {
                if (digit1 != -1 && digit2 == -1) {
                    resNum += digit1 * powFunc(10, pairsDone * 2);
                    isOdd = 1;
                } else resNum += getNewPair(digit1, digit2, pairsDone, 0);
            }
        }
    }
}

```

```
printf("%d ", reverseNum(resNum, pairsDone, isOdd));
resNum = 0;

digit1 = -1;
digit2 = -1;
isOdd = 0;
pairsDone = 0;

continue;
}

if (digit1 == -1 || digit2 == -1) {
    if (digit1 == -1) digit1 = c - '0';
    else digit2 = c - '0';
} else {
    resNum += getNewPair(digit1, digit2, pairsDone, 0);

    digit1 = c - '0';
    digit2 = -1;

    pairsDone++;
}

}

if (digit1 != -1 && digit2 != -1) {
    resNum += getNewPair(digit1, digit2, pairsDone, 0);
    printf("%d", resNum);
} else if (digit1 != -1) {
    printf("%d", digit1);
}
else if (digit2 != -1) {
    printf("%d", digit2);
}
}
```

9. Дневник отладки

№	Лаб. или дом.	Дата	Время	Событие	Действие по исправлению	Примечание
1						

10. Замечания автора
Отсутствуют

11. Выводы
Интересная лаба, по сложности не сильно легче 11. Было интересно работать с числами на СИ (по сути, принцип такой же как и в других языках программирования)

Подпись студента _____

