

Содержание

ВВЕДЕНИЕ.....	5
ЛАБОРАТОРНАЯ РАБОТА 1. ЗНАКОМСТВО СО СРЕДОЙ DEV-C++.....	6
ЛАБОРАТОРНАЯ РАБОТА 2. ПРОСТЕЙШИЕ ПРОГРАММЫ НА ЯЗЫКЕ C/C++	10
ЛАБОРАТОРНАЯ РАБОТА 3. ГРАФИКА В СРЕДЕ DEV-C++ (ПЕРВОЕ ЗНАКОМСТВО).....	15
ЛАБОРАТОРНАЯ РАБОТА 4. ОПЕРАЦИИ С ЦЕЛЫМИ И ВЕЩЕСТВЕННЫМИ ЧИСЛАМИ	18
ЛАБОРАТОРНАЯ РАБОТА 5. ОПЕРАТОР ВЕТВЛЕНИЯ.....	20
ЛАБОРАТОРНАЯ РАБОТА 6. ОПЕРАТОР SWITCH.....	23
ЛАБОРАТОРНАЯ РАБОТА 7 ЦИКЛ С ПАРАМЕТРОМ.....	26
ЛАБОРАТОРНАЯ РАБОТА 8*. КАК НАРИСОВАТЬ ЗАБОР	28
ЛАБОРАТОРНАЯ РАБОТА 9 ⁰ ЗВЕЗДНОЕ НЕБО И ПРОЧИЕ СТРАННОСТИ.....	31
ЛАБОРАТОРНАЯ РАБОТА 10 ЦИКЛ С ПАРАМЕТРОМ: ТИПОВЫЕ АЛГОРИТМЫ.....	33
ЛАБОРАТОРНАЯ РАБОТА 11 ПРОЕКТИРОВАНИЕ ПРОГРАММ И ПРОЦЕДУРЫ.....	37
ЛАБОРАТОРНАЯ РАБОТА 12 ЦИКЛЫ С УСЛОВИЕМ.....	42
ЛАБОРАТОРНАЯ РАБОТА 13. ПОСТРОЕНИЕ ГРАФИКОВ ФУНКЦИЙ.....	45
ЛАБОРАТОРНАЯ РАБОТА 14. ИСПОЛЬЗОВАНИЕ ПРОЦЕДУР.....	49
ЛАБОРАТОРНАЯ РАБОТА 15. ПОДПРОГРАММЫ - ФУНКЦИИ.....	51
ЛАБОРАТОРНАЯ РАБОТА 16. КАК РАБОТАТЬ С МАССИВАМИ: ПЕРВЫЕ ШАГИ.....	54
ЛАБОРАТОРНАЯ РАБОТА 17*. СОРТИРОВКА МАССИВОВ.....	58
ЛАБОРАТОРНАЯ РАБОТА 18. ОСНОВНЫЕ ОПЕРАЦИИ СО СТРОКОВЫМИ ДАННЫМИ.....	61
ЛАБОРАТОРНАЯ РАБОТА 19. РАБОТА С ТЕКСТОВЫМИ ФАЙЛАМИ: ЧТЕНИЕ И ЗАПИСЬ	65
ЛАБОРАТОРНАЯ РАБОТА 20. РАБОТА С ДАННЫМИ ИЗ ТЕКСТОВЫХ ФАЙЛОВ.....	68
ЛАБОРАТОРНАЯ РАБОТА 21*. МНОГОМЕРНЫЕ МАССИВЫ.....	71
ЛАБОРАТОРНАЯ РАБОТА 30. ОСНОВЫ РАБОТЫ С ТИПИЗИРОВАННЫМИ ФАЙЛАМИ	73
ЛАБОРАТОРНАЯ РАБОТА 31. ЗАПИСИ	ОШИБКА! ЗАКЛАДКА НЕ ОПРЕДЕЛЕНА.
ЛАБОРАТОРНАЯ РАБОТА 32. ФАЙЛЫ ПРЯМОГО ДОСТУПА	ОШИБКА! ЗАКЛАДКА НЕ ОПРЕДЕЛЕНА.
ЛАБОРАТОРНАЯ РАБОТА 33*. ЧТО МОЖНО ДЕЛАТЬ С ФАЙЛАМИ И КАТАЛОГАМИ.....	ОШИБКА! ЗАКЛАДКА НЕ ОПРЕДЕЛЕНА.
ЛАБОРАТОРНАЯ РАБОТА 34. СОЗДАНИЕ ПРОГРАММ, УПРАВЛЯЕМЫХ СОБЫТИЯМИ....	ОШИБКА! ЗАКЛАДКА НЕ ОПРЕДЕЛЕНА.

ЛАБОРАТОРНАЯ РАБОТА 35. РАБОТА СО ЗВУКОМ. СПРАЙТЫ..... ОШИБКА! ЗАКЛАДКА НЕ ОПРЕДЕЛЕНА.

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ..... ОШИБКА! ЗАКЛАДКА НЕ ОПРЕДЕЛЕНА.

Лабораторная работа 1.

Знакомство со средой Dev-C++

1. Как начать работу со средой Dev-C++

Сначала подготовимся к работе.

Задание 1

Попробуйте на диске найти папку Рабочая. Если таковой нет, создайте ее. В папке Рабочая создайте папку C++, а в ней – папку с Вашей фамилией. Далее эту папку будем называть «Вашей папкой».

1.1. Запуск среды

Задание 2

Запустите среду программирования Dev-C++ (Пуск → Все программы → BloodShed Dev-C++ → Dev-C++).

Если среда запустилась, перед Вами появится окно, приведенное на рис. 1.

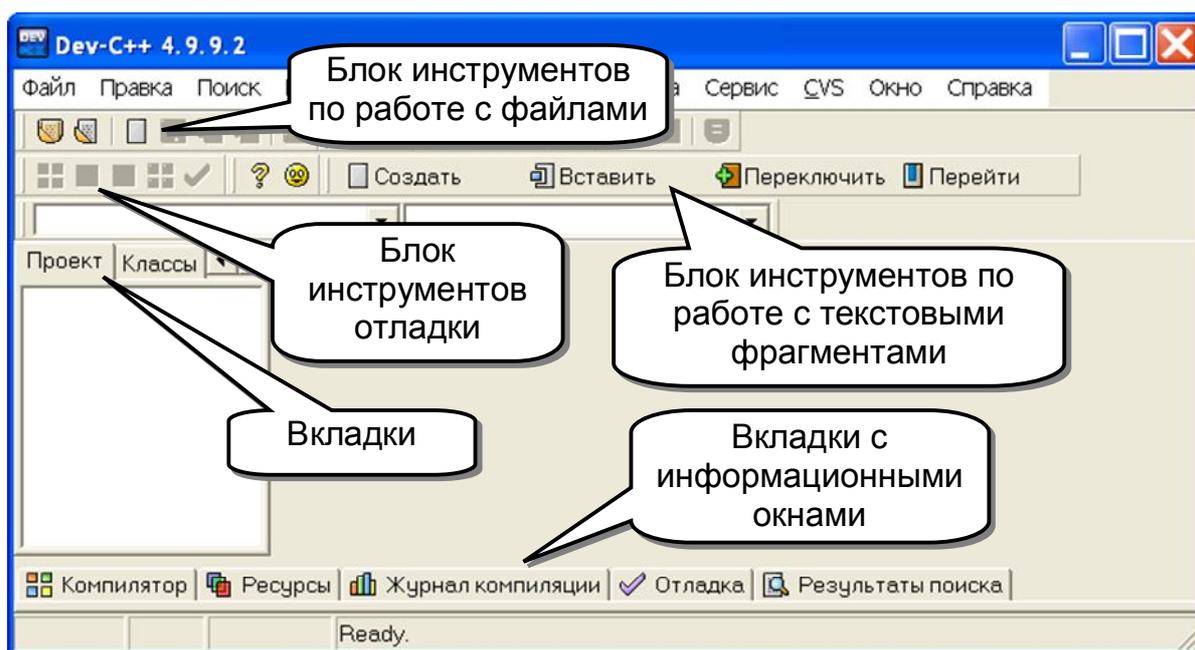


Рис. 1. Структура окна среды Dev-C++

Задание 3

Изучите комментарии, приведенные на рис. 1.

Скопируйте в свою папку все файлы из папки Учебная\Ex_C++.

1.2. Запуск программы в консольном режиме

1.2.1. Как создать программу и запустить ее на выполнение

Задание 4

Откройте программы **ex1.cpp**, **ex2.cpp** и **ex3.cpp** из вашей папки. Программы (точнее – тексты программ) открываются обычным образом: **Файл** → **Открыть**, или кнопка **Открыть** (📁) из блока инструментов по работе с файлами.

Обратите внимание, что имена открытых файлов появляются на закладках в верхней части окна. Переход от программы к программе реализуется с помощью щелчка мыши на нужную закладку.

Задание 5

Запустите каждую из загруженных программ. Для запуска программы следует:

- перейти на страницу с текстом данной программы. Для этого достаточно щелкнуть по закладке с названием нужной программы в верхней части рабочего окна;
- текст программы сначала нужно откомпилировать (т.е. преобразовать в готовую программу в двоичном коде – файл с расширением **exe**). Для этого следует щелкнуть по кнопке **Скомпилировать** (🔧) или нажать на клавиатуре **Ctrl+F9**;
- и, наконец, заключительный шаг: щелкнуть по кнопке **Выполнить** (▶) или нажать **Ctrl+F10**.

👉 В среде *Dev-C++* есть возможность выполнить эти две операции сразу, нажав на кнопку **Скомпилировать и выполнить** (🔧▶) или **F9**.

Задание 6

Просмотрите текст каждой из программ. Постарайтесь понять текст программы.

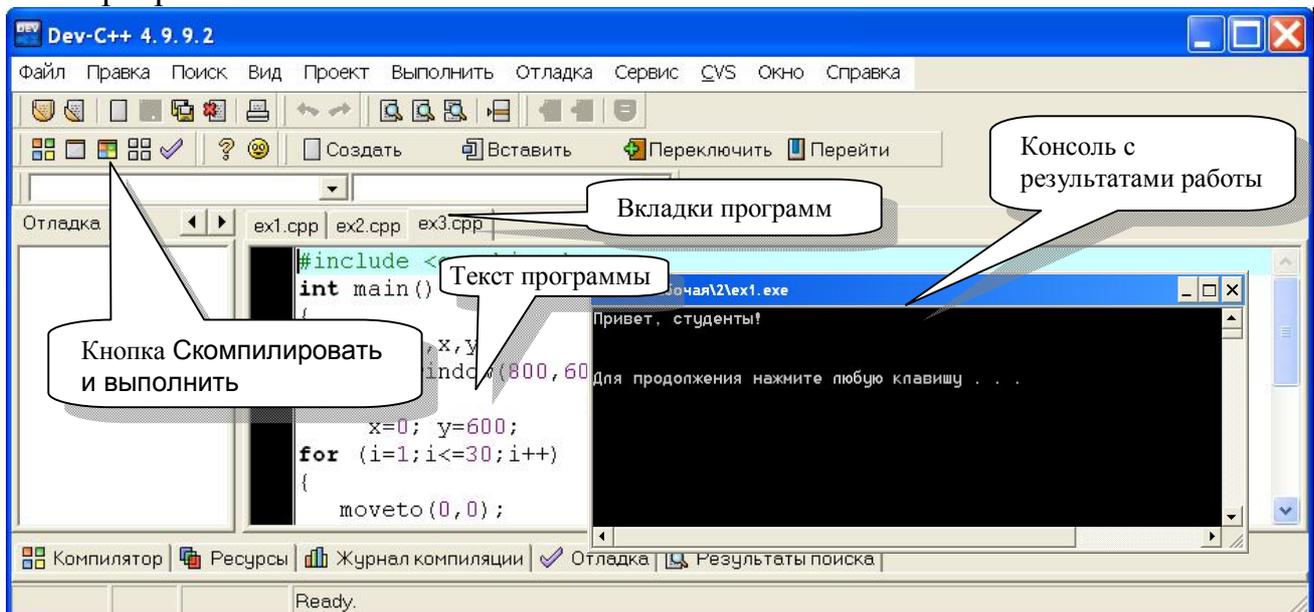


Рис. 2. Основные компоненты, используемые при запуске программы

1.2.2. Как убрать имеющуюся страницу

Если изменений в текстах Вы не производили, а текст больше не нужен для текущей работы, страницу можно убрать с рабочего стола. Для этого достаточно щелкнуть правой клавишей мыши по закладке и в контекстном меню выбрать либо пункт **Заккрыть** (т.е. закрыть текущее окно), либо **Заккрыть все**, т.е. закрыть все окна. И, наконец, можно выбрать пункт **Заккрыть все**, кроме этого (т.е. того, по которому щелкнули мышкой).

Задание 7

Закройте все неактивные страницы. Выясните, можно ли закрыть оставшуюся страницу.

1.2.3. Как создать страницу

Следует иметь в виду, что среда Dev-C++ ориентирована на работу в двух режимах: создание проектов, реализуемых с помощью объектно-ориентированных и визуальных технологий, и работе в консольном режиме.

Для освоения языка мы будем использовать последний из этих режимов.

Фактически новая страница – это новый документ для консольной программы. Для создания нового документа используется пункт меню **Файл** → **Создать** → **Исходный файл**. В этом случае появится новая страница, а информация с этой страницы будет записываться в файл с именем **Безымянный1.cpp**. В случае, если в новом листе началась разработка новой программы (т.е. просто начали набирать некий текст), то по попытке создать новую страницу будет создан файл с именем **Безымянный2.cpp** и т.д.

Задание 8

Создайте новую страницу. После этого наберите текст следующей программы:

```
#include <stdio.h>
#include <conio.h>
#include <iostream>
main()
{ int a;
  printf("Моя первая программа:\n");
  printf("Введите число 5:\n");
  scanf ("%d", &a); // ввод числа
  if (a==5) printf("Вы неожиданно послушны\n");
  else printf("Хорошо, что это не светофор... \n");
  system( " pause " ); // ожидаем нажатия кнопки;
}
```

Запустите программу. Выясните, что делает эта программа. Постарайтесь разобраться в ее тексте.

1.3. Среда Dev-C++ как рабочий стол программиста

С точки зрения внешнего наблюдателя программирование – это процесс, схожий с писательским трудом. Программисту приходится много писать, опро-

бовать созданное на «читателях» (компьютерах и людях, использующих эти компьютеры), выяснять, что тем «не нравится», исправлять написанное, снова проверять – и так до тех пор, пока не получится нечто, являющееся законченным продуктом.

Что требуется программисту для работы? Понятно, что стол, листы бумаги, ручка и компьютер с необходимыми программами. Все это и предлагает среда программирования Dev-C++.

Когда Вы запускаете среду Dev-C++, Вы получаете пустой «стол». На этот стол можно положить пустой лист «бумаги» (страницу) либо уже подготовленные «листы» (файлы с текстами). На этих «листах» можно писать любой текст. Для этого в среду Dev-C++ включен текстовый редактор. Все тексты можно сохранить в файлах на диске.

Если текст представляет собой программу на языке программирования C++, то можно дать команду на компиляцию (т.е. проверку правильности, а также преобразования в машинный код) и исполнение написанной программы. Ранее этот шаг мы называли «запустить программу». Дело в том, что в современных средах программирования шаг компиляции, проверки и запуска на выполнения объединены, и если ошибок в тексте программы нет, то программа действительно просто выполняется.

Задание 9

Сохраните первую вашу программу в вашей папке. Для этого выберите пункт меню **Файл**, там – режим **Сохранить как...**, найдите свою папку, задайте имя (например FirstPrg.pas) и щелкните по кнопке **Сохранить**.

Контрольные вопросы

1. Как запустить среду Dev-C++?
2. Как создать новую страницу? Как открыть текст ранее созданной программы?
3. Как закрыть текущую страницу с текстом программы? Можно ли закрыть все неактивные страницы?
4. Пусть у Вас открыто несколько программ. Как сделать активной какую-либо из них?
5. Как исполнить программу, размещенную на какой-либо странице?
6. Чем ограничивается тело программы на языке C++?
7. Какую команду (процедуру) следует использовать в программе для вывода текста?

 *Пригласите преподавателя и продемонстрируйте ему вашу программу. Будьте готовы ответить на его вопросы. После этого сохраните все открытые программы и выйдите из среды Dev-C++.*

 *Если вы обучаетесь в условиях использования дистанционных образовательных технологий, просто вышлите файл с подготовленными Вами программой тьютору (если это предусмотрено технологической картой).*

Лабораторная работа 2.

ПРОСТЕЙШИЕ ПРОГРАММЫ НА ЯЗЫКЕ C/C++

1. Структура программы на языке C/C++

Программа на языке C++ имеет следующую структуру:

<pre>#include <ctdio.h> ... int main() { Описание данных; Оператор 1; Оператор 2; ... Оператор N; }</pre>	<ul style="list-style-type: none">☞ подключение библиотек;☞ заголовок программы;☞ начало исполняемой части программы;☞ текст («тело») программы;☞ конец программы.
---	--

Обратите внимание:

- (1) операторы отделяются друг от друга ТОЧКОЙ С ЗАПЯТОЙ;
- (2) после заголовка программы точка с запятой не ставится;
- (3) программа начинается с открывающейся фигурной скобки и заканчивается закрывающейся фигурной скобкой.

2. Простейшие операторы

2.1. Резервирование памяти для работы или описание переменных

Самое главное действие, которое выполняет компьютер, – это запись информации в память. Суть действия проста: необходимо указать компьютеру, КУДА записывать и ЧТО записывать.

Для определения того, КУДА записывать, необходимо сначала дать указание компьютеру ЗАРЕЗЕРВИРОВАТЬ место в памяти. Выполняется это в *блоке описания данных* после слова VAR:

Int a; целочисленный знаковый тип данных размером в 4 байта. Int – сокращение от английского «integer» – «целое». Это наиболее распространённый целочисленный тип данных. Он позволяет хранить значения от –2'147'483'648 до 2'147'483'647.

short int a; сокращённый вариант int, имеет размер 2 байта и позволяет хранить значения от -32'768 до 32'767.

 **Идентификатор** - это последовательность букв и/или цифр, начинающаяся с буквы. Имя может содержать и знак подчеркивания: _

 **ВНИМАНИЕ!** Слова, зарезервированные в языке C, не могут использоваться в качестве имени блока памяти!

Например, посмотрев на схему структуры программы, Вы уже можете определить для себя несколько таких слов:

if, int, float, while.

☞ **ВНИМАНИЕ!** Блок описание данных предлагает компьютеру **ВЫДЕЛИТЬ (ЗАРЕЗЕРВИРОВАТЬ)** память для работы программы. При этом запись информации в блок **НЕ ПРОИСХОДИТ**, там находится мусор!

Для начала Вы можете воспользоваться следующими стандартными (т.е. «понимаемыми» компилятором) типами для описания переменных:

int - для целых чисел в диапазоне от -2147483648 до 2147483647;

short int - для целых чисел в диапазоне от -32'768 до 32'767;

Char - целочисленный знаковый тип данных размером в 1 байт. Char – это сокращение от английского «character» – «символ». Дело в том, что в переменной типа char удобно хранить символы первой (английской) половины кодов ASCII. Однако, это лишь удобное обозначение, и в переменной типа char можно хранить любое целое число от -128 до 127.

Char - При обработке текстовой информации неудобно каждый раз писать ASCII-код символа. Гораздо логичнее поручить поиски нужного кода компилятору. Для этого символ с искомым кодом помещается в одинарные кавычки. Например, 'a' — это то же самое, что и 97. а такое объявление:

```
char c='0';
```

проинициализует переменную с значением 48. Символ, заключённый в одинарные кавычки называется *символьной константой*.

float - вещественное одинарной точности;

double - вещественное двойной точности;

long double - вещественное расширенной точности. Не все сопроцессоры могут работать с числом длиной в 10 байт, поэтому иногда long double расширяют до 12 байт, заполняя неиспользуемые разряды нулями;

string - для строк текста, содержащих не более 255 символов.

2.2. Запись данных в память, или оператор присваивания

В предыдущем пункте Вам фактически предложено работать с данными трех типов: целыми, вещественными и строковыми. Следует запомнить:

(1) целые числа в программах записываются так же, как принято в математике. Например: 345 , -12222 ;

(2) вещественные числа могут записываться двумя способами: первый – аналогичный математической записи: -123.234, 92929.3456 (обратите внимание,

что здесь используется десятичная ТОЧКА, а не запятая), второй – в так называемом «плавающем» виде (правильное название: «десятичное число с плавающей точкой»). Например, число $1,23 \cdot 10^{12}$ может быть записано следующим образом:

1.23E+12

Здесь «E+12» читается как «умножить на десять в двенадцатой степени»;

 **Оператор присваивания** предназначен для указания компьютеру **ЗАПИСИ ДАННЫХ В КОНКРЕТНЫЙ БЛОК ПАМЯТИ**. Общий вид оператора:
<имя блока памяти>=<значение выражения>

Пример 1.

```
int main()           // заголовок программы
{
  int a = 7;         // описали переменную целого типа и задали ей начальное значение
  float x;           // описали переменную вещественного типа
  x = a / 4;          // поделили значение переменной a на 4
  x = 4 / a;          // поделили 4 на значение переменной a
  x = float(a) / 4; // поделили значение a, преобразованное в вещественное число, на 4
  x = 1.*a / 4;      // поделили значение a, преобразованное в вещественное число, на 4
}
```

Как следует из определения оператора присваивания, в правой части оператора присваивания может быть и **ВЫРАЖЕНИЕ**. В этом случае в блок памяти (*переменную*) записывается **ЗНАЧЕНИЕ** выражения.

Допустимые арифметические операции языка:

<i>Сокращённая запись</i>	<i>Полная запись</i>
a++;	a=a+1;
a--;	a=a-1
a+=b;	a=a+b;
a-=b;	a=a-b;
a*=b;	a=a*b;
a/=b;	a=a/b;
a %= b;	a = a % b; // остаток от деления

2.3. Вывод данных на экран дисплея

Примеры программ, приведенных выше, иллюстрируют команду присваивания, но для Вас их выполнение на компьютере бессмысленно. Ведь программа записывает данные в память компьютера и заканчивает работу. Вы ничего не увидите. Чтобы увидеть, в языке есть оператор *вывода* **printf**.

Задание 10

Считайте с диска файл **ex2.cpp**. Разберитесь в тексте программы.

Измените программу так, чтобы она вводила 4 числа и вычисляла их сумму.

☞ Оператор *printf* размещен в библиотеке `<stdio.h>`!

Если Вы обратили внимание, здесь рассматривается два языка: C и C++. C++ – это более развитая (расширенная) версия языка C. В ней есть достаточно много дополнений и других подходов. В частности, оператор `cout` обеспечивает выходной поток на экран монитора.

Задание 11

Считайте с диска файл **ex1.cpp**. Разберитесь в тексте программы. Запишите в тетрадь название библиотеки, которая содержит оператор `cout`.

Составьте программу, которая выводит на экран ваши фамилия, имя, и отчество через строчку.

В языке C++ есть непечатаемые символы (такие как переход на новую строку), которые в кавычках написать невозможно. Для того, чтобы их записывать, существуют *escape-последовательности*. *Escape-последовательность* – это несколько символов, трактуемые как один. В языке C++ все *escape-последовательности* начинаются с символа `\`, после которого обязательно должен следовать какой-то другой. В одиночку этот символ смысла не имеет. Наиболее распространённые *escape-последовательности* в C++ таковы:

Символ	Значение
<code>\n</code>	Переход на новую строку (от англ. new)
<code>\t</code>	Табуляция (от англ. tab)
<code>\v</code>	Вертикальный отступ (от англ. vertical)
<code>\b</code>	Возврат на один символ назад (от англ. backspace)
<code>\r</code>	Возврат в начало строки (от англ. rewind)
<code>\\</code>	Символ <code>\</code>
<code>\'</code>	Символ <code>'</code>
<code>\"</code>	Символ <code>"</code>
<code>\0</code>	Нуль-символ (символ с кодом 0)

Задание 12

Модифицируйте программу **ex1.cpp** так, чтобы она после вывода сообщения обеспечивала пропуск четырех пустых строк.

2.4. Форматированный ввод и вывод информации

Формат – символьная строка, которая показывает, какие числа вводятся (выводятся). В языке C++ используются следующие форматы:

%d – целое число;

%f – вещественное число;

%c – 1 символ;

`%s` – символьная строка.

Задание 13

Составьте программу, которая вводит четыре числа и вычисляет значение выражения $a/b+c/d$. Например, если вводятся числа 4, 2, 9, 3, то на экране должно появиться выражение: $4/2+9/3=5$ (пример – в программе `ex2.cpp`).

3. Задачи для самостоятельной работы

1. Известны значения двух переменных, содержащие числа. Напишите программу, меняющую значения этих переменных между собой без использования третьей, буферной переменной.
2. Вычислите объем и площадь поверхности параллелипипеда. Определите, какие данные должны быть введены, какие – выданы на экран.
3. Составьте программу определения корней квадратного уравнения, имеющего решения.
4. Составьте программу определения идеального веса человека, если он считается по формуле: *идеальный вес (кг) = рост (см) – 100*.
5. Средняя наценка на товар составляет 18%. Пусть известна розничная цена товара. Вычислите оптовую цену.
6. Даны координаты двух точек. Вычислите расстояние между ними.
7. Вы положили деньги в банк под 15% годовых. Составьте программу определения общей суммы вклада через заданное число месяцев.
8. Длина некоторого отрезка составляет p метров. Напишите программу перевода ее в русскую неметрическую систему. Указание: 1 верста = 500 сажень, 1 сажень = 3 аршина, 1 аршин = 16 вершков, 1 вершок = 44,45 мм.
9. Составьте программу, которая вычисляет площадь равностороннего треугольника.
10. Напишите программу вычисления стоимости покупки, состоящей из нескольких карандашей, линеек и тетрадей. Их количество и цену задать вводом. Ответ вывести в виде:
Сумма к оплате: ... руб ... коп.
11. Напишите программу, переводящую время, указанное в минутах, во время в часах и минутах.

4. Вопросы для контроля

1. Для чего предназначена команда ввода?
2. Чем отличаются и что общего у команд ввода и присваивания?
3. В чем разница между командой присваивания и равенством в математике?

 *Пригласите преподавателя и продемонстрируйте ему все выполненные Вами задания. После этого сохраните все программы и выйдите из среды Dev-C++.*

 *Если вы обучаетесь в условиях использования дистанционных образовательных технологий, просто вышлите все файлы с подготовленными Вами программами тьютору (если это предусмотрено технологической картой).*

Лабораторная работа 3.

ГРАФИКА В СРЕДЕ DEV-C++ (ПЕРВОЕ ЗНАКОМСТВО)

В консольном режиме можно выводить не только текстовые сообщения, но и графические изображения. Базовой библиотекой, обеспечивающей реальный графический режим, является библиотека `<graphics.h>`.

1. Подключение библиотеке `<graphics.h>`

Вам уже известно, что библиотека подключается с помощью команды `#include`. Процедуры рисования графики размещены в библиотеке `<graphics.h>`. Поэтому, если требуется чего-то нарисовать, прежде всего требуется подключить графическую библиотеку. Подключение данной библиотеки выполняется следующим образом:

```
#include <graphics.h>;
```

...

Процедуры рисования и сама технология рисования реализована с помощью технологии Borland Graphics Interface (BGI) for Windows и весьма похожи на графику, реализованную в среде PascalABC.

Задание 14

Откройте веб-страничку с описанием библиотеки BGI для Dev-C++ по адресу <http://www.cs.colorado.edu/~main/cs1300/doc/bgi/>. Постарайтесь вспомнить, с помощью каких процедур можно нарисовать окружность, прямоугольник и линию, как можно закрасить замкнутый контур.

☞ *Обратите внимание, что система координат в графическом окне отличается от привычной – ось Y направлена не вверх, а вниз (Ошибка! Источник ссылки не найден.)!*



Рис. 3. Система координат в графическом окне

Задание 15

Считайте с диска файл с именем `ex3.cpp`. Он представляет собой программу рисования линии на экране дисплея. Запустите эту программу – и вы увидите на экране красную диагональную сплошную линию. Модифицируйте программу так, чтобы:

- а) линия располагалась вертикально;
- б) линия располагалась по диагонали экрана;
- в) на экране рисовались две диагонали – одна красного цвета, а другая – желтого.

Задание 16

Составьте программу рисования на экране дисплея квадрата и окружности. При этом окружность должна быть:

- а) вписана в квадрат;
- б) описана около квадрата.

Задание 17

Нарисуйте на экране дисплея:

- а) российский флаг 300x200 пиксель (горизонтальные полосы: белая, синяя, красная - одинаковой ширины);
- б) французский флаг 300x200 пиксель; (вертикальные полосы: синяя, белая, красная - одинаковой ширины);
- б) японский флаг 300x200 пиксель (красный круг в центре белого прямоугольника).

Задание 18

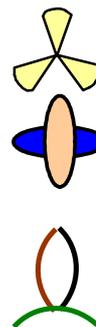
Модифицируйте составленные программы рисования:

- а) российского флага;
 - б) французского флага;
 - в) японского флага,
- так, чтобы размеры и место на экране вводились с клавиатуры.

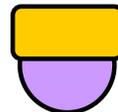
3. Задачи для самостоятельной работы

Составьте программу, которая:

- 1) рисует закрашенное кольцо (●). Радиус внешней окружности следует задавать в диалоге;
- 2) выводит в центр окна Вашу фамилию. Используйте для этого процедуру TextOut. При этом текст должен быть жирным, размер – 26 пт, цвет – зеленый;
- 3) рисует прямоугольник со скругленными углами. Необходимо, чтобы этот прямоугольник располагался в центре экрана, был закрашен любым цветом, толщина границы – не менее 5 точек, цвет границы – фиолетовый; цвет фона окна – черный;
- 4) рисует три закрашенных сектора окружности, расположенных в центре графического окна следующим образом:
- 5) рисует два эллипса, расположенных в центре окна размером 250×250 точек следующим образом:
- 6) рисует три дуги толщиной 5 точек, яркого цвета, расположенные в окне 150×150 точек следующим образом:



7) рисует фигуру, состоящую из окружности и прямоугольника с закругленными углами. Толщина линий – 5 точек, яркого цвета, такой же, как на рисунке, штриховки. Фигура расположена в центре окна 250×150 точек.



8) рисует треугольник в центре графического окна. При этом окно должно быть 200×200 точек, цвет фона – **clGold**, а треугольник нарисован с помощью процедур **MoveTo** и **LineTo**.

☞ *Пригласите преподавателя и продемонстрируйте ему все выполненные Вами задания. После этого сохраните все программы и выйдите из среды Dev-C++.*

☞ *Будьте готовы ответить на вопросы. Если вы обучаетесь в условиях использования дистанционных образовательных технологий, просто вышлите все файлы с подготовленными Вами программами преподавателю (если это предусмотрено технологической картой).*

Лабораторная работа 4.

ОПЕРАЦИИ С ЦЕЛЫМИ И ВЕЩЕСТВЕННЫМИ ЧИСЛАМИ

Задание 19

Вспомните, какие операции можно выполнять с целыми числами. Для этого:

- а) загрузите программу **ex_5_1.cpp** и запустите ее на выполнение;
- б) запишите в тетрадь новую для вас информацию;
- в) составьте программу, которая позволит для любых заданных целых чисел найти разность квадратов и квадрат разности.

Задание 20

Составьте программу, которая позволит определить доходы от вкладов в двух банках. Программа должна обеспечивать следующий диалог:

Здравствуй. Задайте размер Вашего вклада : 120000
На сколько месяцев Вы делаете вклад: 10
Сколько процентов годовых Вам обещано в банке 1: 355
Сколько процентов годовых Вам обещано в банке 2: 400
Через 10 месяцев на Вашем счете в банке 1 будет 475000
Через 10 месяцев на Вашем счете в банке 2 будет 520000

Задание 21

Вспомните, какие операции можно выполнять с вещественными числами. Для этого:

- а) загрузите программу **ex_5_2.cpp** и запустите ее на выполнение;
- в) запишите в тетрадь новую для вас информацию;
- в) запишите в тетрадь новую для вас информацию;
- г) составьте программу, которая по заданным сторонам треугольника определяет его площадь (напомним, что это можно сделать с помощью формулы Герона:

$$S = \sqrt{p(p-a)(p-b)(p-c)},$$

где p - полупериметр, a , b , c - стороны треугольника);

- д) результат выведите, используя форматированный вывод.

Задачи для самостоятельной работы

1. Составить программу, которая:
 - а) определяет целое количество недель, которое осталось до Нового года. Для простоты считать количество дней в месяце равное 30 дням;
 - б) вводит двухзначное число и определяет сумму его цифр;
 - в) вводит трехзначное число и выводит цифры этого числа в обратном порядке;
 - г) вводит трехзначное число и определяет сумму его цифр.
2. Даны координаты вершин треугольника. Вычислите его площадь и периметр.
3. Напишите программу, переводящую время, указанное в минутах, во время в часах и минутах.

4. Ипотека обычно берется на большое количество лет (10, 15 и более). Составьте программу вычисления суммы, которая будет выплачена банку (срок выберите сами, скажем 15 лет), если в год предполагается платить, кроме части долга, еще и 15% годовых.
5. Длина отрезка задана в дюймах. Один дюйм равен 2,54 см. Перевести значение длины в метрическую систему, т.е. выразить ее в метрах, сантиметрах и миллиметрах. Например, 21 дюйм = 0 м 53 см 3,4 мм.
6. В языке C/C++ стандартная операция «возведение в степень» реализуется с помощью готовой функции. Однако при определенной математической подготовке можно получить требуемый результат, не используя эту функцию. Пусть нам необходимо возвести число a в степень x : $y=a^x$. Возьмем натуральный логорифм от обеих частей равенства:
7. Составьте программу, которая выводит на экран день недели по дате. Воспользуйтесь следующим алгоритмом вычисления дня недели:

день недели = остаток от деления X на 7,

где $X = \text{abs}(\text{trunc}(2.6*m-0.2)+d+y/4+y+c/4-2*c)$;

m - номер месяца (см. ниже);

d - число (день месяца);

c - номер столетия (см. ниже);

y - номер года в столетии.

Формула верна для григорианского календаря нового стиля (от 1582 до 4903 года). Во-вторых, год и месяц следует предварительно преобразовать так, как если бы начало года приходилось на 1 марта. Иными словами, март в этой формуле имеет порядковый номер 1, апрель 2, ..., январь 11 и февраль 12, причем январь и февраль следует отнести к предыдущему году. Например, для 1 февраля 1991 года номер месяца должен быть равен 12, а год 1990, в то время как для 31 декабря 1991 года номер месяца - 10, а год - 1991. Результат вычисления дается в виде целого числа в диапазоне от 0 до 6, причем 0 соответствует воскресенью.

 Пригласите преподавателя и продемонстрируйте ему все выполненные Вами задания. После этого сохраните все программы и выйдите из среды Dev-C++.

 Будьте готовы ответить на его вопросы. Если вы обучаетесь в условиях использования дистанционных образовательных технологий, просто вышлите все файлы с подготовленными Вами программами тьютору (если это предусмотрено технологической картой).

Лабораторная работа 5.

ОПЕРАТОР ВЕТВЛЕНИЯ

1. Что такое ветвление и как оно организуется в языке C/C++

При разработке алгоритмов очень часто возникают ситуации, когда выполняемые действия зависят от некоторого условия, т.е. сначала требуется проверить условие (или несколько условий) и в зависимости от того, выполняется оно или нет, выполнить различные действия. Например, при начислении стипендии проверяется, как студент сдал сессию. Если у него остались «хвосты», то ему будет назначена нулевая стипендия, если он сдал все на «отлично», то – повышенная, в остальных случаях – обычная.

Для организации таких действий в языке C имеется оператор ветвления (условный оператор, оператор IF).

2. Условный оператор IF

Задание 22

Запустите среду Dev-C++.

В папке Ex_C++ откройте электронную книгу Ex_C++\Пособие\Введение в программирование.pdf и прочтите раздел 3 «Выбор вариантов» (с. 14-17).

Задание 23

Запустите среду Dev-C++. Загрузите файл **ex_if_1.cpp**. Проанализируйте текст программы. Запустите программу на выполнение и проверьте ее работу для различных данных. Для каких данных программа выдает некорректный результат? Модифицируйте программу таким образом, чтобы она в случае ввода одинаковых чисел выдавала сообщение: «Числа равны!»

Покажите результат преподавателю.

Задание 24

Психологи рекомендуют учитывать оптимальное соотношение возрастов при вступлении в брак. Оно таково: возраст невесты равен половине возраста жениха плюс 7. Составьте программу для определения, подходят ли жених и невеста друг другу по возрасту.

Задание 25

1. Составьте программу, которая позволяет найти максимальное из трех заданных чисел.
2. Протестируйте эту программу (проверьте корректность ее работы в разных случаях). Сколько наборов входных данных нужно проверить? Запишите их в тетрадь.
3. Продемонстрируйте программу преподавателю.

Задание 26

1. Загрузите файл **ex_if_2.cpp**.
2. Прочитайте и проанализируйте текст программы. Обратите внимание на составное условие.
3. Модифицируйте эту программу так, чтобы в том случае, когда введены значения сторон (такие, что из них можно построить треугольник, помимо выдачи текста «можно»), еще подсчитывался периметр треугольника.

Контрольное задание

Вариант 1

Составьте программу, которая умеет решать квадратные уравнения (по заданным коэффициентам уравнения A , B , C находит корни уравнения. Рассматриваются различные случаи: нет корней, есть один корень, два различных корня).

Вариант 2

По заданной стороне квадрата и радиусу круга выяснить, поместится ли:

- а) круг в квадрат;
- б) квадрат в круг?

Результаты вывести на экран.

Вариант 3

Составьте программу, которая умеет решать линейные уравнения (уравнения вида $Ax+B=0$). Рассматриваются различные случаи: нет решений, решением является любое x , единственное решение (вычисляет его).

Задачи для самостоятельной работы

8. Напишите программу, которая определяет, попадает ли точка с координатами (x,y) в круг радиусом R и центром в начале координат.
9. Напишите программу, которая определяет, в какой координатной четверти (квadrante) находится точка с координатами (x,y) . Вывести на экран номер квадранта. Учесть случай, когда точка лежит на одной из координатных осей или в начале координат.
10. Напишите программу, которая определяет, принадлежит ли точка с координатами (x,y) прямой $y=1-x$. Если это имеет место, то программа должна вывести на печать координаты точки, а в противном случае – сообщение «Точка не принадлежит прямой».
11. Известны координаты трех точек. Напишите программу, которая определяет, находятся ли точки на одной прямой.
12. Напишите программу, которая определяет по введенным значениям длин сторон квадрата и радиуса круга верно ли утверждение «Круг вписан в квадрат».
13. С клавиатуры вводится трехзначное число. Напишите программу, которая проверяет, равна ли сумма его цифр значению 9.

14. Дано трехзначное число. Определить, равен ли квадрат этого числа кубов его цифр.
15. Даны два числа: a и b . Составить программу, которая определяет, является ли первое число делителем второго. А наоборот?
16. Дано трехзначное число. Составьте программу, которая определяет, есть ли среди его цифр одинаковые.
17. Год является високосным, если его номер кратен 4, однако из кратных 100 високосными являются лишь кратные 400 (например, 1700, 1800 и 1900 – невисокосные, а 2000 – високосный). Составьте программу, которая вводит число и определяет, является ли год с таким номером високосным.
18. Составьте программу, которые вводит три числа и определяет, является ли треугольник со сторонами данной длины прямоугольным.

Пригласите преподавателя и продемонстрируйте ему все выполненные Вами задания. После этого сохраните все программы и выйдите из среды Dev-C++. Будьте готовы ответить на его вопросы.

Если вы обучаетесь в условиях использования дистанционных образовательных технологий, просто вышлите все файлы с подготовленными Вами программами преподавателю (если это предусмотрено технологической картой).

Лабораторная работа 6.

ОПЕРАТОР SWITCH

1. Для чего нужен оператор switch

В ряде задач возникают ситуации, при которых нужно использовать оператор ветвления, поскольку в зависимости от некоего условия нужно выполнять то или иное действие, но при этом число условий достаточно большое. Например, если необходимо определить название для недели по его номеру, то решение задачи с помощью оператора ветвления будет выглядеть не очень красиво:

```
...
int n;
printf('Номер дня недели = ');
scanf("%d", n);
if (n==1) printf("понедельник");
    if (n==2) printf("вторник"); else
        if (n==3) printf("среда"); else
            if (n==4) printf("четверг"); else
                if (n==5) printf("пятница"); else
                    if (n==6) printf("суббота"); else
                        if (n==7) printf("воскресенье");
                            else printf("неверный номер");
...

```

Можно, конечно, решать подобные задачи рассмотренным способом. Однако для описания подобных действий в языке C/C++ имеется оператор выбора (оператор switch).

Задание 27

В электронном пособии (Ex_C++\Пособие\Введение в программирование.pdf) прочитайте раздел «Переключатель switch (множественный выбор)» (с. 17-18).

2. Переключатель switch

Задание 28

1. Запустите среду Dev-C++.
2. Загрузите файл **ex_switch_1.cpp**. Прочитайте текст программы и проанализируйте его. Запустите программу на выполнение и посмотрите, как она работает.
3. Модифицируйте данную программу таким образом, чтобы она при вводе числа 0 выводила сообщение «Понедельник начинается в субботу?», а при вводе числа 8 – «Вторник вам не пятница!».

Задание 29

Составьте программу, которая позволяет для любых двух заданных чисел А и В и операции («+» – сложение, «-» – вычитание, «*» – умножение, «/» – деление) получать на экране результат.

Подсказка

В данной задаче для хранения знака операции лучше всего использовать переменную типа **char**.

Задание 30

Составьте программу, которая позволяет определить, является введенная буква гласной или согласной.

Задание 31*

Известно, что астрологи делят год на 12 периодов и ставят каждому из них в соответствие знак Зодиака:

20.1 - 18.2	⇒ Водолей	23.7 - 22.8	⇒ Лев
19.2 - 20.3	⇒ Рыбы	23.8 - 22.9	⇒ Дева
21.3 - 19.4	⇒ Овен	23.9 - 22.10	⇒ Весы
20.4 - 20.5	⇒ Телец	23.10 - 22.11	⇒ Скорпион
21.5 - 21.6	⇒ Близнецы	23.11 - 21.12	⇒ Стрелец
22.6 - 22.7	⇒ Рак	22.12 - 19.1	⇒ Козерог

Составьте программу определения знака Зодиака для любого человека.

Подсказка

Идеальное решение для данной задачи – преобразовать дату в одно число: месяц умножить на 100 и прибавить день. Тогда, например, для Водолеев можно указать точный диапазон: 120..218.

Задачи для самостоятельной работы

1. Составить программу, которая вводит номер месяца и выводит название времени года, к которому относится этот месяц.
2. Составить программу, которая вводит номер года и номер месяца, а выводит количество дней в этом месяце. Предусмотреть ситуации, когда год високосный, когда – нет (пояснения см. в задании 10 для самостоятельной работы предыдущей лабораторной работы).
3. Игральный картам условно присвоены следующие порядковые номера в зависимости от их достоинства: «валету» – 11, «даме» – 12, «король» – 13, «тузу» – 14. Порядковые номера остальных карт соответствуют их названию («шестерка», «девятка» и т.п.). Составьте программу, с помощью которой можно будет определить карту по ее номеру.
4. Составьте программу, которая вводит числовое значение оценки (1-5) и выводит ее вузовское название (1 – «плохо», 2 – «неудовлетворительно», 3 – «удовлетворительно», 4 – «хорошо», 5 – «отлично»).
5. Дано целое число в диапазоне от 10 до 40, определяющее количество заданий теста. Вывести текстовое описание заданного количества заданий, обеспечив согласование числа со словами «учебное задание», например: 18 – «восемнадцать учебных заданий», 21 – «двадцать одно учебное задание».

6. Дано целое число в диапазоне от 100 до 999. Составьте программу, которая выводит текстовое описание данного числа, например: 145 – «сто сорок пять», 768 – «семьсот шестьдесят восемь».

 *Пригласите преподавателя и продемонстрируйте ему все выполненные Вами задания. После этого сохраните все программы и выйдите из среды Dev-C++. Будьте готовы ответить на вопросы преподавателя.*

 *Если вы обучаетесь в условиях использования дистанционных образовательных технологий, просто вышлите все файлы с подготовленными Вами программами преподавателю.*

Лабораторная работа 7

ЦИКЛ С ПАРАМЕТРОМ

Задание 32

В электронном пособии (Ex_C++\Пособие\Введение в программирование.pdf) прочитайте раздел «Циклы» (с. 19-20).

Запустите среду Dev-C++ и проверьте работу всех типовых алгоритмов, приведенных в этом разделе.

Задание 33

Считайте с диска файл с именем **ex-for_1.cpp**. Содержащаяся в нем программа 15 раз печатает на экране слово «Халва...» (убедитесь в этом, запустив программу). Модифицируйте программу так, чтобы:

- слово печаталось не 15, а 10 раз;
- слова печатались в одну строку;
- перед первым словом печаталось слово “Начало”, а после последнего - слово “Конец”;
- каждое слово печаталось с новой строки, и между ними была пустая строка;
- перед каждым словом «Халва...» печатался его порядковый номер.

Задание 34

В файле **ex_for_2.cpp** находится программа печати таблицы значений функции $f(x) = \sin(x)$ при $x = 0, 0.1, 0.2, \dots, 2$. Модифицируйте программу так, чтобы:

- печатался заголовок;
- печатались еще и значения аргумента x ;
- печатались значения при $x = 0, 0.2, 0.4, \dots, 3$;
- то же самое печаталось в обратном порядке.

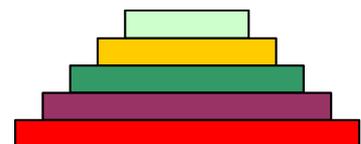
Задание 35

В файле **ex_for_3.cpp** находится программа движения окружности по горизонтали. Используя эту программу, смоделируйте на экране равномерное прямолинейное движение этой окружности:

- по вертикали;
- по диагонали окна.

Задание 36*

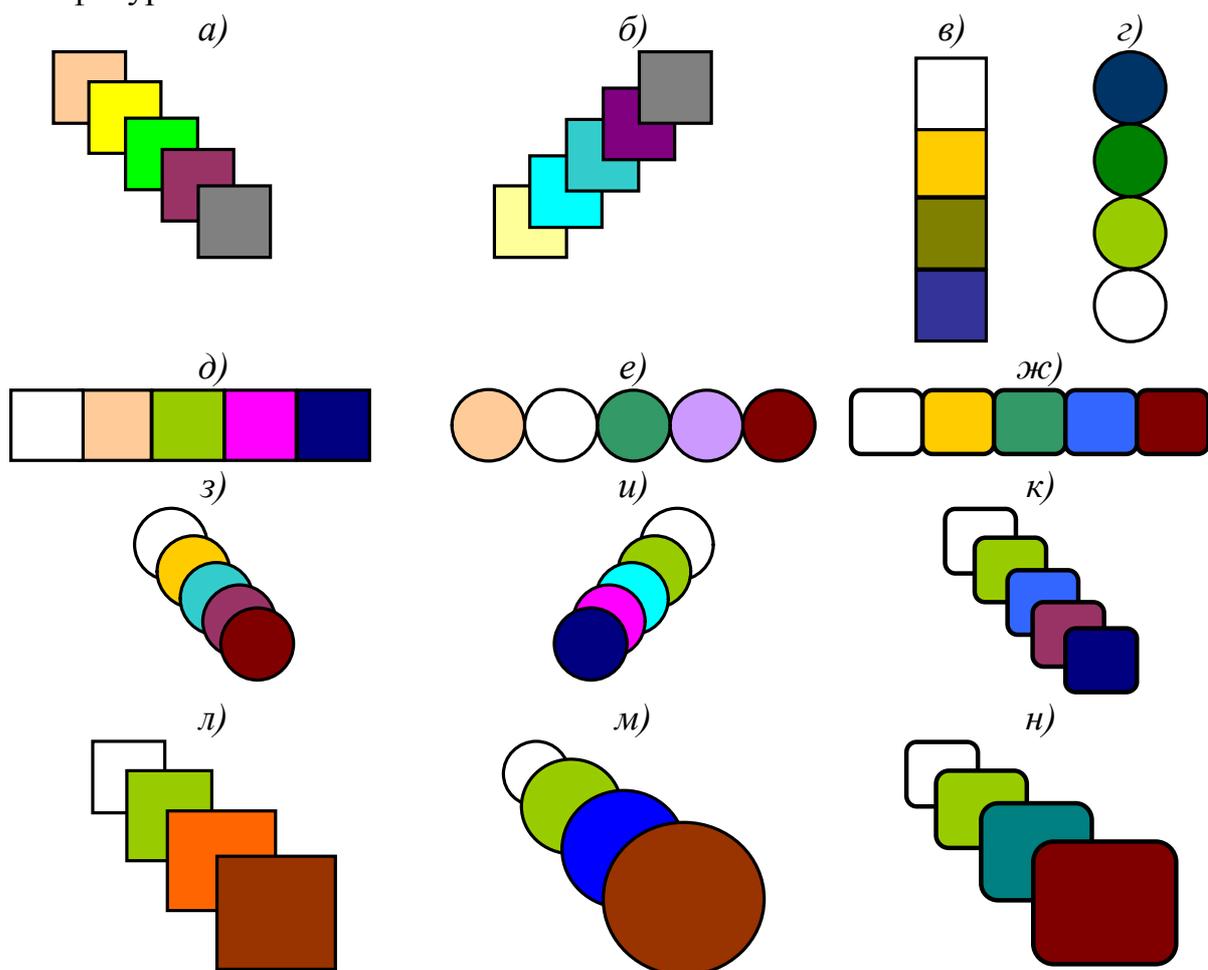
Используя конструкцию цикла со счетчиком, самостоятельно составьте программу рисования детской игрушечной пирамидки (вид сбоку).



Задачи для самостоятельной работы

- Построить окружность, состоящую из 6 секторов, раскрашенных в разные цвета (для задания «разных цветов» можно использовать цвет **clRandom**).
- Составить программу рисования мишени, состоящей из окружностей разного цвета.

2. Составить программу рисования следующей последовательности однотипных фигур:



☞ *Пригласите преподавателя и продемонстрируйте ему все выполненные Вами задания. После этого сохраните все программы и выйдите из среды Dev-C++. Будьте готовы ответить на вопросы преподавателя. Если вы обучаетесь в условиях использования дистанционных образовательных технологий, вышлите все файлы с подготовленными Вами программами преподавателю.*

Лабораторная работа 8*.

КАК НАРИСОВАТЬ ЗАБОР

Конечно, простейший забор можно нарисовать и с помощью прямоугольников. Но это не так интересно. Давайте попробуем нарисовать забор традиционный, из заостренных планочек. Как это сделать?

Познакомимся еще с несколькими процедурами библиотеки <graphics.h>. Для этого сначала поговорим о графическом курсоре.

В консольных приложениях курсор виден в виде мигающей черточки. В графическом же режиме курсора нет. Дело в том, что он указывает на конкретную маленькую точку, и если его нарисовать, то самой точки видно не будет. Поэтому введено понятие «графического пера», под которым понимают некоторое невидимое перо, с помощью которого можно рисовать в графическом окне.

Кроме того, введено понятие «кисти», т.е. некая невидимая кисть, которая позволяет закрашивать заданные области. Технология достаточно простая: если сначала задать цвет кисти (вы уже ею пользовались), а затем нарисовать, например, прямоугольную рамку с помощью процедуры **Rectangle**, то она будет закрашена внутри цветом кисти.

Процедура установки графического пера:

moveto(int x, y); – передвигает перо к точке с координатами (x, y);

Процедуры рисования линий:

lineto(int x, y); – процедура позволяет проводить прямую из точки с текущими координатами в точку (x,y) текущим цветом пера;

line(int x1,y1,x2,y2); – процедура обеспечивает рисование отрезка из точки (x1,y1) в точку (x2,y2) текущим цветом пера.

Пример:

```
...
moveto(10,10);
lineto(100,100); { проводит прямую в точку 110,110 }
...
```

Процедуры задания цвета и стиля линий

setcolor(int color); – процедура позволяет задать цвет линии. Перечень допустимых цветов:

0	BLACK	черный	8	DARKGRAY	темно-серый
1	BLUE	синий	9	LIGHTBLUE	светло-синий
2	GREEN	зеленый	10	LIGHTGREEN	светло-зеленый
3	CYAN	морской волны	11	LIGHTCYAN	светлый морской волны
4	RED	красный	12	LIGHTRED	светло-красный
5	MAGENTA	фиолетовый	13	LIGHTMAGENTA	светло-фиолетовый
6	BROWN	коричневый	14	YELLOW	желтый
7	LIGHTGRAY	светло-серый	15	WHITE	белый

setlinestyle(int ps); – устанавливает стиль линий.

Стиль линии (ps) определяется одной из предопределенных констант. Идентификаторы этих констант и определяемые ими стили см. в электронной книге Введение в программирование.pdf (с. 31-34).

setfillcolor(int color); – устанавливает цвет закрашки, задаваемый параметром **color**.

Пример:

```
#include <graphics.h>
main()
{
  int i,x,y;
  initwindow(800,600); // открыть окно для графики размером 800x600 точек
  setcolor(WHITE); // белый цвет линий
  setlinestyle(0,0,3); // линия утроенной толщины
  bar(100,50,400,100); // рисуем белый закрашенный прямоугольник
  setfillstyle(1,BLUE); // устанавливаем голубой цвет закрашки
  setcolor(BLUE); // голубой цвет линии
  bar(100,100,400,150); // рисуем голубой прямоугольник
  setfillstyle(1,RED);
  setcolor(RED);
  bar(100,150,400,200);
  getch(); // остановка программы до нажатия любой клавиши
  closegraph(); // закрытие окно графики
}
```

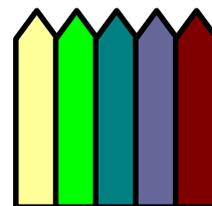
Задание 37

Запустите среду Dev-C++.

Теперь давайте рисовать забор, состоящий из пятиугольных досок (см. рисунок). При рисовании забора не забудьте раскрасить каждую доску в свой цвет.

При рисовании линий используйте **lineto**.

Успехов вам.



Задание 38

Вы нарисовали некий забор. Модифицируйте программу таким образом, чтобы был нарисован забор длиной:

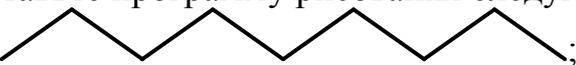
- в 500 точек (пикселей) из 10 досок;
- в 400 пикселей из заданного числа досок (количество досок вводится с клавиатуры);
- в N пиксель и из M досок.

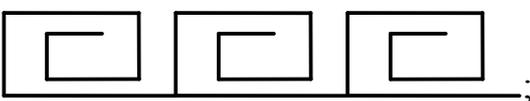
Задание 39

Еще одна модификация: измените программу так, чтобы был нарисован забор длиной в N пиксель, причем длина одной доски равнялась K пиксель.

Задачи для самостоятельной работы

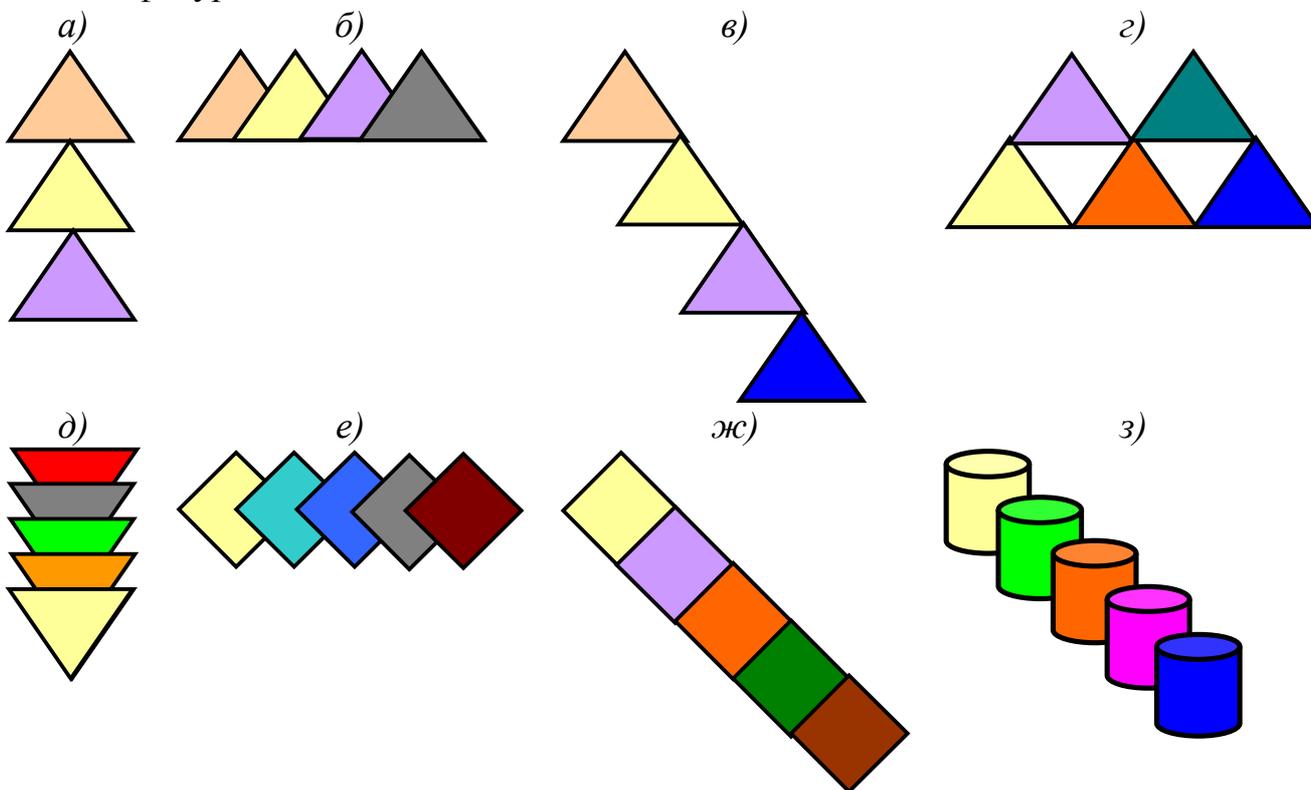
3. Составьте программу рисования следующего орнамента:

а) ;

б) ;



4. Составьте программу (с использованием цикла) рисования правильного шестиугольника.
5. Составьте программу рисования пятиугольной звездочки.
6. Составить программу рисования следующей последовательности однотипных фигур:



☞ Пригласите преподавателя и продемонстрируйте ему все выполненные Вами задания. После этого сохраните все программы и выйдите из среды Dev-C++. Будьте готовы ответить на вопросы преподавателя.

☞ Если вы обучаетесь в условиях использования дистанционных образовательных технологий, вышлите все файлы с подготовленными Вами программами преподавателю.

Лабораторная работа 9⁰

ЗВЕЗДНОЕ НЕБО И ПРОЧИЕ СТРАННОСТИ

1. Получение случайного числа

При создании некоторых программ (например, игр) требуется получение случайного числа при каждом запуске программы (например, если Вы моделируете на компьютере игру в кости, в карты и т.п.).

В языке Си существуют следующие функции для работы со случайными числами (их описание находится в библиотеке **stdlib.h**):

n = rand(); – получить случайное целое число в интервале от 0 до **RAND_MAX** (это очень большое целое число — 32767)

srand (m); – установить начальное значение случайной последовательности, равное **m**.

Задание 40

Запустите среду **Dev-C++**.

Считайте программу из файла **rnd01.cpp** и разберитесь, как она работает и что выдает на экран дисплея. Для этого запустите эту программу 2-3 раза и определите, какую последовательность чисел она выдает: одну и ту же при каждом запуске или различные.

Задание 41

Считайте программу из файла **rnd02.cpp** и разберитесь, как она работает и что выдает на экран дисплея. Для этого запустите эту программу 2-3 раза и определите, какую последовательность чисел она выдает: одну и ту же при каждом запуске или различные. Определите, в чем отличие этих программ.

Задание 42

Считайте программу из файла **rnd03. cpp** и разберитесь, как она работает.

Считайте программу из файла **rnd04. cpp** и попытайтесь разобраться, как она работает. Выясните, как там организован цикл и когда программа завершает свою работу.

Задание 43

Используя алгоритм, предложенный в программе **rnd04. cpp**, составить программу, которая выводит на экран дисплея закрашенные прямоугольники в произвольном месте экрана, произвольных размеров и произвольного цвета. Программа должна работать бесконечно (т.е. до тех пор, пока не будет закрыто окно). Очищать окно после каждого вывода не нужно!

2. Рисование точек в графическом режиме

Точки в языке C/C++ рисуются достаточно просто: в графическом режиме достаточно воспользоваться функцией

putpixel(int x, y,color);

Эта процедура позволяет закрасивает на **экране** точку с координатами x и y цветом **Color** (**int** – это описание типа параметров функции).

Например, если Вам хочется нарисовать точку *голубого* цвета с координатами (100,200), то потребуется лишь в программе обратиться к процедуре вывода точки следующего вида:

putpixel(100,200,BLUE);

Задание 44

Составьте программу «Звездное небо», которая постоянно выводит точки разного цвета в разных местах окна. Программа должна работать бесконечно (т.е. до тех пор, пока не будет закрыто окно).

Задачи для самостоятельной работы

Учебные тренажеры работают следующим образом:

- выдается некоторое случайным образом подобранное задание;
- пользователь вводит ответ;
- если ответ верный, обучаемому выдается следующее задание;
- если ответ неверный, обучаемому выдается верный ответ и сообщение о количестве допущенных им ошибок.

Для создания заданий используйте датчик случайных чисел.

1. Разработайте простейший тренажер для учащихся младших классов, который проверяет:
 - а) знание таблицы умножения (т.е. умение перемножать числа от 1 до 10);
 - б) умение складывать числа от 1 до 100;
 - в) навык вычитания чисел от 1 до 100. Следует иметь в виду, что учащиеся начальной школы еще не знают отрицательных чисел, поэтому уменьшаемое должно быть больше вычитаемого;
 - г) умение делить целые числа до 100. Здесь должны проверяться действия, обратные умножению целых чисел: т.е. деление должно быть целочисленное (т.е. делимое должно делиться делителем нацело!).
2. Разработайте простейший тренажер, который проверяет умение решать:
 - а) линейные уравнения вида $ax=b$;
 - б) квадратные уравнения вида $ax^2+bx+c=0$.

☞ *Пригласите преподавателя и продемонстрируйте ему все выполненные Вами задания. После этого сохраните все программы в своей папке и выйдите из среды Dev-C++. Будьте готовы ответить на вопросы преподавателя.*

☞ *Если вы обучаетесь в условиях использования дистанционных образовательных технологий, вышлите все файлы с подготовленными Вами программами преподавателю.*

Лабораторная работа 10

ЦИКЛ С ПАРАМЕТРОМ: ТИПОВЫЕ АЛГОРИТМЫ

1. Типовые алгоритмы

2.1. Вычисление суммы

Задание 45

Запустите среду **Dev-C++**. Выполните следующие действия:

- а) загрузите файл **ex_sum1.cpp**;
- б) прочитайте текст программы и проанализируйте его;
- в) запустите программу на выполнение (рекомендуем задать в качестве значения числа N значения 1, 10, 0, -5); оцените, как программа ведет себя в различных ситуациях);
- г) измените программу так, чтобы она считала сумму кубов первых N чисел.

Задание 46

Выполните следующие действия:

- а) загрузите файл **ex_pifagor.cpp**;
- б) прочитайте текст программы и проанализируйте его;
- в) запустите программу на выполнение;
- г) модифицируйте программу так, чтобы она выводила таблицу Пифагора 15 на 15.

2.2. Вычисление факториала

Задание 47

Внимательно просмотрите перечисленные ниже программы, поскольку на их основе потребуется самостоятельно реализовывать аналогичные алгоритмы.

- а) **ex_factorial_1.cpp**: если значение факториала записывать в переменную типа `int`, то быстро наступает переполнение. Выясните, при каком значении введенного числа это переполнение наступает (т.е. в качестве ответа выводится отрицательное число или 0);
- б) **ex_factorial_2.cpp**: та же программа вычисления значения факториала, но в ней целый тип `int` заменен `unsigned __int64` (т.е. беззнаковое 64-разрядное целое). Выясните, при каком значении введенного числа наступает переполнение с этим типом данных;
- в) **ex_factorial_3.cpp**: предлагается еще один вариант решения задачи: результат записывать в переменную типа `float`. Выясните, при каком значении введенного числа переполнение наступает в этом случае.

2.3. Вычисление суммы ряда

Задание 48

Квадрат любого натурального числа N равен сумме первых N нечетных чисел:

$$1^2 = 1$$

$$2^2 = 1 + 3$$

$$3^2 = 1 + 3 + 5$$

$$4^2 = 1 + 3 + 5 + 7.$$

Проверьте, верна ли эта закономерность для других натуральных чисел.

Задание 49

Многие из математических величин или значений функций могут быть выражены как суммы бесконечных последовательностей. Чем больше членов ряда участвует в сложении, тем более точным получается искомое значение.

Составьте программы и попытайтесь определить минимальное число членов ряда, обеспечивающих вычисление следующих данных:

а) $e = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!} + \dots + \frac{1}{n!} + \dots$; (значение $e = 2,718\ 281$);

б) $\pi = 4\left(1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots + \frac{(-1)^{n-1}}{2n-1} + \dots\right)$ (значение $\pi = 3,141\ 592$);

в) $\sqrt{2} = 2 \cdot \left(1 - \frac{1}{3}\right) \cdot \left(1 + \frac{1}{5}\right) \cdot \left(1 - \frac{1}{7}\right) \cdot \dots \cdot \left(1 - \frac{(-1)^k}{2k+1}\right)$ (значение $\sqrt{2} \approx 1,414\ 213$)

2.4. Определение среднего

Задание 50

Для определения рентабельности авиалинии вычисляется среднее количество пассажиров за год. Составьте программу вычисления среднего количества пассажиров, перевезенных за год, если известно общее количество пассажиров, перевезенных за каждый месяц.

Задание 51

Составить программу определения среднемесячной температуры воздуха, если известна средняя температура за каждый день месяца. Исходные данные для вычислений лучше использовать реальные.

2.4. Определение экстремума

Задание 52

В память компьютера вводят по очереди координаты N точек. Определить координаты точки, максимально удаленной от начала координат

Задание 53

У тренера футбольного клуба имеется список членов команды и количества очков, которые принес команде каждый игрок в данном сезоне. Составить программу, с помощью которой можно определить самого результативного игрока.

2.5. Рекуррентные соотношения

В 1202 году итальянский математик Леонард Пизанский, известный под именем Фибоначчи, предложил такую задачу:

Пара кроликов каждый месяц дает приплод – пару кроликов (самца и самку), от которых через два месяца уже получается новый приплод. Сколько кроликов будет через год, если в начале года мы имели одну пару молодых кроликов?

Обратим внимание на то, что числа, соответствующие количеству пар кроликов, которые имеются через каждый месяц, составляют последовательность:

1, 1, 2, 3, 5, 8, 13, 21, 34, ...

Каждый из членов этой последовательности, начиная с третьего, равен сумме двух предыдущих членов. Эта последовательность получила название ряда Фибоначчи, а ее члены - *числами Фибоначчи*.

Обозначив n -й член ряда Фибоначчи $F(n)$, мы получим следующую зависимость, называемую *рекуррентной* (т.е. *зависимость, в которой последующие элементы определяются предыдущими*):

$$F(1)=1$$

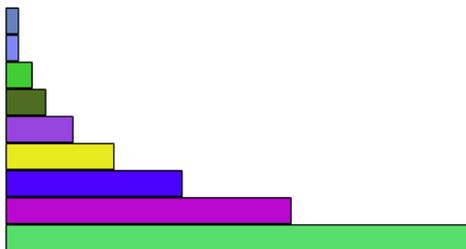
$$F(2)=1$$

$$F(n)=F(n-1)+F(n-2), \text{ при } n \geq 3$$

Задание 54

Составить программу, которая:

- выводит первые N чисел Фибоначчи;
- определяет, сколько кроликов будет через M месяцев; через G лет;
- позволит определить номер члена последовательности, значение которого превосходит заданное число K ;
- представить ряд Фибоначчи графически. Например:



(длина прямоугольника пропорциональна числу; например, его длина равна числу, умноженному на 10).

Задачи для самостоятельной работы

1. Напишите программу вычисления наибольшего общего делителя двух натуральных чисел.
 2. Оборудование фирмы в результате износа и старения уценяется на $p\%$ ежегодно. Составьте программу, которая по первоначальной стоимости оборудования и времени его эксплуатации вычисляет текущую стоимость этого оборудования.
 3. Напишите программу, которая проверяет, является ли простым введенное пользователем положительное целое число. Простым называется такое число, которое делится без остатка только на единицу и само себя (например, 5, 11, 19).
 4. Напишите программу вычисления совершенных чисел, не превосходящих заданного числа N . Совершенным называется такое число, сумма делителей которого совпадает с самим числом (например, $6=1+2+3$).
 5. Напишите программу, которая проверяет, является ли простым введенное пользователем положительное целое число. Простым называется такое число, которое делится без остатка только на единицу и само себя (например, 5, 11, 19).
 6. Для каждого посетителя парикмахерской (с одним мастером) известны следующие величины: t – момент его прихода, τ – продолжительность его обслуживания. Сколько клиентов обслужит мастер за смену продолжительностью T ? Сколько рабочего времени он потратит на обслуживание?
 7. **Расписание звонков.** В учебном заведении задается начало учебного дня, продолжительность «пары» или урока, продолжительность обычного и большого перерывов (и их «место» в расписании), количество пар (уроково). Составить программу, которая выведет на экран расписание звонков на весь учебный день.
 8. **Гуси и кролики.** У гусей и кроликов $2n$ лап. Сколько может быть гусей и кроликов (вывести все возможные сочетания)?
 9. Составить программу, которая позволит определить сколько можно купить быков, коров и телят, платя за быка 10 руб., за корову — 5 руб., а за теленка – 0,5 руб., если на 100 руб. надо купить 100 голов скота?
 10. Составьте программу, которая выводит на экран календарь на любой месяц по заданному году. Воспользуйтесь алгоритмом вычисления дня недели, описанном в лабораторной работе 5.
- ☞ *Пригласите преподавателя и продемонстрируйте ему все выполненные Вами задания. После этого сохраните все программы в своей папке и выйдите из среды Dev-C++. Будьте готовы ответить на вопросы преподавателя.*
- ☞ *Если вы обучаетесь в условиях использования дистанционных образовательных технологий, вышлите все файлы с подготовленными Вами программами преподавателю.*

Лабораторная работа 11

ПРОЕКТИРОВАНИЕ ПРОГРАММ И ПРОЦЕДУРЫ

1. Зачем нужна технология программирования

Предположим, Вам захотелось построить дом. Возможны два варианта:

- «любительский»: берем кирпич (дерево, плиту, бетон...) и строим. Что-нибудь получится. Если не развалится. А если развалится – будем пробовать еще. Пока не получится;

- «профессиональный»: создаем сами или заказываем ЭСКИЗ дома (фасад, количество комнат, подземных гаражей, туалетов, размеры и количество окон и дверей и т.д.), затем – ПРОЕКТ (подробнейший план дома с размерами деталей и структурой материала), затем – привязка дома к ландшафту (чтобы было удобно строить и красиво смотрелось) и только затем уж начинается практическая РЕАЛИЗАЦИЯ: фундамент, стены, перекрытия, крыша, системы обеспечения (отопление, водопровод, канализация, электричество, телефон), внутренняя отделка, строительство газонов, установка мебели.

Цепочка ЭСКИЗ → ПРОЕКТ → РЕАЛИЗАЦИЯ имеет отношение практически ко всем видам производственной деятельности: создание машин, картин, одежды, и в том числе программ для компьютеров. Когда речь идет о программе в 5-50 строк, можно воспользоваться «любительским» вариантом – сел и пиши. Если же речь идет о БЫСТРОМ написании БОЛЬШИХ программ, необходима специальная технология, обеспечивающая как скорость, так и качество программного продукта. Одной из таких технологий является технология программирования «сверху вниз». С примером разработки программы по такой технологии Вы и познакомитесь в данной лабораторной работе.

2. Знакомство с технологией проектирования «сверху вниз»

Выполним следующее задание.

Составить программу, которая рисует на экране дисплея еловый лес.

РЕШЕНИЕ

Часть 1. Уточнение постановки задачи (ЭСКИЗ)

Выясним подробнее, что же все-таки нам нужно нарисовать. Договоримся, что под «еловым лесом» будем понимать набор елочек различных размеров и различной расцветки, расположенных в произвольных местах экрана.

Часть 2. Первый вариант решения

Предположим, что у нас есть уже готовая процедура $Forest(n)$ (n – количество деревьев в лесу), при вызове которой рисуется требуемый лес. Тогда наша программа выглядит достаточно просто:

```

#include <conio.h>
#include <graphics.h>
main()
{
    Forest(25); // Вызов процедуры – рисуем лес из 25 деревьев
}

```

К сожалению, если эту программу запустить в Dev-C++, будет выдана ошибка, означающая, что среда не знает процедуры **Forest**.

Часть 3. Уточнение решения

Следовательно, такую процедуру надо создать. Сначала познакомимся с тем, как это делается.

ТЕОРИЯ

Вы уже знакомы со стандартными процедурами типа **rectangle(x,y,x1,y1)**, **setcolor(m)** и т.п. Язык программирования C/C++ позволяет вам создавать свои собственные процедуры и функции.

 **Процедура** представляет собой часть программы, оформленной специальным образом.

Любая процедура имеет структуру, напоминающую структуру программы:

```

void <имя >[(<список формальных параметров>);]
{
    <операторы>;
}

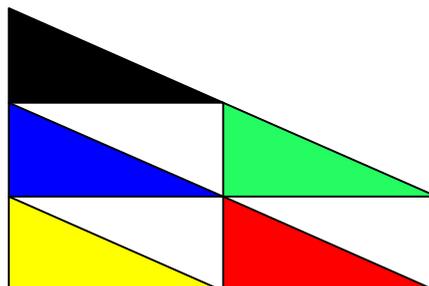
```

Обратите внимание на структуру заголовка процедуры. Слово **void** означает, что она предназначена не для вычислений, а для выполнения некоторых действий.

Процедуры должны быть описаны перед программой, в которой они используются.

Задание 55

1. Запустите среду Dev-C++. Загрузите программу **Proc1.cpp**. Разберитесь в тексте программы. Обратите внимание на оформление заголовка и расположение процедуры в тексте программы.
2. Запустите программу на выполнение и выясните, что она делает.
3. Модифицируйте программу так, чтобы она рисовала пять треугольников:



ТЕОРИЯ

Заголовок процедуры включает в себя ключевое слово `void` (читается «войд» и переводится как «пустота» или «пустой»), имя процедуры и описание параметров, т.е. тех данных, которые передаются в процедуру извне и используются в теле процедуры. ТИПЫ ПАРАМЕТРОВ ОПИСЫВАЮТСЯ НЕПОСРЕДСТВЕННО В ЗАГОЛОВКЕ.

Различные параметры с описанием типов отделяются друг от друга ЗАПЯТОЙ. В процедуре может и не быть параметров.

Например:

`void house(int x, int y, char name, float usdl);` – процедура с 4 параметрами (2 целых числа, символ и вещественное число).

...

`void man(float name, float adress);` – заголовок процедуры с 2 параметрами (числами).

...

`void girl(char name, int eye, float nice);` – процедура с 3 параметрами (символ, целое число и вещественное число).

...

`void wait;` – заголовок процедуры без параметров.

...

Задание 56

1. Загрузите программу **Proc2.cpp**. Разберитесь в тексте программы. Выясните последовательность описания вызываемой и вызывающей процедур.
2. Переделайте программу таким образом, чтобы сам фрагмент рисования трех заборов был выделен в отдельную процедуру **pict** БЕЗ ПАРАМЕТРОВ. При этом тело программы должно выглядеть следующим образом:

```
...  
{  
     pict; // Обращение к процедуре рисования трех заборов  
}
```

Задание 57

Загрузите программу из файла **Proc3.cpp**. Допишите в ней процедуру **triangle**, рисующую равносторонний закрашенный треугольник.

- ☞ *Сохраните все программы, пригласите преподавателя и продемонстрируйте их ему. В случае, если Вы обучаетесь дистанционно, после выполнения всей лабораторной работы тексты программ отправите тьютору.*

Часть 4. Уточнение решения

Теперь Вы знаете, что в программе рисования леса следует описать процедуру **Forest**.

Давайте предположим, что у нас есть еще одна графическая процедура **Fir** (по-английски «ель»), которая позволяет нарисовать елочку. Пусть данная процедура имеет следующий заголовок:

void fir(int x,int y, int h, int c);

где x,y – координаты елочки, h – высота елочки, c – цвет линий и закраски. Под «координатами елочки» договоримся понимать, например, координаты ее вершины.

Задание 58

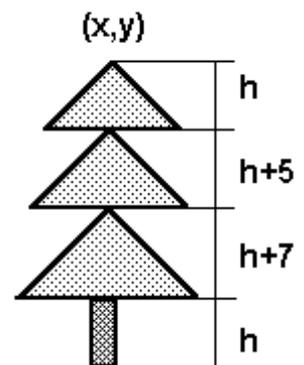
Загрузите файл **Proc4.cpp**. В этом файле находится текст уточненной программы. Запустите программу. Убедитесь в том, что программа не имеет ошибок и выполняется. Но программа ничего не делает, поскольку ее следует еще дописать, вставив вместо комментариев необходимые операторы.

Перенесите в данный файл текст процедуры **triangle**.

Часть 5. Уточнение решения

Теперь давайте попробуем понять, а как, собственно, рисует процедура **Fir**. Для простоты пусть елочка состоит из 3 равнобедренных треугольников и ножки (см. рисунок).

Под «высотой» елочки проще всего считать высоту верхнего треугольника. Цвет елочки – цвет треугольника. Высоту ножки елочки можно задать, например, равной высоте первого треугольника. Треугольники, расположенные ниже, могут иметь высоту немного побольше (например, $h+5$ и $h+7$ соответственно). Вариант размеров елочки Вы можете посмотреть на рисунке. Понятно, что эти размеры условны и, если Вам хочется, Вы можете выбрать их другими.



Таким образом, процедура **Fir** рисует елочку из 3 закрашенных треугольников и ножки.

Задание 59

Учитывая, что процедура рисования треугольника у Вас уже написана (не забудьте ее использовать в проекте), завершите реализацию программы рисования леса. Для ускорения работы воспользуйтесь уже готовой частью программы из файла **Proc4. cpp**.

При рисовании вы, несомненно, заметили дефекты закрашки треугольников. Эту ситуацию можно исправить, если воспользоваться процедурой рисования многоугольников **polygon**.

Задание 60

Откройте файл **ромбы.cpp**. Разберитесь в технологии рисования ромбов.

Модифицируйте программу так, чтобы она рисовала треугольники.

На основе этой программы составьте процедуру рисования треугольника и замените ею процедуру **triangle**. Все проблемы будут устранены.

Задание 61

Модифицируйте программу таким образом, чтобы вместе с лесом рисовались домики (🏠) или машинки (🚗). Количество фигурок должно

соответствовать числу елей. В программе должна быть отдельная процедура рисования выбранного объекта.

ЭТО ВАМ ПРИГОДИТСЯ

Обратите внимание на этапы деятельности при проектировании программы рисования леса – это **ОЧЕНЬ ВАЖНО!** Дело в том, что **ПРАКТИЧЕСКОЕ ПРОГРАММИРОВАНИЕ** – это на 70 % проектирование программы и только на 30 % – запись программы на языке программирования.

Описанный в данной лабораторной работе вариант проектирования разработан и используется в наиболее известных программистских фирмах мира. По этой технологии (напоминаем, что называется она «технология программирования “сверху вниз”»), в частности, разработана и реализована сама среда Dev-C++. Такая технология позволяет достаточно быстро разрабатывать сколь угодно большие программы.

Быстрота разработки обеспечивается тем, что после создания проекта программы (на уровне простых процедур) реализация процедур может быть поручена нескольким программистам и, если в проекте нет серьезных ошибок, программа будет готова тогда, когда последний программист закончит писать свою процедуру. Далее, правда, следует этап отладки программы, но это предмет другой лабораторной работы.

☞ Пригласите преподавателя и продемонстрируйте созданный Вами проект.

☞ При обучении в условиях дистанционного обучения отправьте проект тьютору по электронной почте.

Лабораторная работа 12

ЦИКЛЫ С УСЛОВИЕМ

1. Теория: Цикл с предусловием (WHILE)

Цикл со счетчиком используется тогда, когда количество повторений цикла заранее известно.

Цикл WHILE используется тогда, когда количество повторений заранее неизвестно, но известно *УСЛОВИЕ ЗАВЕРШЕНИЯ* выполнения цикла. При этом условие должно проверяться *ДО ВЫПОЛНЕНИЯ* оператора в цикле.

Общая структура цикла:

```
WHILE <условие>
```

Цикл выполняется до тех пор, пока <условие> *истинно*.

Задача. Ввести целое число и определить, сколько в нем цифр.

Задание 62

Выполните следующие действия:

- а) загрузите файл **ex_12_2.cpp**; там находится программа, решающая приведенную выше задачу;
- б) прочитайте текст программы и проанализируйте его;
- в) запустите программу на выполнение (рекомендуем задать в качестве значения числа М значения 1, 20, 5555, 1234567, -234). Оцените, как программа ведет себя в различных ситуациях;
- г) измените программу так, чтобы она вводила вещественное число и определяла сколько цифр в целой части.

1.3. Цикл с постусловием (do - while)

Этот цикл используется также в ситуации, когда количество повторений неизвестно, а известно УСЛОВИЕ завершения работы цикла. При этом само условие должно проверяться ПОСЛЕ ВЫПОЛНЕНИЯ действий в цикле.

Общая структура цикла:

```
do
{
  <оператор 1>;
  <оператор 2>;
  ...
  <оператор N>
}
while (<условие>;
```

Задача. Ввести натуральное число и найти сумму его цифр. Организовать ввод числа так, чтобы нельзя было ввести отрицательное число или нуль.

Задание 63

Выполните следующие действия:

- а) загрузите файл **ex_12_3.cpp**; там находится решение приведенной выше задачи;
- б) прочитайте текст программы и проанализируйте его;
- в) запустите программу на выполнение;
- г) измените программу так, чтобы она заканчивала работу при вводе числа 999.

Задание 64

Выполните следующие действия:

- а) загрузите файл **ex_гонки.cpp**;
- б) прочитайте текст программы и проанализируйте его;
- в) запустите программу на выполнение;
- д) измените программу так, чтобы в гонке участвовал еще третий участник.

Задание 65

Составьте программу, которая вычисляет сумму ряда:

а) $\pi = 4\left(1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \dots + \frac{(-1)^{n-1}}{2n-1} + \dots\right)$ (значение $\pi \approx 3,141\ 592$);

б) $e = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!} + \dots + \frac{1}{n!} + \dots$; (значение $e \approx 2,718\ 281$).

Добавлять члены последовательности необходимо до тех пор, пока очередной член не станет меньше заданной точности ε .

Задачи для самостоятельной работы

1. В компьютер автомастерской после обслуживания клиента вводится стоимость ремонта. В конце дня вместо стоимости вводится число 0, после чего компьютер выводит на экран дисплея общую сумму выручки за день. Составьте такую программу.
2. Напишите программу, которая выводит двоичное представление введенного с клавиатуры десятичного числа.
3. Напишите программу, которая выводит шестнадцатеричное представление введенного с клавиатуры двоичного числа.
4. Напишите программу, которая выводит десятичное представление введенного с клавиатуры двоичного числа.
5. Напишите программу рисования шахматной доски.
6. Напишите программу-игру «Угадай число». Один игрок, в роли которого выступает компьютер, «задумывает» число в заданных пределах (например, от 1 до 999), другой игрок – человек – пытается его отгадать за минимальное число попыток. Для каждого числа, введенного человеком в качестве очередной версии, компьютер выдает сообщение «мое число

больше», «мое число меньше» или «поздравляю, вы выиграли!», если число угадано точно. Предусмотрите начисление очков обратно пропорционально количеству затраченных попыток.

7. Дано натуральное число. Определить сумму его цифр.
8. В одной старинной задаче мудрец предложил шаху в качестве награды за услугу выдать ему зерно. Причем количество зерна оговорил так: на первую клетку шахматной доски положить 1 зерно, на 2-ую - 2, на 3-ю - 4, на 4-ую - 8 и т.д. Составить программу, с помощью которой можно вычислить, на какой клетке суммарное количество зерен превысит N штук.
9. В компьютер по очереди вводятся баллы, полученные участницей на конкурсе красоты «Мисс Урал», а также номер участницы. Составить программу, которая обеспечит этот ввод и вывод на экран номер участницы, имеющей лучший результат.

☞ *Пригласите преподавателя и продемонстрируйте ему все выполненные Вами задачи и задания. После этого сохраните все программы в своей папке и выйдите из среды Dev-C++. Будьте готовы ответить на вопросы преподавателя.*

☞ *Если вы обучаетесь в условиях использования дистанционных образовательных технологий, вышлите все файлы с подготовленными Вами программами преподавателю.*

Лабораторная работа 13.

ПОСТРОЕНИЕ ГРАФИКОВ ФУНКЦИЙ

Задание 66

В электронном пособии (Пособия\3_Разработка программ.pdf) изучите раздел «Построение графиков и работа с ними» (с. 7-16).

Проверьте работоспособность программ, приведенных в разделе.

Задание 67

Используя программу из файла **ex_13_1.cpp**, постройте график функции $f(x) = 2x + 30$. Убедитесь, что программа работает правильно (из алгебры Вам известно, что графиком такой функции является прямая линия, пересекающая оси Ox и Oy в точках $(-15,0)$ и $(0,30)$ соответственно).

Задание 68

Пользуясь своей программой, постройте графики функций:

а) $f(x) = |2x + 30|$;

б) $f(x) = |||x - 80| - 40| - 20| - 10|$

(вместо знака модуля используйте функцию **abs** из библиотеки **math.h**).

Задание 69

Попробуйте с помощью той же программы построить график функции $f(x) = 0.01x^2 + 3x - 7$. Удалось ли это Вам? Если нет, давайте разберемся, что произошло. А дело вот в чем: координаты точки на экране обязательно должны быть целыми числами, а значения функции $f(x)$ оказываются в этом примере вещественными. Придется изменить описание переменных **x** и **y** на **float**. Но это еще не все. Изменить нужно не только описание, но и вычисление значения переменной, округлив его до целого, поскольку координаты точки в процедуре **PutPixel** должны быть целыми. Для этого в языке Паскаль предусмотрена функция **round** – округление. Используется она так: $a = \text{round}(b)$, где a – переменная типа **Integer**, а b – типа **float**. Внесите изменения в программу и отладьте ее.

Задание 70

Воспользуйтесь усовершенствованной программой для построения графика функции $f(x) = \sin(x)$. Похож ли полученный график на знакомую из алгебры синусоиду? Не очень, правда? Опять что-то не так. Действительно, значение функции $\sin(x)$ лежит в пределах от -1 до $+1$, а график ее лежит совсем близко к оси Ox , почти сливаясь с ней. Когда вы рисуете график в тетради, вы интуитивно выбираете такой масштаб, чтобы график был хорошо виден. Компьютеру же надо дать точные указания о масштабе. Если умножить значение функции, например на 50, то ординаты точек будут находиться в интервале шириной не 2, а 100 точек, и это уже будет хорошо видно на экране. Кроме того, чтобы масштаб по вертикали и по горизонтали был одинаковым, нужно при переходе от экранной координаты **x** к «реальной»,

поделить экранную координату на то же число (в данном случае 50), на которое умножили значение координаты **у**.

Если самостоятельно не удастся реализовать эту идею, разберитесь в программе **ex_13_2.cpp**.

Задание 71

Пользуясь рассмотренной в лекции программой построения графика в полярной системе координат (файл **ex_13_3.cpp**), постройте график функции $r(a) = a$. У вас должен получиться кусок спирали, (часть графика может оказаться за пределами экрана).

Модифицируйте программу, чтобы нарисовать:

- а) окружность;
- б) спираль меньшего размера;
- в) несколько оборотов спирали;
- г) цветок, задаваемый в полярных координатах функцией $r(a)=\sin(n*a)$, где параметр n задает число лепестков цветка. (Если ваш цветок будет слишком маленьким, воспользуйтесь идеей изменения масштаба, описанной в предыдущем задании.)

Задание 72

Смоделируйте на экране дисплея движение шарика:

- а) по окружности;
- б) по спирали.

Задание 73

Нарисуйте на экране несколько цветков, различных по окраске, размерам, количеству лепестков.

Задачи для самостоятельной работы

1. Постройте следующие кривые, заданные в полярных координатах:

- трехлепестковую розу: $r=2 \sin 2\varphi$;
- циссоиду: $r=\operatorname{tg} \varphi \cdot \sin \varphi$;
- гиперболическую спираль: $r = \frac{2}{\varphi}$;
- строфоиду: $r = \frac{1 + \sin \varphi}{\cos \varphi}$; ($r = \frac{1 - \sin \varphi}{\cos \varphi}$);
- лемнискату Бернулли: $r^2 = \cos 2\varphi$;
- улитку Паскаля: $r = a (1 + b \cos \varphi)$;
- функция $r = \sqrt{\frac{\pi}{\varphi}}$.

2. Постройте графики функций, заданных параметрически:

- кардиоиду: $x = a(t - \sin t)$; $y = a(1 - \cos t)$;

- астроиду: $x = a \cos^3 t; y = a \sin^3 t;$
- декартов лист: $x = \frac{3t}{1+t^3}, y = \frac{3t^2}{1+t^3}.$

3. Постройте следующие кривые, заданные в декартовой системе координат:

- $y = \frac{x^3 - 1}{4x^2};$
- $y = (x^2 - x - 6)^2;$
- $|x| + |y| = 1;$
- $||x| + |y| - \frac{3}{2}| = \frac{1}{2}.$

4. Изобразить на экране отрезок, вращающийся в плоскости экрана вокруг:

- а) своей середины;
- б) своего конца.

5. Изобразить на экране два отрезка, вращающихся в плоскости экрана каждая вокруг своего конца и со своей угловой скоростью.

6. Изобразить одновременное вращение двух стрелок - большой и малой, при котором одному полному обороту большой стрелки соответствует 1/12 оборота малой стрелки (как на циферблате часов). Стрелки можно для простоты заменить отрезками.

7. Изобразить на экране правильный треугольник, вращающийся в плоскости экрана вокруг своего центра.

8. Изобразить на экране разносторонний треугольник, вращающийся в плоскости экрана вокруг своего центра тяжести.

9. Изобразить на экране прямоугольник, вращающийся в плоскости экрана вокруг своего центра.

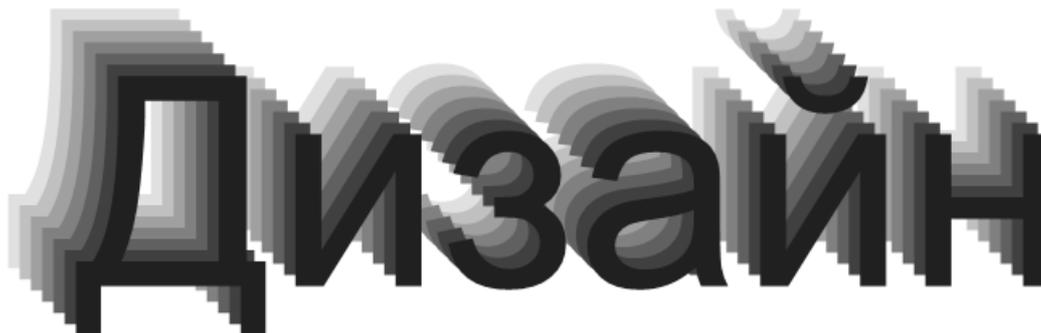
10. Изобразить на экране прямоугольник, вращающийся в плоскости экрана вокруг одной из своих вершин. Цвет фигуры должен изменяться при переходе от предыдущего положения к следующему.

11. Круглое кольцо вращается с постоянной угловой скоростью вокруг своего диаметра, расположенного параллельно горизонтальной оси экрана. Изобразить на экране процесс вращения. Считать, что в момент времени t кольцо выглядит для наблюдателя как эллипс, большая ось которого равна постоянной величине c , а малая равна $c \cos \omega t$, где ω - угловая скорость вращения.

12. Изобразить равнобедренный треугольник, вращающийся с постоянной угловой скоростью вокруг своей высоты, расположенной параллельно вертикальной оси экрана.

13. Изобразите на экране движение шара по бильярду без луз. Используйте правило: угол отражения равен углу падения. Начальное направление движения должно задаваться.

14. Создайте мультфильм «Круги на воде», используя не менее семи концентрированных окружностей. Центры окружностей должны быть совмещены с центром окна, а радиусы изменяться, увеличиваясь с каждой следующей окружностью. Иллюзия движения должна создаваться последовательной сменой цветов всех окружностей, начиная с внутренней и кончая внешней. Процесс смены цветов должен происходить бесконечно.
15. Создайте мультфильм «Капли дождя». На экране в случайном месте должна появляться окружность, увеличиваться ее радиус и одновременно и «выцветать». Процесс падающих капель должен завершаться по закрытию окна.
16. Составьте программу, которая выводит на экран текст в следующем виде:



- ☞ *Сохраните все программы в своей папке. Пригласите преподавателя и продемонстрируйте ему все выполненные Вами задания. Будьте готовы ответить на вопросы преподавателя. После беседы с преподавателем выйдите из среды Dev-C++.*
- ☞ *Если вы обучаетесь в условиях использования дистанционных образовательных технологий, вышлите все файлы с подготовленными Вами программами преподавателю.*

Лабораторная работа 14.

ИСПОЛЬЗОВАНИЕ ПРОЦЕДУР

Задание 74

В электронном пособии (Examples\Пособия\Введение в программирование.pdf) изучите раздел «Процедуры» (с. 35).

Рассмотрим использование процедур на примерах.

Задание 75

1. Запустите среду Dev-C++.
2. Загрузите файл **Proc2.cpp**.
3. Проанализируйте текст программы. Это сделать будет совсем нетрудно, так как Вы уже знакомы с алгоритмом рисования забора.
4. Найдите описание процедуры. Ответьте на следующие вопросы:
 - а) какое у процедуры имя?
 - б) сколько параметров у процедуры?
 - в) Какие из них входные, выходные?
 - г) Найдите конец процедуры. Чем он отличается от конца программы?
5. Найдите тело головной программы. Сколько раз вызывается процедура? Посмотрите, как описаны параметры. Совпадает ли это с описанием формальных параметров? Если найдете несоответствие в описании параметров, то обратитесь к преподавателю.
6. Запустите программу на выполнение. Проверьте, правильно ли она работает.
7. Модифицируйте программу так, чтобы делала 2 забора по 5 досок.
8. Заставьте программу рисовать пять заборов друг под другом, но чтобы высота забора была в два раза меньше.

Задание 76

Загрузите файл **Proc5.cpp**. В файле приведена программа, в которой идет отладка процедур рисования дирижабля и мишени. Используя данный пример как основу, разработайте две процедуры, которые рисуют:

- | | |
|---------------|------------|
| а) дерево; | г) машину; |
| б) человечка; | д) солнце; |
| в) цветок; | е) забор. |

(Выберите, что вам больше нравится, и подумайте, как можно будет задавать параметры этих объектов – размер, цвет и т.п.).

Текст доработанной программы с добавленной процедурой запишите с новым именем в свою папку.

Задание 77

1. Загрузите файл **Proc6.cpp**.

2. Перед вами программа TOWN – она строит город из домов. Проанализируйте структуру программы. Она содержит процедуру HOUSE, которая умеет строить дом по заданным параметрам. Посмотрите, какие параметры имеет процедура.
3. Найдите вызовы процедуры. Сколько, на ваш взгляд, будет построено домов?
4. Запустите программу на выполнение и проверьте свой ответ.
5. Придумайте свой город (свою комбинацию домов) и, исправив в головной программе вызовы процедуры, получите свою картинку. Покажите ее преподавателю.

Задание 78

1. В одном из предыдущих заданий Вы составили свою процедуру с изображением некоторых объектов (деревья, человечки, цветы, солнце). Оформите ее в виде процедуры и добавьте в программу TOWN.
2. Если у вашего соседа по компьютерному классу своя процедура для рисования объектов, то возьмите ее у него. Обратите внимание: если оформлять алгоритмы в виде процедур, то одну программу могут разрабатывать сразу несколько человек. Это очень удобно!

Задачи для самостоятельной работы

1. Составьте процедуру, которая вместо даты, заданной в числовом виде (25.06), выводит его текстовое название (двадцать пятое июня).
 2. Составьте процедуру, которая определяет в заданном предложении все слова-перевертыши и выводит их на экран (*казак, шалаи*).
 3. Составьте процедуру выравнивания строки, которая преобразует ее к строке указанной длины путем добавления пробелов между словами.
 4. Разработайте процедуру, которая рисует в заданном месте произвольный закрашенный треугольник.
 5. Разработайте процедуру, которая рисует параллелепипед по заданным параметрам.
 6. На основе процедуры **proc7.cpp** составьте процедуру, которая закрашивает задаваемую прямоугольную область окна всеми возможными цветами.
- ☞ *Пригласите преподавателя и продемонстрируйте ему сделанные Вами программы и процедуры. Будьте готовы ответить на вопросы преподавателя. После беседы с преподавателем выйдите из среды Dev-C++.*
- ☞ *Если вы обучаетесь в условиях использования дистанционных образовательных технологий, вышлите все файлы с подготовленными Вами программами преподавателю.*

Лабораторная работа 15.

ПОДПРОГРАММЫ - ФУНКЦИИ

Задание 79

В электронном пособии (Examples\Пособия\Введение в программирование.pdf) изучите раздел 8 «Функции» (с. 38-41).

1. Теория

Подпрограммы-функции предназначены преимущественно для вычисления какого-либо значения (числового или текстового). У них три основных отличия от процедур.

Первое отличие – в записи заголовка. В заголовке вместо void указывается тип возвращаемого значения. Например:

```
int <имя>(<список параметров>)
```

Список параметров может быть пустым.

Второе отличие заключается в том, что в теле функции должно быть указано значение, которое и есть результат работы функции. Реализуется это указание за счет оператора **return**.

Третье отличие заключается в том, что результат функции должен быть использован либо в арифметическом, либо в логическом выражении.

Рассмотрим пример вычисления факториала числа N. Известно, что $N! = N*(N-1)*(N-2)*(N-3)*...*2*1$

Подпрограмма-функция для вычисления факториала может иметь следующий вид:

```
Int factorial(int n);  
{  
  int fact, i;  
  Fact=n;  
  for (i=n-1; i>=2; i--) fact:=fact*i;  
  return fact;  
};
```

Для вызова функции из основной программы или другой подпрограммы следует в выражении, где необходимо использовать значение функции, указать имя функции со списком фактических параметров, которые должны совпадать по количеству и типам с формальными параметрами.

Так вызов вышеописанной функции Factorial может иметь следующий вид:

```
part=factorial(20)/b;
```

Вызывается функция Factorial с фактическим параметром 20 (параметр-значение)

```
part=factorial(m)+c/2;
```

Вызывается функция Factorial с фактическим параметром M (параметр-переменная)

```
part:=factorial(m+l);
```

В качестве параметра передается выражение, значение которого должно иметь тот же тип, что и описанный в функции формальный параметр.

2. Практика

Задание 80

Запустите среду Dev-C++.

Составьте программу вычисления величины S по формуле $S = F(b) - F(a)$, где

$$F(x) = \begin{cases} \cos x, & \text{при } x \leq 0 \\ \ln(x + 1), & \text{если } x > 0. \end{cases}$$

Вычисление функции $F(x)$ оформите в виде подпрограммы-функции. Подберите тесты для проверки правильности работы программы. Если убедитесь, что ваша программа работает корректно, покажите ее преподавателю.

Задание 81

Вычисление значения функции $y = x^n$ может быть оформлено в виде рекурсивной функции:

$$x^n = \begin{cases} 1 & , \text{ если } n=0 \\ x \cdot x^{n-1} & , \text{ если } n>0. \end{cases}$$

Выполните следующие действия:

- а) загрузите файл **ex_15_1.cpp** и посмотрите, как это записано в виде рекурсивной функции;
- б) используя эту функцию, напишите программу, которая для заданных значений N и M вычисляет и выводит на экран значение функции $z = x^n + y^n$;
- в) попробуйте переделать подпрограмму-функцию таким образом, чтобы в ней не было рекурсии. Можно использовать простую итерацию: $x^n = x \cdot x \cdot x \cdot \dots \cdot x$ (n - раз) ;
- г) запустите на выполнение программу с использованием своей новой процедуры. Оцените время работы программы и память.

Задание 82

Наверное, все вы умеете находить наибольшее кратное двух чисел. Попробуйте составить функцию, которая будет находить наибольший общий делитель для двух положительных целых чисел.

Задание 83

Сравните свою функцию с той, что находится в файле **ex_15_2.cpp**. Составьте схему работы функции для чисел 9 и 12. Сколько раз будет вызываться функция? С какими аргументами? Попробуйте добавить отладочную печать и проверить свои рассуждения.

Задание 84

Большое число практических задач требует каким-то образом «усреднить» данные. Это и обработка результатов научных экспериментов, и задачи статистики, и т.п.

Самое простое – это взять среднее арифметическое всей последовательности чисел. Другой способ – взять среднее арифметическое между наибольшим и наименьшим элементами последовательности. Еще один способ, более сложный, применяется, в частности, при судействе соревнований по фигурному катанию. Состоит он в том, чтобы отбросить крайние (наибольшее и наименьшее) числа, а от остальных взять среднее арифметическое.

Запрограммируйте все эти способы для вводимой последовательности чисел с помощью отдельных функций. Сравните результаты.

Задание 85

Составьте программу, которая позволяла бы определять наибольший общий делитель для трех заданных чисел.

Задачи для самостоятельной работы

1. Составьте функцию, которая вместо даты, заданной в числовом виде (25.06), выводит его текстовое название (двадцать пятое июня).
2. Часто бывает важно не только определить среднее значение последовательности, но и охарактеризовать величину «разброса». Одной из таких характеристик может служить, например, разность между наибольшим и наименьшим значениями. С помощью функции запрограммируйте вычисление этой величины для вводимой последовательности чисел.
3. Составьте функцию, с помощью которой можно будет определять расстояние между двумя точками в трехмерном пространстве.
4. Разработайте функцию, с помощью которой определяется, является ли введенное число простым.
5. Составьте процедуру, которая приводит строку, состоящую из русских символов, к одному регистру (например, прописные буквы преобразует в строчные).
6. Составьте процедуру, с помощью которой из заданного текста будут удалены все английские символы.
7. Составьте процедуру, являющуюся синтаксическим анализатором идентификатора. Процедура должна получать строку и проверять, является ли полученная строка идентификатором (т.е. последовательностью символов, состоящую из английских букв, цифр и(или) знака подчеркивания, начинающейся с буквы).

 *Пригласите преподавателя и продемонстрируйте ему сделанные Вами программы. Будьте готовы ответить на вопросы преподавателя.*

 *Если вы обучаетесь в условиях использования дистанционных образовательных технологий, вышлите все файлы с подготовленными Вами программами преподавателю.*

Лабораторная работа 16.

КАК РАБОТАТЬ С МАССИВАМИ: ПЕРВЫЕ ШАГИ

Задание 86

В электронном пособии (Examples\Пособия\Хранение и обработка данных.pdf) изучите раздел «Массивы» (с.3-6).

Задание 87

Запустите среду Dev-C++.

Составить программу, в которой:

- а) всем элементам массива присваивается заданное целое значение (количество элементов массива – не более 50);
- б) всем элементам массива присваивается вводимый текст;
- в) каждому элементу массива с нечетным индексом присваивается его номер, а элементу с четным индексом - его номер с противоположным знаком;
- г) элементам массива, стоящим на четных местах, присваивается символ "@", на нечетных местах - "#".
- д) каждому элементу, стоящему на 1-м, 4-м, 7-м и т.д. местах, присваивается число 0, остальным – число 3, и все элементы массива выводятся на экран дисплея.

Задание 88

Загрузите в новое окно программу из файла **array_1.cpp**. Эта программа позволяет ввести последовательность из 15 чисел и вывести их в обратном порядке. Разберитесь в том, как работает эта программа. Выполните задание, написанное в конце программы.

Задание 89

Модифицируйте программу из файла **array_1.cpp** так, чтобы сначала печатались в строчку все положительные числа, а в следующей строке – все отрицательные.

Задание 90

Представьте себе, что Вы смешиваете несколько компонентов в одну смесь (может быть, Вы создаете новые духи, может быть, особо прочный и легкий сплав металлов и т.п.). По окончании работы Вы знаете, сколько чего входит в смесь, т.е. массу каждого компонента. Чтобы в дальнейшем изготавливать такую же смесь в промышленных масштабах, вам нужен «рецепт», т.е. процентное содержание каждого компонента в смеси.

Такая программа составлена и размещена в файле **array_2.cpp**. Загрузите этот файл. Модифицируйте программу, размещенную в этом файле, так, чтобы она запрашивала не только массы, но и числовой шифр компонентов и печатала их в рецепте. Для этой программы вам потребуется два массива – один, как и прежде, для хранения масс, а другой – для хранения шифров (чтоб никто не догадался).

Задание 91

В массиве хранится информация о количестве осадков за каждый день ноября. Составьте программу, с помощью которой определяется общее количество осадков за этот месяц.

Задание 92

В массиве хранится информация о количестве студентов в каждой группе. Составьте программу, с помощью которой определяется среднее число студентов в одной группе.

Задание 93

В электронном пособии (Examples\Пособия\Хранение и обработка данных.pdf) изучите подраздел «Простой поиск в массиве» (с.11-14). После прочтения выполните следующие задания.

Задание 94

В массиве хранится информация о результатах квалификационных выступлений фигуристов. На соревнования допускаются только те, кто получил результаты выше среднего. Составьте программу, которая определяет число спортсменов, прошедших квалификацию.

Задание 95

В массиве хранится информация о результатах соревнований по прыжкам в длину. Составьте программу, с помощью которой определяется лучший и худший из результатов.

Задание 96

В массиве содержатся оценки ученика по математике. Сосчитайте общее количество пятерок, четверок, троек и двоек.

Задачи для самостоятельной работы

1. В массиве рост юношей задан отрицательным числом, рост девушек – положительным. Определите отдельно средний рост юношей и девушек.
2. В массиве хранится информация о стоимости книг. Составить программу, которая выводит на экран номера книг, которые стоят ниже средней стоимости всех этих книг.
3. В массиве хранится число жителей в каждом доме улицы. Индекс элемента массива соответствует номеру дома. Известно, что на одной стороне улицы дома с четными номерами, а с другой – с нечетными. Составьте программу, которая определяет, на какой стороне улицы больше жителей: на четной или нечетной.
4. Определить, имеются ли в заданном массиве одинаковые элементы.
5. Определить, является ли заданный массив упорядоченным по возрастанию. В случае отрицательного ответа вывести номер элемента, нарушающего эту упорядоченность.
6. В массиве хранятся года рождения некоторого числа людей. Составьте программу, которая определит разницу в возрасте между самым старшим и

самым младшим из этого списка. Считать, что возраст можно определить как разность между текущим годом и их годом рождения.

7. В массиве хранится информация о стоимости различных марок автомобилей. Определите два самых дорогих автомобиля из этого списка. Попробуйте решить эту задачу, используя один проход по массиву.
8. Удалите из введенного файла отрицательные элементы. В конец массива впишите нули.
9. Известна стоимость автомобилей и их мощность (в л.с.). Определить общую стоимость автомобилей, имеющих мощность больше 100 л.с.
10. Для характеристики «разброса» элементов последовательности в статистике используется величина, называемая дисперсией.

Дисперсия – это среднее арифметическое квадратов разностей между элементами последовательности и ее средним значением, т.е. вычисляется по формуле:

$$D = \frac{(a_1 - M)^2 + \dots + (a_n - M)^2}{n},$$

где a_i - элементы последовательности, а M – среднее значение этих элементов.

Напишите программу для вычисления дисперсии (и обязательно сохраните – она нам еще понадобится!).

11. Для иллюстрации дискретных динамических процессов довольно часто используются столбиковые диаграммы (гистограммы) (рис. 4).

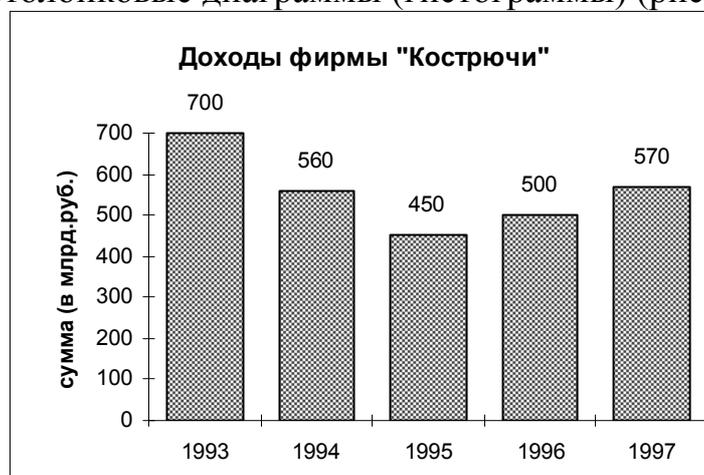


Рис. 4. Пример столбиковой диаграммы (гистограммы)

Составьте программу, которая по введенным данным строила бы такую гистограмму на экране. Сделайте 2 варианта программы: из 2-мерных столбиков (рис. 4) и 3-мерных параллелепипедов (рис. 5).

Уточнение

Программа будет универсальной только в том случае, если она будет работать для «произвольного» (например, от 1 до 100) набора чисел и для любых значений этих чисел. Чтобы столбики были пропорциональны и помещались на экране, требуется пропорционально увеличивать размеры

столбиков, если значение чисел малы, и уменьшать, если они слишком велики (этот процесс назовем масштабированием).

Цвета каждого из столбиков и подписи под ними также следует вводить.

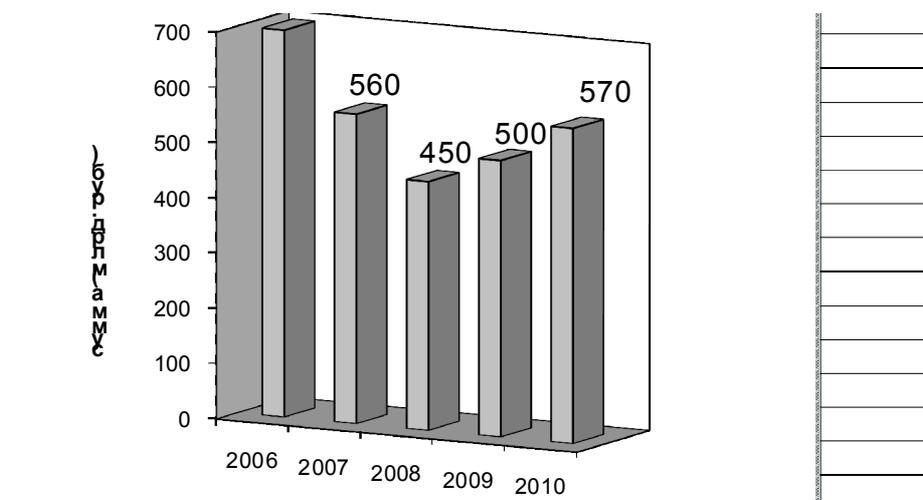


Рис. 5. Пример трехмерной гистограммы

12. Напишите программу, которая строит круговую диаграмму, интерпретирующую последовательность заданных чисел (рис. 6).

Расходы фирмы "Мальвина" (млрд. руб.)

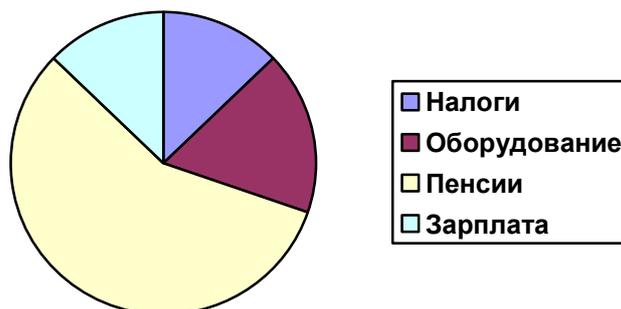


Рис. 6. Пример круговой диаграммы

Подсказка

Предположим, заданы числа 3, 5 и 2. Сумма этих чисел: $3+5+2=10$. Следовательно, число 3 составляет 30% от суммы, 5 – 50% и 2 – 20%. 30% окружности – это $360 \cdot 0,3=108^\circ$, 50% – 180° и 20% – 72° . В этом случае Вам следует построить окружность из 3 разноцветных секторов, величина которых соответствует 108° , 180° и 72° . Это и будет круговая диаграмма, соответствующая заданной последовательности чисел.

Решите задачу в общем случае.

- 👉 Пригласите преподавателя и продемонстрируйте сделанные Вами программы. Будьте готовы ответить на его вопросы.
- 👉 Если вы обучаетесь в условиях использования дистанционных образовательных технологий, вышлите все файлы с подготовленными Вами программами тьютору.

Лабораторная работа 17*.

СОРТИРОВКА МАССИВОВ

Задание 97

В электронном пособии (Examples\Пособия\Хранение и обработка данных.pdf) изучите подраздел «Сортировка массивов» (с.16-18).

Скопируйте тексты программ и проверьте их работоспособность.

Задание 98

Имеется список спортсменов, участвующих в соревнованиях по прыжкам в длину, и их результаты. Составить программу, которая позволит вывести список участников на экран дисплея в порядке занятых ими мест.

Задание 99

Введите и отладьте программу сортировки числового массива по возрастанию методом «всплывающего пузырька». Для того чтобы увидеть результаты работы программы, включите в нее ввод данных и печать результатов.

Задание 100

Измените программу так, чтобы массив сортировался в порядке убывания.

Задание 101*

Напишите алгоритм слияния двух массивов, отсортированных в порядке возрастания, в один массив, также отсортированный в порядке возрастания.

Самый простой способ решения этой задачи состоит в том, чтобы попросту «склеить» эти два массива, а затем отсортировать получившийся массив, например, методом пузырька. Подсчитайте количество сравнений, которые придется выполнить.

Другой путь: сразу помещать элементы в новый массив в нужном порядке, пользуясь тем, что исходные массивы уже упорядочены. Реализуйте эту идею. Подсчитайте количество сравнений и сравните с предыдущим результатом. Понятно, что этот путь предпочтительнее.

Задание 102*

Составить программу обслуживания конькобежных соревнований. Программа должна:

- 1) обеспечивать ввод фамилия спортсмена и его результат;
- 2) выводить после каждого ввода информации о спортсмене полного списка спортсменов (фамилии и результата) в порядке занятых мест.

Задачи для самостоятельной работы

☞ Следующие задачи предполагают работу с массивом в 10-20 элементов. Если по условию задачи вводимая последовательность чисел любая, создайте процедуру, которая задаст значения этого массива в заданном диапазоне, после чего, собственно, и начинайте разработку программы. Если последовательность должны отвечать каким-то условиям, задайте ее в виде набора констант.

1. Пусть задан массив из 20 элементов, содержащий информацию о количестве очков, которые заработала каждая из теннисисток. Составьте программу, определяющую первую десятку в данном виде спорта.
2. Пусть задан массив из 20 элементов, содержащую информацию о 20 участниц конкурса красоты (точнее, количество набранных ими очков). Призы выдаются обычно следующим образом: сначала 10-е место, затем 9-е, затем 8-е, и так до 1-го. А второй десяток остается по возрастианию. Составьте программу, которая распределит результаты по этому алгоритму и выведет их на экран.
3. Пусть задан массив из 20 элементов, содержащий информацию о количестве очков, которые заработал каждый из участников конкурса «Учитель года». Количество участников, занявших одно из призовых мест (их бывает до 10), на самом деле не ограничено. Например, если первые три человека набрали одинаковое количество очков, то есть три первых места и т.д. Составьте программу, которая задает подобный массив, сортирует его и выводит количество 1-х, 2-х, ..., 10-х мест.
4. Организуйте массив, содержащий 15 различных целых чисел. После этого отдельно первых 5 элементов, вторых 5 элементов и последних 5 элементов сортируются по возрастанию. Содержимое отсортированного таким образом массива выводится на экран.
5. Организуйте массив, содержащий 20 различных символов. Отсортируйте его по возрастанию.
6. Организуйте массив, содержащий 20 целых чисел. Отсортируйте отдельно элементы с чётными индексами по возрастанию, и элементы с нечётными индексами по убыванию.
7. Создайте массив, содержащий 20 целых чисел. Отсортируйте его по возрастанию. После этого определите и выведите на экран сумму элементов с чётными индексами и сумму элементов с нечётными индексами.
8. Создайте массив, содержащий 15 целых чисел. Отдельно первых 5 элементов массива, вторых 5 элементов и последних 5 элементов отсортируйте по убыванию. Определите и выведите на экран сумму каждой пятёрки отсортированного таким образом массива.
9. Создайте массив, содержащий 15 различных символов. Отсортируйте его по убыванию. После этого определите и выведите на экран «наименьший» и «наибольший» символы.

10. Создайте массив, содержащий 10 различных символов. Первую половину массива отсортируйте по возрастанию, а вторую по убыванию. Отсортированный массив выведите на экран.
11. Создайте массив, содержащий 8 различных символов. Отсортируйте его по возрастанию. Организуйте и выведите на экран другой целочисленный массив, заполнив его числами, полученными преобразованием символов первого массива в целые числа.
12. Создайте целочисленный массив А, содержащий 10 различных чисел. Отсортируйте первую половину массива А по возрастанию, а вторую по убыванию. Организуйте и выведите на экран символьный массив В, заполнив его символами, полученными преобразованием чисел массива А в символы.
13. Создайте массив, содержащий 20 различных целых чисел. Отсортируйте его по возрастанию. После этого замените все элементы массива на противоположные и выведите содержимое обработанного массива на экран.
14. Создайте массив, содержащий 20 различных целых чисел. Отсортируйте первую половину массива по возрастанию, а вторую по убыванию. Все чётные элементы массива увеличить в три раза, а нечётные в 2 раза. Содержимое обработанного таким образом массива вывести на экран.
15. Дан массив, содержащий рост 25 учеников класса. Отсортируйте массив по убыванию. В начале учебного года в класс поступили два новых ученика. Получить аналогичный массив, учитывающий рост новых учеников.
16. Дан одномерный массив. Упорядочить массив, удалив нули со сдвигом влево ненулевых элементов.

☞ *Пригласите преподавателя и продемонстрируйте ему сделанные Вами программы. Будьте готовы ответить на его вопросы.*

☞ *Если вы обучаетесь в условиях использования дистанционных образовательных технологий, вышлите все файлы с подготовленными Вами программами тьютору.*

Лабораторная работа 18.

ОСНОВНЫЕ ОПЕРАЦИИ СО СТРОКОВЫМИ ДАННЫМИ

Задание 103

В электронном пособии (Пособия\Хранение и обработка данных.pdf) изучите раздел «2. Символьные строки» (с. 21-23).

1. Алгоритмы замены фрагментов строки

Задание 104

Считайте с диска файл с именем **stroka_01.cpp**. Он представляет собой программу поиска всех букв «а» и печать номеров позиций этих букв в строке символов. Запустите эту программу, введите строку, содержащую нужные символы, и убедитесь, что программа работает правильно.

Модифицируйте программу так, чтобы:

- а) подсчитывалось и печаталось число букв «а»;
- б) учитывались еще и заглавные буквы «А».

Задание 105

Считайте с диска файл с именем **stroka_02.cpp**. Он содержит программу замены всех букв «а» на «о».

Модифицируйте программу так, чтобы происходила замена буквы «а» на букву «б» и наоборот. Например, слово «баба» программа превращала бы в «абаб». Сделайте то же с учетом заглавных букв.

Задание 106

В файле **stroka_03.cpp** находится программа поиска в строке всех заданных фрагментов. Разберитесь, в чем ее отличие от предыдущих программ. Модифицируйте программу так, чтобы вместо фрагмента «мама» отыскивался фрагмент «ма» (в слове «мама» он должен обнаруживаться дважды).

Задание 107

В электронном пособии (Пособия\Хранение и обработка данных.pdf) изучите раздел «Функции работы со строками» (с. 25-35). Разберитесь внимательно в приведенных примерах программ, которые можно скопировать и проверить их работу в среде Dev-C++.

Процедуры и функции работы со строками расположены в библиотеке `string.h`. К сожалению, в C/C++ весьма ограниченные возможности работы со строками. Возможностей базовых функций `strlen`, `strcpy`, `strncpy`, `strcmp`, `strncmp`, `strcat`, `strchr`, `strrchr` и `strstr` недостаточно для решения ряда типовых задач. Чтобы разрешить эту проблему, подготовим набор дополнительных процедур.

В пособии показаны ограниченные возможности процедуры `strcpy`, которая умеет вставлять символы из одной строки в другую, но вставка осуществляется по технологии замены: просто с определенного места исходные символы стираются, а копируемые записываются. Чаще требуется при копировании сдвигать

исходные символы вправо, а на вновь образованное место вставлять символы из вставляемой строки. Вот эту процедуру и требуется запрограммировать.

Задание 108

Разработайте универсальную процедуру **copy**, которая обеспечивает копирование символов из одной строки в другую в режиме вставки.

В качестве подсказки используйте процедуру **ins** из файла **stroka_04.cpp**, которая вставляет в текст заданное число пробелов.

Задание 109

Разработайте универсальную процедуру **strdel**, которая обеспечивает удаление указанного количества символов с заданного места строки.

Задание 110

В файле **stroka_05.cpp** находится программа поиска в строке всех заданных фрагментов. Разберитесь, в чем ее отличие от предыдущих программ. Модифицируйте программу так, чтобы вместо фрагмента «мама» отыскивался фрагмент «бабушка».

Задание 111

Составьте программу замены одного слова строки другим.

Программа должны работать так, чтобы происходила замена только целого слова на слово (так, чтобы, заменяя, например, «полковник» на «генерал», не получить нового звания «подгенерал»).

При проверке работы программы обратите особое внимание на слова, стоящие в начале и в конце строки.

2. Алгоритмы распознавания слов и чисел в строке

Задание 112

Напишите программу, подсчитывающую количество слов во введенной фразе, если:

- а) между словами стоит один пробел;
- б) между словами может стоять произвольное число пробелов;
- в) между словами стоит один произвольный разделитель (пробел, ",", ".", "!", "?", ";", "-");
- г) между словами стоит произвольное число разделителей.

Задание 113

В файле **stroka_06.cpp** хранится программа, которая обеспечивает преобразование целого числа в строку, а в файле **stroka_06a.cpp** – программа преобразования строки в целое число, а **stroka_06b.cpp** – программа преобразования строки в вещественное число. Разберитесь, как они работают. Повводите разные (допустимые и недопустимые) данные и выясните реакцию на ошибочный ввод. В частности, для программы **stroka_06a.cpp** посмотрите, что произойдет, если на вводе задать:

- а) целое число;
- б) строку, содержащую цифры и буквы (например, 123абв);
- в) дробное число (например, 12.34).

Задание 114

Используя идеи, предложенные в предыдущем задании, напишите программу, вычисляющую сумму всех целых чисел, находящихся во введенной строке.

Подсказка

Для решения задачи достаточно вырезать все целые числа из строки, преобразовать их в числовые данные и добавить в сумму.

Задание 115

Составьте простейший калькулятор, обеспечивающий ввод строки с простейшим арифметическим выражением вида

$$a1 <оп> a2 <оп> a3 \dots <оп> aN,$$

где $a1\dots aN$ – целые числа, $<оп>$ – знак операции сложения (+) или вычитания (-) и вычисляющий результат описанных действий.

Задание 116

Составьте программу, которая вводит строку и выводит все слова этой строки в столбик.

Модифицируйте программу так, чтобы печатались только:

- а) слова, начинающиеся на букву «л»;
- б) слова, заканчивающиеся на букву «ь»;
- в) слова длиной не менее пяти букв;
- г) самое длинное слово.

Задание 117

Подсчитайте, сколько пятерок потребуется, чтобы записать все числа от 1 до 999.

Задачи для самостоятельной работы

1. Напишите программу, которая в заданном предложении находит самое короткое и самое длинное слова. Напечатайте исходный текст и найденные слова.
2. Напишите программу, которая в заданном предложении удаляет все символы пробела. Напечатайте исходный и преобразованный тексты, а также количество удаленных символов.
3. В качестве исходного текста задайте арифметическое выражение, записанное на C/C++. Напишите программу, которая проверяет, соблюдается ли равенство открывающих и закрывающих скобок. Выведите на печать исходный текст и количество открывающих и закрывающих скобок.
4. Напишите программу, которая в заданном предложении символы самого длинного слова заменяет символами «*». Напечатайте преобразованный текст.

5. Напишите программу, которая в заданном предложении перед каждым словом вставляет символ «?». Напечатайте преобразованный текст, а также длину полученного предложения.
6. Напишите программу, которая в заданном предложении удаляет все запятые. Напечатайте преобразованный текст, а также число удаленных символов.
7. В заданном предложении проверить парность расстановки круглых скобок. Вложение скобок допускается.
8. Напишите программу печати слов предложения в обратном порядке.
9. Составить программу, которая шифрует и дешифрует введенный текст. В качестве ключа шифра можно взять произвольную неповторяющуюся последовательность символов.
10. Составить программу, в которое вводится двухзначное число и выводится его название (вводится «36» – выводится «тридцать шесть»).
17. Составить программу, которая вычисляет частоту использования русских букв в введенном тексте.

Протестируйте разработанную программу. Для этого запустите ее:

- а) сразу нажмите . Программа должна сообщить, что текст отсутствует;
 - б) введите текст "vvv". Программа должна сообщить, что в тексте русских букв нет;
 - в) введите «ююююю». Программа должна выдать только одну строчку: букв «ю» в тексте - 100%;
 - г) введите «мама». Программа должна выдать две строчки: букв «а» в тексте – 50% ; букв «м» в тексте – 50% ;
 - д) введите «123 АбвГ123деЖзий». Программа должна выдать информацию о том, что букв а,б,...,й в тексте по 10%.
11. Составить программу, проверяющую, является ли введенное предложение «перевертышем», т.е. читающееся одинаково слева направо и справа налево (без разделителей, разумеется). Примеры «перевертышей»: *аргентина манит негра*; *а роза упала на лапу азора*.

☞ *Сохраните все программы в вашей папке. Пригласите преподавателя и продемонстрируйте ему все сделанные Вами программы. Будьте готовы ответить на вопросы преподавателя. После беседы с преподавателем выйдите из среды Dev-C++.*

☞ *Если вы обучаетесь в условиях использования дистанционных образовательных технологий, вышлите все файлы с подготовленными Вами программами тьютору.*

Лабораторная работа 19.

РАБОТА С ТЕКСТОВЫМИ ФАЙЛАМИ: ЧТЕНИЕ И ЗАПИСЬ

До сих пор были рассмотрены способы работы с информацией, которая либо была заложена в программу, либо вводилась с клавиатуры компьютера. В данной лабораторной работе предлагается познакомиться с обработкой информации, размещенной в файлах на дисках.

В языке C/C++ предусмотрена работа с тремя типами файлов:

- 1) с текстовыми файлами;
- 2) с двоичными файлами.
- 3) со структурированными файлами.

Рассмотрим работу с двумя первыми видами¹.

Задание 118

В электронном пособии (Пособия\Хранение и обработка данных.pdf) изучите раздел «Работа с текстовыми файлами» (с. 23-25).

Проверьте работоспособность всех программ из данного раздела.

Задание 119

Запустите среду Dev-C++. Загрузите текст программы **file_t1.cpp**. Разберитесь в том, как она работает. Модифицируйте эту программу так, чтобы она выводила на экран содержимое *ЛЮБОГО* текстового файла. Для отладки можно использовать какой-либо файл из Вашего каталога (в том числе и с расширением PAS).

☞ *Следует иметь в виду, что тексты, хранящиеся в текстовом файле, должны быть в формате MS DOS.*

Задание 120

В электронном пособии (Пособия\Хранение и обработка данных.pdf) изучите раздел «Работа с текстовыми файлами» (с. 7-11).

Проверьте работоспособность всех программ из данного раздела.

Задание 121

В файле **numbers.txt** хранятся вещественные числа, записанные в столбик. Составьте программу, которая позволит Вам считать эту информацию и вычислить среднее арифметическое этих чисел. Известно, что эти числа могут быть любые: целые, вещественные, положительные, отрицательные. Количество чисел НЕИЗВЕСТНО!

Задание 122

Загрузите текст программы **file_t2.cpp**. Используя эту программу, проверьте правильность работы программы, составленной в предыдущем задании.

¹ Освоив работу с текстовыми и двоичными файлами, Вы без труда научитесь работать со структурированными файлами с помощью любого справочника.

Задание 123

Считайте с диска файл **file_t3.cpp**. Разберитесь в том, как работает программа:

- внимательно прочтите текст программы и комментарии;
- запустите программу на выполнение и введите не менее 10 чисел;
- считайте с диска файл **ex0.txt** и проверьте, правильно ли записались числа;
- модифицируйте программу так, чтобы числа печатались в одну строку и между ними был, по крайней мере, один пробел.

Задание 124

Составьте программу – простейший текстовый редактор. Алгоритм работы достаточно прост: в диалоге вводится имя файла. Затем предлагается вводить текст. После набора строки и нажатия клавиши **<ENTER>** введенная строка записывается в файл. В случае введенного специального символа (это может быть * или что-нибудь другое) файл закрывается и работа программы завершается.

Проверьте правильность работы Вашей программы с помощью программы, разработанной при выполнении задания 119.

Задачи для самостоятельной работы

1. Создайте файл, содержащий массив целых чисел. Напишите программу, вычисляющую наименьшее из этих чисел.
2. Создайте файл, содержащий массив целых чисел. Напишите программу, которая определяет количество элементов массива.
3. Создайте файл, содержащий массив вещественных чисел. Напишите программу, определяющую количество чисел, меньших среднего арифметического значения всех чисел массива.
4. Дан текстовый файл. Подсчитать количество символов в нем.
5. Имеется текстовый файл. Удалите из него первую строку, в конце которой стоит вопросительный знак. Результат записать в другой файл.
6. Создайте файл, содержащий массив вещественных чисел. Напишите программу, которая считывает числовые значения из файла, вычисляет значение полусуммы наибольшего и наименьшего элементов, а затем подсчитывает количество значений, не превышающих по величине полусумму и количество значений, больших, чем полусумма.
7. Напишите программу, которая считывает текстовый файл, выравнивает его содержимое по правой границе и выводит результат в другой текстовый файл.
8. Напишите программу, которая считывает текстовый файл, выравнивает его содержимое по центру и выводит результат в другой текстовый файл.
9. Создайте файл, содержащий фамилии студентов и их возраст. Напишите программу печати фамилий студентов, имеющих наименьший возраст.

10. Текстовый файл, содержит фамилии рабочих и их среднемесячный заработок. Напишите программу, выводящую фамилии рабочих, имеющих заработок, находящийся в указанных пользователем пределах.

 *Пригласите преподавателя и продемонстрируйте ему сделанные Вами программы. Будьте готовы ответить на вопросы преподавателя.*

 *Если вы обучаетесь в условиях использования дистанционных образовательных технологий, вышлите все файлы с подготовленными Вами программами тьютору.*

Лабораторная работа 20.

РАБОТА С ДАННЫМИ ИЗ ТЕКСТОВЫХ ФАЙЛОВ

Задание 125

Имеется текстовый файл `Shop_1.txt`, содержащий список товаров, их количество и стоимость единицы товара:

	холодильники	←	название товара	
→	10	900	←	стоимость единицы товара
	стиральный порошок			
	1000	5		
	микрокалькуляторы			
	300	12.6		
	...			

← количество единиц товара.

Составить программу, которая создаст новый файл, содержащий в одной строке информацию о данном виде товара (название, количество единиц товара и стоимость одной единицы) плюс общую стоимость товара, а в конце файла – итоговую стоимость товаров всего этого списка.

Задание 126

Модифицируйте программу, разработанную в предыдущем задании так, чтобы она решала эту же задачу, но для файла немного другой структуры (можно воспользоваться файлом `shop_2.txt`):

холодильники	* 10	900
стиральный порошок	* 1000	5
микрокалькуляторы	* 300	12.6
...		

Обратите внимание на то, что название товара кончается символом `*`, а затем уже идет количество и цена единицы товара.

Задание 127*

Имеется файл `numbers.txt`, содержащий последовательность вещественных чисел. Составить программу, которая перепишет все числа из этого файла в файл `num_1.txt` в порядке возрастания (убывания).

Задание 128*

Имеется два файла, содержащие списки фамилий сотрудников фондовой биржи. Фамилии в файлах расположены в алфавитном порядке. Составить программу, которая позволит на основе этих файлов создать третий, в который будут входить все фамилии, содержащиеся в первом и втором файлах. При этом фамилии должны быть также упорядочены по алфавиту.

Примечание

Для отладки программы составьте два таких файла прямо в редакторе среды Dev-C++. Проследите, чтобы в файлах было не менее пяти строк и информация была *ДЕЙСТВИТЕЛЬНО* упорядочена по алфавиту!

Задание 129*

Пусть есть текстовый файл, содержащий список (товаров, фамилий, марок автомобилей и т.д.). Составить программу, которая рассортирует этот список по алфавиту.

Задачи для самостоятельной работы

1. Создайте файл, содержащий фамилии студентов и их экзаменационные оценки по 5 предметам. Напишите программу печати фамилий студентов, сдавших экзамены только на "5".
2. Создайте файл, содержащий фамилии студентов. Напишите программу печати списка студентов, имеющих самые длинные фамилии.
3. Создайте файл, содержащий фамилии студентов и их экзаменационные оценки по 3 предметам. Напишите программу печати фамилий тех студентов, которые успешно сдали все экзамены.
4. 14. Создайте файл, содержащий фамилии студентов и их экзаменационные оценки по 3 предметам. Напишите программу печати фамилий студентов, имеющих хотя бы одну задолженность, и количество не сданных экзаменов.
5. Напишите программу, которая считывает текстовый файл и определяет, содержатся ли в нем символы латинского алфавита.
6. Создайте файл, содержащий произвольные текстовые строки. Напишите программу подсчета количества строк, начинающихся с буквы "a".
7. Напишите программу печати в файл таблицы значений функций $y = \sin x$ и $z = \operatorname{tg} x$ на отрезке $[0, 3]$ с шагом $dx = 0,1$. Значения x запишите с 1 цифрой в дробной части, значения y, z – с двумя цифрами.
8. Создайте файл, содержащий произвольные текстовые строки. Напишите программу печати строк, имеющих наименьшую длину.
9. Создайте файл, содержащий фамилии абонентов и их телефонные номера. Составьте программу, которая по заданной фамилии абонента находит его номер телефона.
10. Создайте файл, в котором содержатся семизначные номера телефонов и информация о сумме задолженности абонента данного телефона. Записи об одном абоненте могут повторяться. Напишите программу печати суммарной задолженности абонента по введенному номеру телефона.
11. Напишите программу, которая считывает исходный текст программы на C++ и подставляет номера строк (в виде комментариев) в начало каждой строки файла с исходным текстом.
12. Напишите программу, которая считывает записи из одного файла и записывает их в обратном порядке во второй файл.
13. Напишите программу, которая считывает текстовый файл и сравнивает содержащиеся в нем слова со словами, хранящимися в отдельном файле-словаре. В случае, когда слово не найдено, программа должна выдавать соответствующее сообщение.
14. Напишите программу, которая вводит с клавиатуры произвольный текст, затем записывает в первый файл закодированный текст, а во второй – дешифрованный, если известен алгоритм шифрования. Указание: код

шифрования – это число, указывающее смещение букв в алфавите (например, код 3 означает, что вместо буквы "а" в зашифрованном тексте указана буква "в").

15. Напишите программу, которая определяет количество предложений и слов в текстовом файле. Считайте, что предложением является любая последовательность символов, начинающаяся пробелом и заканчивающаяся точкой, восклицательным или вопросительным знаками.

 *Пригласите преподавателя и продемонстрируйте ему сделанные Вами программы. Будьте готовы ответить на его вопросы.*

 *Если вы обучаетесь в условиях использования дистанционных образовательных технологий, вышлите все файлы с подготовленными Вами программами тьютору.*

Лабораторная работа 21*.

МНОГОМЕРНЫЕ МАССИВЫ

Задание 130

В электронном пособии (Пособия\Хранение и обработка данных.pdf) изучите раздел «3. Матрицы (двухмерные массивы)» (с. 37-43).

Проверьте работоспособность всех программ из данного раздела (включая ввод матрицы из текстового и двоичного файлов).

Задание 131

Составьте программу, которая обеспечивает ввод двумерного массива целых чисел и вывод этого же массива, но в нем все числа больше 10 заменены на 0.

Задание 132

Составьте программу, которая обеспечивает ввод двумерного массива целых чисел и вывод этого же массива, но в нем все числа больше 10 заменены на 0.

Задание 133

Дана турнирная таблица, содержащая информацию о числе очков, полученных каждой командой в игре. Составьте программу вывода на экран суммарного числа очков, полученных каждой командой.

Задание 134

Имеется таблица, включающая фамилии студентов и оценки за 5 экзаменов. Составить программу:

- а) вывода фамилий отличников;
- в) вывода фамилий студентов, имеющих только оценки «уд»;
- г) вывода фамилий студентов, обучающихся только на «хор» и «отл»;
- д) вывода фамилий студентов, имеющих не более одной «уд»;
- е) вывода фамилий учащихся, имеющих двойки;
- ж) определения номера предмета, по которому учащиеся учатся хуже всех;
- з) определения номера предмета, по которому учащиеся учатся лучше всех.

Задание 135

Дана следующая таблица, содержащая данные о планетах Солнечной системы:

Планета	Расстояние до Солнца	Относит. объем	Относит. масса
Меркурий	57.9	0.06	0.05
Венера	108.2	0.92	0.81
Земля	149.6	1.00	1.00
Марс	227.9	0.15	0.11
Юпитер	978.3	1345.00	318.40
Сатурн	1429.3	767.00	95.20

Напишите программу, которая:

- а) определяет наиболее и наименее удаленные от Солнца планеты, а данные таблицы сортирует по возрастанию масс планет;
- б) определяет средний объем планет Солнечной системы и находит среди них ту, объем которой наименее отличается от среднего. Сортирует данные таблицы по возрастанию расстояний до Солнца;
- в) определяет самую тяжелую и самую удаленную от Солнца планеты. Сортирует данные таблицы по возрастанию объемов планет.

Задание 136

Составьте программу, которая запишет информацию о зарплате отдела (20 человек) за каждый месяц года.

Сделайте два варианта задачи: а) перекачку данных из текстового файла в двоичный; б) ввод данных с экрана и запись в двоичный файл.

Задание 137

В двумерном массиве хранится информация о зарплате отдела (20 человек) за каждый месяц года (первого человека – в первой строке, второго – во второй и т.д.). Составить программу для расчета общей зарплаты, полученной за год каждым из сотрудников, а также общую сумму, полученную всем отделом.

Задачи для самостоятельной работы

1. В двумерном массиве хранится информация о зарплате отдела (20 человек) за каждый месяц года (первого человека – в первой строке, второго – во второй и т.д.). Составить программу, в которой вычисляется:
 - а) средняя зарплата отдела за заданный месяц;
 - б) средняя зарплата конкретного сотрудника за год.
2. В двумерном массиве строк хранится информация о каждом из 20 учеников класса по тому или иному предмету (в первой строке – информация об оценках первого ученика, во второй – второго и т.д.). Общее число предметов равно 10. Определить среднюю оценку по классу в целом.
3. Дан двумерный массив целых чисел. Определить:
 - а) сумму четных элементов массива;
 - б) количество элементов массива, меньших 50;
 - в) среднее арифметическое нечетных элементов массива;
 - г) сумму тех элементов массива, сумма индексов которых кратно трем.
4. В двумерном массиве хранится информация о среднемесячной температуре в 10 регионах страны (в первой строке – температура помесечная в первой территории, во второй строке – на второй территории и т.д.). Составьте программу, которая позволит определить:
 - а) территорию и месяц с самой холодной температурой;
 - б) среднегодовую температуру по каждой из территорий;
 - в) среднюю температуру по каждому месяцу на всех территориях.

 *Пригласите преподавателя и продемонстрируйте ему сделанные Вами программы. Будьте готовы ответить на вопросы преподавателя.*

Лабораторная работа 22.

ОСНОВЫ РАБОТЫ С ДВОЧНЫМИ ФАЙЛАМИ

Задание 138

В электронном пособии (Пособия\Хранение и обработка данных.pdf) повторите материал по теме «Двоичные файлы» (с. 10-11, 23-25, 40-41).

Задание 139

Запустите среду Dev-C++.

Составьте программу, которая записывает в файл произвольное количество натуральных чисел. Введите в файл несколько пар чисел 11 и 13 (это нужно для решения следующей задачи).

Задание 140

Составьте программу, которая читает натуральные числа из созданного файла и:

- а) выводит их на экран;
- б) считает среднее арифметическое этих чисел (если вы правильно решали предыдущую задачу, то ответ должен быть 12);
- в) определяла максимальное и минимальное этих чисел;
- г) определяла количество минимальных чисел в файле.

Задание 141

Составьте программу, которая создает сразу два файла. Первый файл (назовем его `fam.dat`) должен содержать название товара, второй (`costs.dat`) - его стоимость. Ввод данных с экрана должен осуществляться последовательно: название товара, его стоимость, название товара, его стоимость...

Задание 142

Пусть дан файл `fam.dat`, содержащий название товара, и `costs.dat`, содержащий цены этих товаров. Составить программу, которая определяет:

- а) название самого дорогого товара;
- б) название самого дешевого товара;
- в) среднюю стоимость товаров.

Определите размеры файлов в байтах. Какой из файлов будет занимать больше места на диске?

Задачи для самостоятельной работы

1. Дан файл, содержащий целые числа (положительные и отрицательные). Составьте программу, которая создаст два файла, в одном из которых будут только положительные числа из первого файла, а во втором – только отрицательные.

2. Имеются два файла, элементами которых являются числа. Составить программу, которая создаст третий файл, в котором записаны сначала числа из первого файла, затем – из второго.

3.

4.

☞ *Пригласите преподавателя и продемонстрируйте ему сделанные Вами программы. Будьте готовы ответить на вопросы преподавателя.*

☞ *Если вы обучаетесь в условиях использования дистанционных образовательных технологий, вышлите все файлы с подготовленными Вами программами преподавателю.*