**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**



**МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**Факультет Информационных технологий**

***Кафедра Информатики и информационных технологий***

**направление подготовки**

**09.03.02 «Информационные системы и технологии»**

**ЛАБОРАТОРНАЯ РАБОТА № 1**

**Дисциплина:**

Объектно-ориентированное программирование

**Тема:**

Основы программирования на «С».

**Выполнил(а): студент(ка) группы 211-7210**

**Салов Д.К.**

(Фамилия И.О.)

**Дата, подпись** 28.02.22

(Дата) (Подпись)

**Проверил: \_\_*\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_***

(Фамилия И.О., степень, звание) **(Оценка)[[1]](#endnote-1)**

**Дата, подпись** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

**Замечания: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Москва**

**2021**

**Цель:**  Получить практические навыки в использовании в языка С для создания программ.

**Задания:**

1. Считая, что кубический фут равен 7,481 галлона, написать программу, запрашивающую у пользователя число галлонов и выводящую на экран эквивалентный объем в кубических футах.

// converts gallons to cubic feet #include <iostream> using namespace std;

#include <iostream>

using namespace std;

int main(){

float gallons, cufeet;

cout << "\nEnter quantity in gallons: ";

cin >> gallons; cufeet = gallons / 7.481;

cout << "Equivalent in cublic feet is " << cufeet << endl;

return 0;

}

2. Напишите программу, выводящую следующую таблицу:

1990 135

1991 7290

1992 11300

1993 16200

В программе использовать только один оператор с cout.

// generates table

#include <iostream>

#include <iomanip>

using namespace std;

int main(){

cout << 1990 << setw(8) << 135 << endl << 1991 << setw(8) << 7290 << endl

<< 1992 << setw(8) << 11300 << endl

<< 1993 << setw(8) << 16200 << endl;

return 0;

}

3. Напишите программу, генерирующую следующий вывод:

10

20

19

Используйте представление числа 10 в виде целой константы. Для вывода числа 20 воспользуйтесь одной из арифметических операций с присваиванием, а для вывода числа 19 — операцией декремента.

// exercises arithmetic assignment and decrement

#include <iostream>

using namespace std;

int main(){

int var = 10;

cout << var << endl; // var is 10

var \*= 2; // var becomes 20

cout << var-- << endl; // displays var, then decrements it

cout << var << endl; // var is 19

return 0;

}

4. Напишите программу, выводящую па экран ваше любимое стихотворение. Для разбиения на строчки используйте подходящую управляющую последовательность.

#include <iostream>

using namespace std;

int main(){

setlocale(LC\_ALL, "Russian");

cout << "Ты прозаик, я поэт -\n"

<< "Это наш с тобой дуэт.\n"

<< "Мы друг друга понимаем,\n"

<< "Стихи тоже сочиняем.\n";

return 0;

}

5. Библиотечная функция islower() принимает в качестве аргумента один символ (букву) и возвращает ненулевое целое значение в том случае, если буква является строчной, и нулевое, если буква является заглавной. Описание функции хранится в файле CTYPE.H. Напишите программу, которая принимает букву от пользователя, а затем выводит нулевое или ненулевое значение в зависимости от того, является ли буква строчной или нет.

#include <iostream>

using namespace std;

int main(){

char letter;

cin >> letter;

int output = islower(letter);

cout << output;

}

6. На биржевых торгах за 1 фунт стерлингов давали $1.487, за франк — $0.172. за немецкую марку — $0.584, а за японскую йену — $0.00955. Напишите программу, которая запрашивает денежную сумму в долларах, а затем выводит эквивалентные суммы в других валютах.

#include <iostream>

using namespace std;

int main(){

cout << "Enter $";

float bucks;

cin >> bucks;

cout << "pound - " << bucks / 1.487 << endl;

cout << "franc - " << bucks / 0.172 << endl;

cout << "german mark - " << bucks / 0.584 << endl;

cout << "yen - " << bucks / 0.00955 << endl;

return 0;

}

7. Температуру, измеренную в градусах по Цельсию, можно перевести в градусы по Фаренгейту путем умножения на 9/5 и сложения с числом 32. Напишите программу, запрашивающую температуру в градусах по Цельсию и отображающую ее эквивалент по Фаренгейту.

#include <iostream>

using namespace std;

int main(){

cout << "Enter temperature (C)...\n";

float C;

cin >> C;

cout << "Temperature (F) = " << C \* 9 / 5 + 32 << endl;

return 0;

}

8. Когда размер величины, выводимой на экран с помощью манипулятора setw(), оказывается меньше размера зарезервированного поля, по умолчанию незаполненные поля заполняются пробелами. Манипулятор setfill() принимает в качестве аргумента один символ, который замещает все пробелы на незаполненных позициях поля. Модифицируйте пример 2 так, чтобы символы, разделяющие пары значений из столбцов, были не пробелами, а точками, например

Москва........ 8425785

// generates table

#include <iostream>

#include <iomanip>

using namespace std;

int main() {

cout << setfill('.') << 1990 << setw(8) << 135 << endl

<< 1991 << setw(8) << 7290 << endl

<< 1992 << setw(8) << 11300 << endl

<< 1993 << setw(8) << 16200 << endl;

return 0;

}

9. Две дроби а/b и c/d можно сложить следующим образом: a/b + c/d = (a\*d + b\*c)/(b\*d)

Например, 1/4 + 2/3 = (1\*3 + 4\*2)/4\*3 = 11/12

Напишите программу, запрашивающую у пользователя значения двух дробей, а затем выводящую результат, также записанный в форме дроби. Взаимодействие программы с пользователем может выглядеть, например, следующим образом:

Введите первую дробь: 1/2 Введите вторую дробь: 2/5 Сумма равна 9/10

Вы можете использовать тот факт, что операция извлечения >> может считывать более одного значения за раз: cin >> а >> dummychar >> b;

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_ALL, "Russian");

int a, b, c, d;

cout << "Введите первую дробь: \n";

cin >> a;

cout << "/\n";

cin >> b;

cout << "Введите вторую дробь:\n";

cin >> c;

cout << "/\n";

cin >> d;

cout << "Сумма: " << a \* d + b \* c << '/' << b \* d;

return 0;

}

10. Устаревшая денежная система Великобритании состояла из фунтов, шиллингов и пенсов. 1 фунт был равен 20 шиллингам, а 1 шиллинг — 12 пенсам.

Для записи использовалась система, состоящая из знака £ и трех десятичных значений, разделенных точками. Например, запись £5.2.8 обозначала 5 фунтов, 2 шиллинга и 8 пенсов (пенс — множественное число от пенни). Современная денежная система, принятая в 50-е годы XX века, состоит только из фунтов и пенсов, причем один фунт равен 100 пенсам. Такой фунт называют десятичным. Таким образом, в новой денежной системе указанная сумма будет обозначаться как £5.13 (если быть точнее, £5.1333333). Напишите программу, которая будет преобразовывать сумму, записанную в старом формате (фунты, шиллинги, пенсы), в новый формат (фунты, пенсы). Форматом взаимодействия программы с пользователем может являться следующий:

Введите количество фунтов: 7

Заедите количество шиллингов: 17

Введите количество пенсов: 9 Десятичных фунтов: £7.89

В большинстве компиляторов для представления знака £ используется десятичный код 156. Некоторые компиляторы позволяют скопировать знак фунта прямо из таблицы символов Windows.

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_ALL, "Russian");

int f, s, p;

cout << "введите количество фунтов : ";

cin >> f;

cout << "введите количество шиллингов: ";

cin >> s;

cout << "введите количество пенсов: ";

cin >> p;

cout << "Десятичных фунтов: " << f + ((float(s) \* 12 + p) / 240) << endl;

return 0;

}

11. По умолчанию форматирование вывода производится по правому краю поля. Можно изменить форматирование текста на левостороннее путем использования манипулятора setiosflags(ios::left) (не беспокойтесь о смысле новой формы записи, встретившейся в манипуляторе). Используйте этот манипулятор вместе с setw() для того, чтобы произвести следующий вывод:

Фамилия Имя Адрес Город

Петров Василий Кленовая 16 Санкт-Петербург

Иванов Сергей Осиновая 3 Находка

Сидоров Иван Березовая 21 Калининград

#include <iostream>

#include <iomanip>

using namespace std;

int main() {

cout << setiosflags(ios::left) << setw(15) << "Lastname" << setw(15) <<

"Name" << setw(15) << "City" << endl <<

setw(15) << "Ivanov" << setw(15) << "Oleg" << setw(15) << "Moscow" << endl <<

setw(15) << "Petrov" << setw(15) << "Sergei" << setw(15) << "Nahodka" << endl <<

setw(15) << "Sidorov" << setw(15) << "Vasiliy" << setw(15) << "Kaliningrad" << endl;

return 0;

}

12. Напишите программу, выполняющую действия, обратные тем, которые описаны в упражнении 10, то есть запрашивающую у пользователя сумму, указанную в десятичных фунтах, и переводящую ее в старую систему

фунтов, шиллингов и пенсов. Пример взаимодействия программы с пользователем может выглядеть так:

Введите число десятичных фунтов: 3.51

Эквивалентная сумма в старой форме записи: £3.10.2

Обратите внимание на то, что если вам придется присваивать вещественное значение (например, 12,34) переменной целого типа, то его дробная часть (0,34) будет потеряна, а целая переменная получит значение 12. Чтобы избежать предупреждения со стороны компилятора, используйте явное преобразование типов. Можно использовать операторы, подобные приведенным ниже:

float decpounds: // сумма в десятичных фунтах int

pounds: // сумма в старых фунтах float decfrac:

// десятичная дробная часть pounds = static\_cast<int>(decpounds): //отбрасывание

//дробной части

decfrac = decpounds - pounds; //прибавление дробной части

Чтобы получить число шиллингов, следует умножить на 20 значение переменной decfrac. Аналогичным образом можно получить число пенсов.

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_ALL, "Russian");

float newd;

int pounds, shillings, pence;

float d;

cout << "В новых фунтах ";

cin >> newd;

pounds = static\_cast<int>(newd);

d = 240 \* (newd - pounds);

pence = (static\_cast<int>(d)) % 12;

shillings = static\_cast<int>((d - pence) / 12);

cout << "В старых тугриках" << pounds << "." << shillings << "." << pence << endl;

}

1. [↑](#endnote-ref-1)