

ГУАП
КАФЕДРА №43

ОТЧЁТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

К.Т.Н., доц. _____
должность, уч. степень, звание

подпись, дата

А.В. Туманова _____
инициалы, фамилия

ОТЧЁТ О ЛАБОРАТОРНОЙ РАБОТЕ №3

ПРОГРАММИРОВАНИЕ ПОРАЗРЯДНЫХ ОПЕРАЦИЙ

по дисциплине: ОСНОВЫ ПРОГРАММИРОВАНИЯ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. Z7431

 23.11.2018
подпись, дата

 М.Д.Семочкин
инициалы, фамилия

Санкт-Петербург
2018

1. Цель работы

Целью работы является изучение поразрядных операций типа НЕ, И, ИЛИ, исключающее ИЛИ и операций сдвига.

2. Задание

Согласно варианту №24,

Составить две программы, первая из которых вводит составные части структуры данных, приведённой 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 её составных частей как десятичные числа.

Для чтения и записи в потоки ввода/вывода следует использовать манипуляторы dec и hex.

Вариант 24

Заголовок пакета в системе передачи данных имеет формат:

№ разряда	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Значение	T	T	T	0	S	S	S	S	L	L	L	L	L	L	L	L

где *T ... T* – тип пакета, *S ... S* – идентификатор источника, *L ... L* – длина пакета.

3. Описание созданных функций

Программа 1:

Для реализации задания нам потребуются следующие функции:

Имя: getCorrectInput

Назначение: Запрашивать ввод данных с клавиатуры до тех пор, пока введенное значение не будет меньше переданного как аргумент числа

Входные данные:

- varName – название вводимой переменной (чтобы выводить правильную подсказку)
- limit – верхняя граница допустимых значений введенного числа

Выходные данные:

- userInput

Побочный эффект: отсутствует.

Прототип: short getCorrectInput(char varName, const short limit)

Алгоритм:

- псевдокод

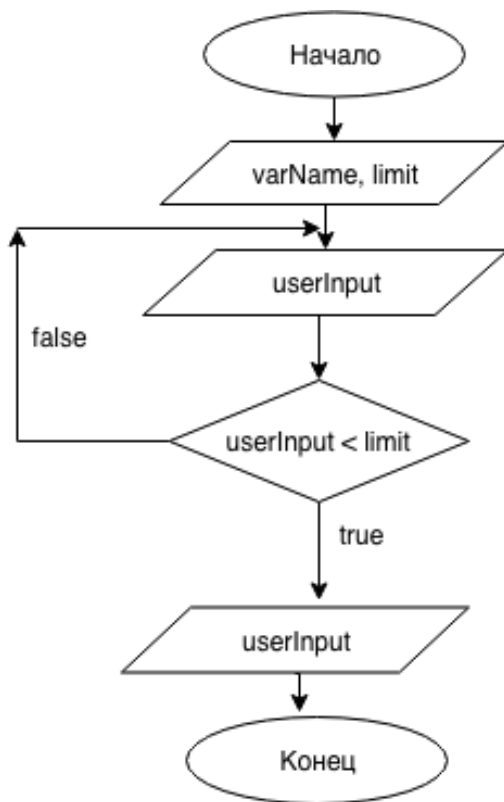
пока не будет введено корректное значение

вывести предложение ввести переменную

записать переменную из ввода

если значение введенной переменной меньше limit, вернуть его

- блок-схема



Имя: pack

Назначение: Сформировать структуру в виде 16-ричного числа из трех десятичных чисел

Входные данные:

- t – число T
- s – число S
- l – число L

Выходные данные:

- упакованная структура

Побочный эффект: отсутствует.

Тестовые данные:

t	s	l	результат
3	14	89	6e59

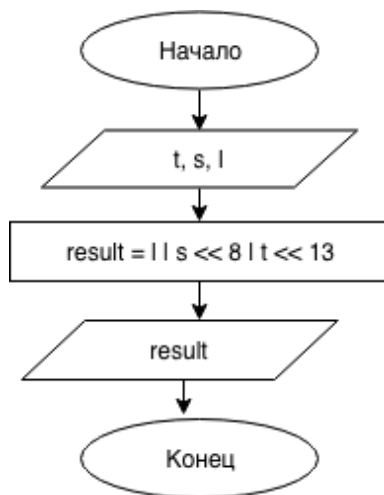
Прототип: pack(short t, short s, short l)

Алгоритм:

- псевдокод

вернуть (значение l) И (побитово сдвинутое на 8 влево значение s) И (побитово сдвинутое на 13 влево значение t)

- блок-схема



Программа 2

Имя: unpackAndPrint

Назначение: Разбить структуру на составляющие ее числа T, L, S, и вывести их на экран

Входные данные:

- packedHex – упакованная 16-ричная структура

Выходные данные:

нет

Побочный эффект: выводит на экран значения T, L, S, представленные в десятичном виде

Тестовые данные:

packedHex	t	l	s
6e59	3	14	89

Прототип: unpackAndPrint(short packedHex)

Алгоритм:

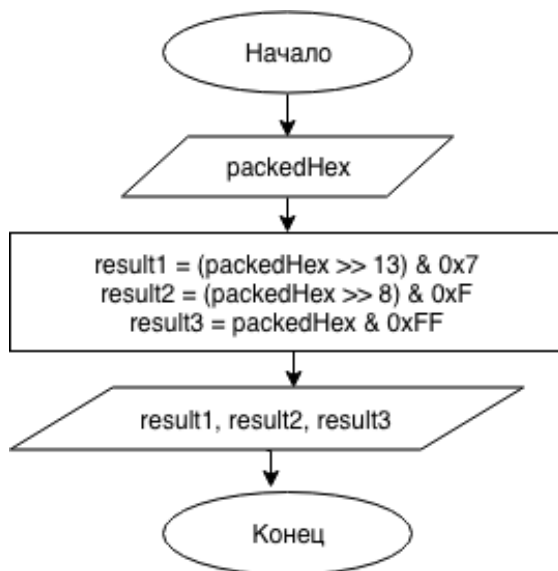
- псевдокод

вывести на экран ((побитово сдвинутое на 13 влево значение T) И (максимально возможное значение T в 16-ричном виде)) в десятичной форме

вывести на экран ((побитово сдвинутое на 8 влево значение S) И (максимально возможное значение S в 16-ричном виде)) в десятичной форме

вывести на экран ((значение L) И (максимально возможное значение L в 16-ричном виде)) в десятичной форме

- блок-схема



4. Листинг программы

Программа 1:

```
#include <iostream>
using namespace std;

short pack(short t, short s, short l) {
    return l | s << 8 | t << 13;
}

short getCorrectInput(char varName, const short limit) {
    int userInput;
    cout << "Введите " << varName << ": ";
    cin >> userInput;
    if (cin.fail()) {
        cin.clear();
        cin.ignore(256, '\n');
        cout << "Вы должны ввести целое число!" << endl;
        return getCorrectInput(varName, limit);
    }
    if (userInput <= limit) {
        return userInput;
    } else {
        cout << "Введено слишком большое число, попробуйте снова" << endl;
        return getCorrectInput(varName, limit);
    }
}

int main() {
    setlocale(LC_ALL, "russian");
    int t = getCorrectInput('t', 7);
    int s = getCorrectInput('s', 15);
    int l = getCorrectInput('l', 255);
    cout << "Упакованная структура в 16-ричной системе: " << hex << pack(t, s,
l) << endl;
    return 0;
}
```

Программа 2:

```
#include <iostream>
using namespace std;

void unpackAndPrint(int packedHex) {
    cout << "Значение T в данной структуре десятичном виде: ";
    cout << dec << ((packedHex >> 13) & 0x7) << endl;
    cout << "Значение S в данной структуре десятичном виде: ";
    cout << dec << ((packedHex >> 8) & 0xF) << endl;
    cout << "Значение L в данной структуре десятичном виде: ";
    cout << dec << (packedHex & 0xFF) << endl;
}

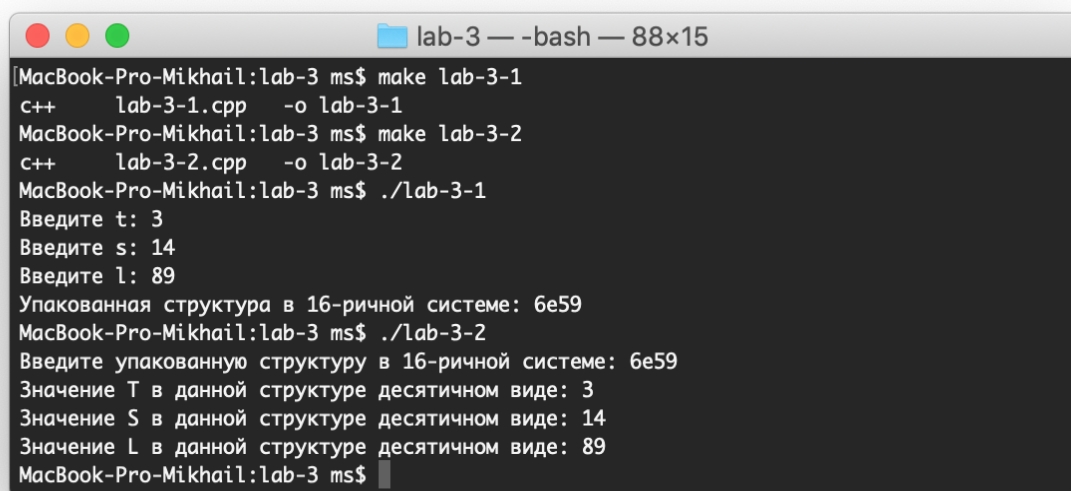
int main() {
    int packedHex;
    setlocale(LC_ALL, "russian");
    cout << "Введите упакованную структуру в 16-ричной системе: ";
    cin >> hex >> packedHex;

    if (
        cin.fail() ||
        ((packedHex >> 12) & 1) != 0 // Неверный тип
        // 12-й бит не 0
    ) {
        cout << "Некорректно введена структура" << endl;
        return 0;
    }

    unpackAndPrint(packedHex);
    return 0;
}
```

5. Пример выполнения программы

Ниже показан пример выполнения программы.



```
MacBook-Pro-Mikhail:lab-3 ms$ make lab-3-1
c++ lab-3-1.cpp -o lab-3-1
MacBook-Pro-Mikhail:lab-3 ms$ make lab-3-2
c++ lab-3-2.cpp -o lab-3-2
MacBook-Pro-Mikhail:lab-3 ms$ ./lab-3-1
Введите t: 3
Введите s: 14
Введите l: 89
Упакованная структура в 16-ричной системе: 6e59
MacBook-Pro-Mikhail:lab-3 ms$ ./lab-3-2
Введите упакованную структуру в 16-ричной системе: 6e59
Значение T в данной структуре десятичном виде: 3
Значение S в данной структуре десятичном виде: 14
Значение L в данной структуре десятичном виде: 89
MacBook-Pro-Mikhail:lab-3 ms$
```

Видно, что результаты расчётов совпадают с тестовыми данными.

6. Анализ результатов и выводы

К достоинствам программы можно отнести:

- Программа выполняет поставленную задачу и работает без ошибок (для корректных тестовых данных).
- Задание реализовано в виде отдельной функции.

Из недостатков можно отметить:

- Функция работает только для конкретного задания.