ГУАП

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

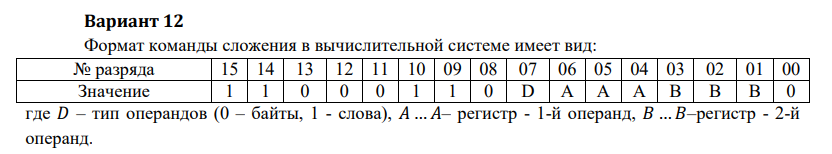
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ассистент |  |  |  | М. А. Мурашова |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ |
| Программирование поразрядных операций |
| по курсу: ОСНОВЫ ПРОГРАММИРОВАНИЯ |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4135К |  |  |  | Столяров Н.С. |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2022

**1.Цель работы:** Целью работы является изучение поразрядных операций типа НЕ, И, ИЛИ, исключающее ИЛИ и операций сдвига.  
  
**2.Задачи работы:** Составить две программы, первая из которых вводит составные части структуры данных, приведѐнной в индивидуальном варианте, как десятичные числа и формирует из них заданную упакованную структуру как 16-ричное число. Вторая программа вводит упакованную структуру как 16-ричное число и выводит значения отдельных еѐ составных частей как десятичные числа.   
  
  
**3.Описание функций:  
  
Файл <<code.cpp>>  
1)Имя: main**   
**Назначение:** Запрос от пользователя операндов в регистров для получения формата команды сложения.  
**Входные данные:** нет.   
**Выходные данные:** нет.  
**Побочный эффект:** отсутствует.   
**Тестовые данные:**

|  |  |  |  |
| --- | --- | --- | --- |
| d (Тип операндов) | A (1 операнд) | B (2 операнд) | Ответ |
| 1 | 7 | 7 | 0xfe |
| 0 | 8 | 12 | Числа не входят в диапазон |
| 3 | 4 | 3 | Числа не входят в диапазон |

Прототип: int main()

|  |  |
| --- | --- |
| **Псевдокод** | **Блок-схема** |
| Функция проверки ввода данных read\_int()  Функция проверки ввода данных read\_int()  Функция проверки ввода данных read\_int()  Вызов функции code() Вывод в консоль вернувшегося значения | C:\Users\nikit\Desktop\diagram.png |

**2)Имя: read\_int**  
**Назначение:** проверка переменной a на корректность ввода  
**Входные данные:** x(целочисленное значение).   
**Выходные данные:** x(целочисленное значение).  
**Побочный эффект:** отсутствует.   
**Прототип:** double read\_int()  
  
**Псевдокод:**  
Цикл продолжается до тех пор, пока пользователь не введет корректное значение

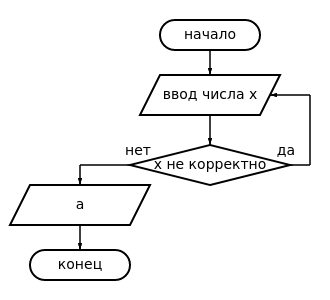
Ввод числа x

Если предыдущее извлечение оказалось неудачным, то...

Возвращаем scanf в 'обычный' режим работы(обнуление битов состояния)

Удаляем значения предыдущего ввода из входного буфера

Если всё хорошо, то возвращаем x

Возврат значения x  
**Блок-схема:**

**2)Имя: code**  
**Назначение:** перевод значений в формат.  
**Входные данные:** a ,b, d.   
**Выходные данные:** формат сложения  
**Побочный эффект:** отсутствует.   
**Прототип:** unsigned short code()

|  |  |
| --- | --- |
| **Псевдокод** | **Блок-схема** |
| Проверка корректности данных  Возврат команды | C:\Users\nikit\Desktop\diagram (1).png |

**ЛИСТИНГ КОДА:**/\*

Составить две программы, первая из которых вводит составные части структуры данных,

приведѐнной в индивидуальном варианте, как десятичные числа и формирует из них заданную

упакованную структуру как 16-ричное число. Вторая программа вводит упакованную структуру

как 16-ричное число и выводит значения отдельных еѐ составных частей как десятичные числа.

\*/

#include <iostream>

#include "lib.h"

#include "add.h"

using namespace std;

int main() {

// смена кодировки

system("chcp 65001");

int d = 0;

int a = 0;

int b = 0;

draw\_line(20);

cout << "Тип операндов (0-байты, 1-слова): ";

d = read\_int();

cout << "Регистр - 1 операнд: ";

a = read\_int();

cout << "Регистр - 2 операнд: ";

b = read\_int();

draw\_line(20);

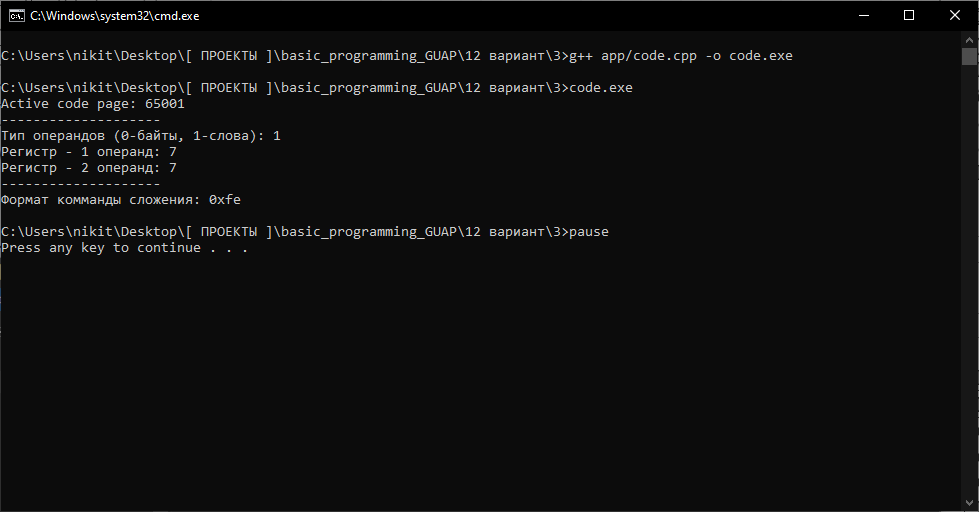
unsigned short z = code(a, b, d);

cout << "Формат комманды сложения: 0x" << hex << z << endl;

return 0;

}

**Пример выполнения программы:**



Видно, что результаты расчётов совпадают с тестовыми данными.

**Файл <<decode.cpp>>  
1)Имя: main**   
**Назначение:** Ввод формата сложения для получения введённых значений  
**Входные данные:** нет.   
**Выходные данные:** нет.  
**Побочный эффект:** отсутствует.   
**Тестовые данные:**

|  |  |
| --- | --- |
| X (16-ричное число от 0 до 0xFFFF) | Ответ |
| 0xfe | Тип операндов: 1  Регистр - 1 операнд: 7  Регистр - 2 операнд: 7 |

**Прототип:** int main()

|  |  |
| --- | --- |
| Псевдокод | Блок-схема |
| Ввод X (Формат сложения)  Вызов функции decode() Вывод в консоль вернувшегося значения | C:\Users\nikit\Desktop\diagram (2).png |

**2)Имя: decode**  
**Назначение:** Перевод формата в значения.  
**Входные данные:** X (формат сложения).   
**Выходные данные:** a, b, d.  
**Побочный эффект:** отсутствует.   
**Прототип:** void decode()

|  |  |
| --- | --- |
| **Псевдокод** | **Блок-схема** |
| Перевод формата в значение  Вывод всех значений | C:\Users\nikit\Desktop\diagram (3).png |

**ЛИСТИНГ КОДА:**/\*

Составить две программы, первая из которых вводит составные части структуры данных,

приведѐнной в индивидуальном варианте, как десятичные числа и формирует из них заданную

упакованную структуру как 16-ричное число. Вторая программа вводит упакованную структуру

как 16-ричное число и выводит значения отдельных еѐ составных частей как десятичные числа.

\*/

#include <iostream>

#include "lib.h"

#include "add.h"

using namespace std;

int main() {

// смена кодировки

system("chcp 65001");

unsigned int x;

draw\_line(20);

cout << "Введите команду сложения (16-ричное число от 0 до 0xFFFF): ";

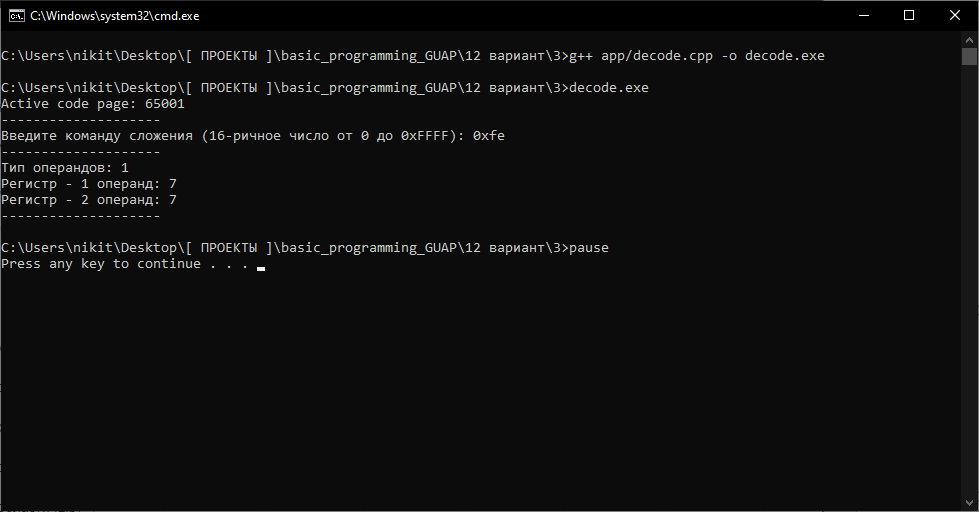
scanf("%x", &x);

decode(x);

return 0;

}

**Пример выполнения программы:**



Видно, что результаты расчётов совпадают с тестовыми данными.

**Анализ результатов и выводы:**

В ходе этой лабораторной работы мы научились работать с поразрядными операциями.

Из недостатков можно отметить то что при неправильном вводе операндов и регистров, программа просто выведет ошибку и не даст пользователю ещё раз ввести данные.

Из плюсов можно отметить то что программа корректно переводит данные.