Южно-Уральский государственный колледж, корпус Информационных Технологий и Экономики

**Отчет**

По дисциплине МДК.01.04. Системное программирование

Лабораторная работа №14 «Разработка и использование

рекурсивных процедур и функций»

Вариант №15

Выполнил студент Яковлев Е.К.

Группа ИСп320Д

2023 г.

**Ход работы:**

1. **Задание:** Рекурсивно описать функцию maxdig(N), которая находит наибольшую цифру в десятичной записи неотрицательного целого числа N. Например, maxdig(27306) = 7.
   1. Код программы:

#include <stdio.h>

#include <locale>

int maxdig(int n) {

if (n < 10) {

return n;

}

else {

int max = maxdig(n / 10);

int lastDigit = n % 10;

return (lastDigit > max) ? lastDigit : max;

}

}

int main() {

setlocale(LC\_ALL, "");

int number;

int result;

while (true)

{

printf("Введите неотрицательное целое число: ");

scanf\_s("%d", &number);

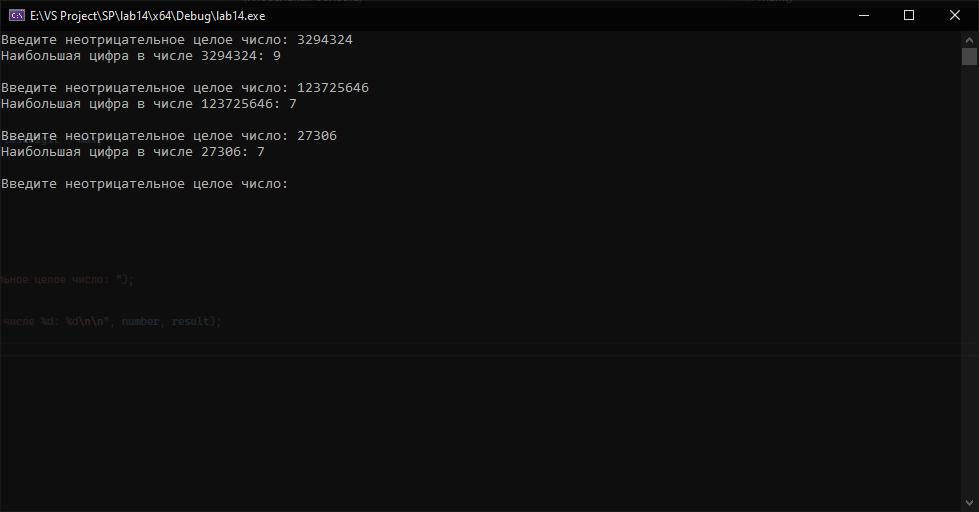
result = maxdig(number);

printf("Наибольшая цифра в числе %d: %d\n\n", number, result);

}

}

* 1. Снимок экрана:



**Контрольные вопросы**

1. **Понятие рекурсивных функций.**

Рекурсивная функция — это функция, которая вызывает сама себя.

1. **Особенности использования рекурсивных функций.**

Базовый случай: это условие, при котором функция прекращает вызывать саму себя и возвращает результат.

Рекурсивный случай: Рекурсивная функция должна также иметь рекурсивный случай, в котором она вызывает саму себя с другими аргументами. Это обеспечивает продолжение выполнения функции до достижения базового случая.

Стек вызовов: при каждом вызове рекурсивной функции создается новый фрейм стека вызовов, который содержит локальные переменные и возвращаемый результат. Каждый фрейм стека вызовов сохраняется в памяти до достижения базового случая, а затем извлекается в обратном порядке для получения результата.

Ограничение глубины рекурсии: Рекурсивные функции могут вызываться множество раз, что может привести к переполнению стека вызовов и вызвать ошибку "Stack Overflow". Поэтому важно обосновывать, что рекурсия будет завершена в конечном итоге или использовать итеративные алгоритмы вместо рекурсивных.

Эффективность: в некоторых случаях рекурсивные функции могут быть менее эффективными, чем итеративные алгоритмы. Это связано с дополнительными расходами на вызов функции и управление стеком вызовов.

**Вывод:** таким образом, я изучил особенности работы с рекурсивными функциями в языке Си и написал программу обработки данных с использованием рекурсивных данных в соответствии с заданным вариантом.