Лабораторная работа №4

По дисциплине «Информатика»

Тема: Перегрузка операций

Выполнил:

студент группы [EMPTY]

[YOUR NAME]

Принял:

[TEACHER NAME]

**Цель работы**: приобрести навыки в использовании перегрузки операций.

**Структура программы**

* deps.h - содержит макросы, что позволяет программе работать как на Windows так и на Unix-подобных платформах
* deps.cpp - содержит определения Windows-зависимых функций
* main.cpp - содержит интерфейс для работы с бинарными строками

**Ниже представлен код программы**

// deps.h

#ifndef \_\_DEPS\_H

#define \_\_DEPS\_H

# ifdef \_\_WIN32

# include <windows.h>

# include <conio.h>

# define KB\_ENTER 13

# else

# include <unistd.h>

# include <termios.h>

# include <sys/ioctl.h>

# define KB\_ENTER 10

int putch (int c);

int getch (void);

int kbhit (void);

# endif

#define KB\_ESCAPE 27

#endif

// deps.cpp

#include <cstdio>

#include "deps.h"

#ifndef \_\_WIN32

int putch (int c)

{

return putchar (c);

}

int kbhit(void) {

static const int STDIN = 0;

static bool initialized = false;

if (! initialized) {

// Use termios to turn off line buffering

termios term;

tcgetattr(STDIN, &term);

term.c\_lflag &= ~ICANON;

tcsetattr(STDIN, TCSANOW, &term);

setbuf(stdin, NULL);

initialized = true;

}

int bytesWaiting;

ioctl(STDIN, FIONREAD, &bytesWaiting);

return bytesWaiting;

}

int getch (void)

{

int ch;

struct termios oldtc;

struct termios newtc;

tcgetattr(STDIN\_FILENO, &oldtc);

newtc = oldtc;

newtc.c\_lflag &= ~(ICANON | ECHO);

tcsetattr(STDIN\_FILENO, TCSANOW, &newtc);

ch=getchar();

tcsetattr(STDIN\_FILENO, TCSANOW, &oldtc);

return ch;

}

#endif

// main.cpp

#include <iostream>

#include <string.h>

#include <cstdio>

#include <clocale>

#include "deps.h"

using namespace std;

/\* Bin2Dec

binary to (2) decimal - бинарное в деситичное

\*/

class Bin2Dec {

char \*binstr; //указатель на строку, которая хранит число

int count;

/\* convert binary string to decimal values. \*/

int bin2dec(char \*n)

{

char \*num = n;

int size = strlen (n);

int decval = 0;

int sign = 1;

int base = 1; /\* Initializing base value to 1, i.e 2^0 .\*/

if (n[0] == '1') {

sign = -1;

}

/\* binary convertion with 2's complement \*/

for (int i = size - 1; i > 0; --i) {

int last\_digit = (num[i]) - 48; /\* convert ascii to digit. \*/

decval += last\_digit \* base;

base = base \* 2;

}

return sign \* decval;

}

char \*dec2bin (int num)

{

int size = 20;

int arr[size];

char \*str = new char[size];

int i = 0, k = 0;

if (num < 0) {

num = -num;

str[k++] = '1';

}

else {

str[k++] = '0';

}

while (num > 0) {

arr[i] = num % 2;

num = num / 2;

i++;

}

for (int j = i - 1; j >= 0; j--) {

str[k++] = arr[j] + 48;

}

return str;

}

public:

Bin2Dec(int n = 100) //конструктор

{

count = 0;

binstr = new char[n]; //выделение памяти под строку

}

Bin2Dec(const Bin2Dec &c1) //конструктор копирования

{

binstr= new char[c1.count+1]; //выделение памяти под строку

strcpy (binstr,c1.binstr);

count = c1.count;

}

~Bin2Dec()

{

if (binstr) {

delete binstr;

}

}

friend istream & operator>>(istream &, Bin2Dec &);//дружественная функция для ввода строки

friend ostream & operator<<(ostream &, Bin2Dec &);//дружественная функция для ввода строки

int convert (void)

{

return bin2dec (binstr);

}

Bin2Dec& operator =(const Bin2Dec& c1) //перегруженный оператор присваивания

{

if (this!=&c1)

{

strcpy(binstr,c1.binstr); //копируем строку из аргумента функции в текущий объект

count=c1.count;

}

return \*this;

}

void operator -(const Bin2Dec& c1) //перегруженный оператор умножить

{

int d1, d2;

d1 = bin2dec (binstr); // перевели строку из текущего в вещественное число

d2 = bin2dec (c1.binstr); // перевели строку из аргумента функции в вещественное число

d1 -= d2;

binstr = dec2bin (d1); // перевели число обратно в бинарную строку

count = strlen (binstr); // получили новое значение count

}

void operator -=(const Bin2Dec& c1) //перегруженный оператор умножить

{

int d1, d2;

d1 = bin2dec (binstr); //перевели строку из текущего в вещественное число

d2 = bin2dec (c1.binstr); //перевели строку из аргумента функции в вещественное число

d1 -= d2;

binstr = dec2bin (d1); // перевели число обратно в бинарную строку

count = strlen (binstr); //получили новое значение count

}

void operator --(int a)

{

for (int i = count - 1; i >= 0; --i) {

if (binstr[i] == '1') {

binstr[i] = '0';

return ;

}

}

/\* flip all bits. \*/

for (int i = 0; i < count; ++i) {

binstr[i] = '1';

}

}

operator int (void)

{

return bin2dec(binstr);

}

};

istream & operator >>(istream & input, Bin2Dec &c1) //определение дружественной функции ввода строки

{

char key;

cout << "Введите бинарноe число (2's comp): ";

do {

key = getch(); //считываем символ с клавиатуры без вывода на акран

if (key == '0' || key == '1')

//проверяем, является ли перевый введенный символ цифрой или минусом

{

c1.binstr[c1.count++]=key; //сохраняем введенный символ в строке и увеличиваем счетчик числа введенных символов на 1

putch(key); //выводим символ на экран

}

} while (key != KB\_ENTER);

c1.binstr[c1.count]='\x0';

cout << "\n";

return input;

}

ostream & operator <<(ostream & output, Bin2Dec &c1)

{

int i = 0;

cout << "Ваше число (bin): ";

while (i<c1.count) {

putch (c1.binstr[i++]);

}

return output;

}

int main (void)

{

int dec;

Bin2Dec bin1, bin2;

/\* ввети и вывести 1-ое число. \*/

cin >> bin1;

cout << bin1 << endl;

cout << "Ваше число (dec): " << bin1.convert() << endl << endl;

/\* вычесть из 2-го 1-ое число. \*/

cout << "Декремент первого числа\n";

bin1--;

cout << bin1 << endl;

cout << "Ваше число (dec): " << bin1.convert() << endl << endl;

cout << endl << endl;

/\* ввети и вывести 3-е число. \*/

cin >> bin2;

cout << bin2 << endl;

cout << "Ваше число (dec): " << bin2.convert() << endl << endl;

/\* вычесть из 1-го 2-ое число. \*/

cout << "Вычитаем 1-ое число из 2-его\n";

bin2 -= bin1;

cout << bin2 << endl;

cout << "Ваше число (dec): " << bin2.convert() << endl << endl;

/\* превратить бинарные строки в целое число. \*/

cout << "Превращаем строку в деситичное число\n";

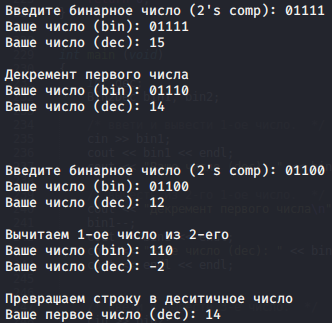
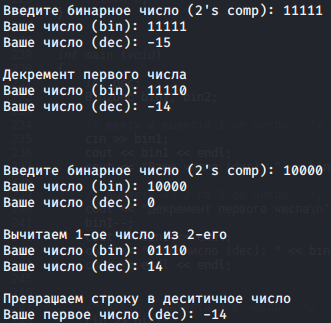
dec = int (bin1);

cout << "Ваше первое число (dec): " << dec << endl;

return 0;

}

Экранные формы с примерами работы программы

 Рис. 1 Рис. 2

На Рис. 1 программа считывает бинарную строку (дополнительный код) от пользователя и производит над этими вычисления. Такие как декремент числа, вычитание 2 чисел и операция сокращенного вычитания в C++ (-=). На данном рисунке изображена работы с положительными числами.

На Рис. 2 программа производит те же операции, но уже над отрицательными числами. Также производит конвертацию бинарной строки в десятичное число.

Заключение

В данной лабораторной работе я приобрел навыки в использовании перегрузки операций и продемонстрировал эти навыки в перегрузке операторов ввода/вывода, присвоения, вычитания и декремента в С++.