



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение высшего образования  
«Дальневосточный федеральный университет»  
(ДФУ)

ШКОЛА ЕСТЕСТВЕННЫХ НАУК

«СОГЛАСОВАНО»  
Руководитель ОП

И.Л. Артемьева

«УТВЕРЖДАЮ»

Заведующая кафедрой прикладной математики, механики, управления и программного обеспечения

И.Л. Артемьева

« 10 » июля 2019 г.



**РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ**

Компьютерный практикум

**Направление подготовки – 02.03.03 «Математическое обеспечение и администрирование информационных систем»**

(Технология программирования)

**Форма подготовки (очная)**

курс 2 семестр 3

лекции 00 час.

практические занятия 00 час.

лабораторные работы 72 час.

в том числе с использованием МАО лек. 0 /пр. 0 /лаб. 72 час.

всего часов аудиторной нагрузки 72 час.

в том числе с использованием МАО 72 час.

самостоятельная работа 72 час.

в том числе на подготовку к экзамену 00 час.

контрольные работы (количество) не предусмотрены

курсовая работа / курсовой проект не предусмотрены

зачет 3 семестр

экзамен не предусмотрен

Рабочая программа составлена в соответствии с требованиями Федерального государственного образовательного стандарта высшего образования, утвержденного приказом Министерства образования и науки Российской Федерации от 23 августа 2017 г. № 809

Рабочая программа обсуждена на заседании кафедры прикладной математики, механики, управления и программного обеспечения, протокол № 7.1 от «04» июля 2019 г.

Заведующий кафедрой прикладной математики, механики, управления и программного обеспечения д.т.н., профессор Артемьева И.Л.

Составители: заведующая кафедрой прикладной математики, механики, управления и программного обеспечения Артемьева И.Л., д.т.н., профессор,

доцент кафедры прикладной математики, механики, управления и программного обеспечения Остроухова С.Н., к.т.н.

**Владивосток  
2019**

**Оборотная сторона титульного листа РПУД**

**I. Рабочая программа пересмотрена на заседании кафедры:**

Протокол от « \_\_\_\_\_ » \_\_\_\_\_ 20\_\_ г. № \_\_\_\_\_

Заведующий кафедрой \_\_\_\_\_  
(подпись) (И.О. Фамилия)

**II. Рабочая программа пересмотрена на заседании кафедры:**

Протокол от « \_\_\_\_\_ » \_\_\_\_\_ 20\_\_ г. № \_\_\_\_\_

Заведующий кафедрой \_\_\_\_\_  
(подпись) (И.О. Фамилия)

**III. Рабочая программа пересмотрена на заседании кафедры:**

Протокол от « \_\_\_\_\_ » \_\_\_\_\_ 20\_\_ г. № \_\_\_\_\_

Заведующий кафедрой \_\_\_\_\_  
(подпись) (И.О. Фамилия)

**IV. Рабочая программа пересмотрена на заседании кафедры:**

Протокол от « \_\_\_\_\_ » \_\_\_\_\_ 20\_\_ г. № \_\_\_\_\_

Заведующий кафедрой \_\_\_\_\_  
(подпись) (И.О. Фамилия)

## ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

**Цель** дисциплины – усвоение и закрепление основных приемов, методов и принципов работы при решении задач на ЭВМ с использованием языков высокого уровня (С, С++).

### Задачи дисциплины:

1. Изучить синтаксис и семантику языка С и основы языка С++
2. Научиться грамотно использовать язык С и С++ при решении простых задач.
3. Научиться записывать алгоритмы с использованием языков С и С++.

Планируемые результаты обучения по данной дисциплине (знания, умения, владения), соотнесенные с планируемыми результатами освоения образовательной программы.

Общепрофессиональные компетенции выпускников и индикаторы их достижения:

Наименование категории (группы) общепрофессиональных компетенций	Код и наименование общепрофессиональной компетенции	Код и наименование индикатора достижения общепрофессиональной компетенции
Информационно-коммуникационные технологии для профессиональной деятельности	ОПК-3. Способен применять современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения	ОПК-3.1. Знает основные положения и концепции в прикладного системного программирования, архитектуры компьютеров и сетей (в том числе и глобальных), современные языки программирования, технологии создания и эксплуатации программных продуктов и программных комплексов. ОПК-3.2. Умеет использовать их в профессиональной деятельности, ОПК-3.3. Имеет практические навыки разработки программного

Информационно-коммуникационные технологии для профессиональной деятельности	ОПК-4 Способен участвовать в разработке технической документации программных продуктов и программных комплексов	ОПК-4.1. Знает основные стандарты, нормы и правила разработки технической документации программных продуктов и программных комплексов. ОПК-4.2. Умеет использовать их при подготовке технической документации программных продуктов. ОПК-4.3. Имеет практические навыки подготовки технической документации
Информационно-коммуникационные технологии для профессиональной деятельности	ОПК-5. Способен устанавливать и сопровождать программное обеспечение для информационных систем и баз данных, в том числе отечественного производства	ОПК-5.1. Знает методику установки и администрирования информационных систем и баз данных. ОПК-5.2. Умеет реализовывать техническое сопровождение информационных систем и баз данных. ОПК-5.3. Имеет практические навыки установки и инсталляции программных комплексов

Профессиональные компетенции выпускников и индикаторы их достижения:

Задача профессиональной деятельности	Объект или область знания	Код и наименование профессиональной компетенции	Код и наименование индикатора достижения профессиональной компетенции	Основание (ПС, анализ иных требований, предъявляемых к выпускникам
Тип задач профессиональной деятельности: производственно-технологический				

<p>Разработка, отладка, проверка работоспособности, модификация программного обеспечения. Создание и сопровождение архитектуры программных средств. Разработка и тестирование программного обеспечения. Проектирование, разработка и сопровождение компьютерных систем автоматизации и производства и управления</p>	<p>Математические и алгоритмические модели, программы, программные системы и комплексы, методы их проектирования и реализации, способы производства, сопровождения, эксплуатации и администрирования в различных областях цифровой экономики.</p>	<p>ПК-8. Способен использовать основные концептуальные положения функционального, логического, объектно-ориентированного и визуального направлений программирования, методы, способы и средства разработки программ в рамках этих направлений.</p>	<p>ПК-8.1. Знает основные концептуальные положения функционального, логического, объектно-ориентированного и визуального направлений программирования, методы, способы и средства разработки программ в рамках этих направлений.  ПК-8.2. Умеет программировать в рамках этих направлений.  ПК-8.3. Имеет практический опыт разработки программ в рамках этих направлений.</p>	<p>06.003 Архитектор программного обеспечения;  06.015 Специалист по информационным системам;  06.016 Руководитель проектов в области информационных технологий;  06.019 Технический писатель (специалист по технической документации в области информационных технологий);  06.022 Системный аналитик;  40.011 Специалист по научно-исследовательским и опытно-конструкторским разработкам;  40.057 Специалист по автоматизированным системам управления производством;  06.004 Специалист по тестированию в области информационных технологий;  06.011 Администратор баз данных;  06.001 Программист;</p>
--	---	--	--	---

Для формирования вышеуказанных компетенций в рамках дисциплины «Компьютерный практикум» применяются следующие методы активного/интерактивного обучения: метод проектов.

## **I. СТРУКТУРА И СОДЕРЖАНИЕ ТЕОРЕТИЧЕСКОЙ ЧАСТИ КУРСА**

Лекции не предусмотрены учебным планом.

## **II. СТРУКТУРА И СОДЕРЖАНИЕ ПРАКТИЧЕСКОЙ ЧАСТИ КУРСА**

### **Лабораторные работы (72 час.)**

#### **Лабораторная работа 1. Основы языка C++. Операторы. (10 часов)**

Получение практических навыков работы с операторами, потоками ввода-вывода

#### **Лабораторная работа 2. Работа с данными. Типы данных C++. (10 часов)**

Изучить встроенные типы данных языка C++, арифметические операции. Работа с константами.

#### **Лабораторная работа 3. Составные типы данных. Массивы. Структуры. (8 часов)**

Изучить составные типы данных языка C++: структуры, массивы, строки. Освоить практические навыки работы с ними. Указатели.

#### **Лабораторная работа 4. Циклы. Выражения отношений. (16 часов)**

Получение практических навыков работы с циклами (for, while do, do while). Работа с вложенными циклами, двумерными массивами.

#### **Лабораторная работа 5. Операторы ветвления. Логические операторы. (14 часов)**

Получение практических навыков работы с оператором if then else, логическими операторами, операторами switch, а также break и continue. Работа с файлами.

### Лабораторная работа 6. Функции. Рекурсия. (14 часов)

Получение практических навыков работы в области написания собственных пользовательских функций а также рекурсивных функций

## Ш. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ

Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине «Компьютерный практикум» представлено в разделе VIII и включает в себя: план-график выполнения самостоятельной работы по дисциплине, в том числе примерные нормы времени на выполнение по каждому заданию; характеристика заданий для самостоятельной работы обучающихся и методические рекомендации по их выполнению; требования к представлению и оформлению результатов самостоятельной работы; критерии оценки выполнения самостоятельной работы.

## IV. КОНТРОЛЬ ДОСТИЖЕНИЯ ЦЕЛЕЙ КУРСА

№ п/п	Контролируемые разделы/темы дисциплины	Коды и этапы формирования компетенций	Оценочные средства - наименование		
			текущий контроль	промежуточная аттестация	
1	Основы языка C++. Операторы.	ОПК-3 ОПК-4 ОПК-5 ПК-8	Знает Умеет Владеет	УО1 собеседование ПР2 контрольная работа ПР-6 лабораторная работа 1	Зачет, вопросы 1-3
2	Работа с данными. Типы данных C++.	ОПК-3 ОПК-4 ОПК-5 ПК-8	Знает Умеет Владеет	УО1 собеседование ПР2 контрольная работа ПР-6 лабораторная работа 2	Зачет, вопросы 4
3	Составные типы данных. Массивы. Структуры.	ОПК-3 ОПК-4 ОПК-5 ПК-8	Знает Умеет Владеет	УО1 собеседование ПР2 контрольная работа ПР-6 лабораторная работа 3	Зачет, вопросы 9-12

4	Циклы. Выражения отношений.	ОПК-3 ОПК-4 ОПК-5 ПК-8	Знает Умеет Владеет	УО1 ПР2 работа ПР-6 лабораторная работа 4	собеседование контрольная лабораторная	Зачет, вопросы 6
5	Операторы ветвления. Логические операторы.	ОПК-3 ОПК-4 ОПК-5 ПК-8	Знает Умеет Владеет	УО1 ПР2 работа ПР-6 лабораторная работа 5	собеседование контрольная лабораторная	Зачет, вопросы 5,7
6	Функции. Рекурсия.	ОПК-3 ОПК-4 ОПК-5 ПК-8	Знает Умеет Владеет	УО1 ПР2 работа ПР-6 лабораторная работа 6	собеседование контрольная лабораторная	Зачет, вопросы 8, 13-17

Типовые контрольные задания, методические материалы, определяющие процедуры оценивания знаний, умений и навыков и (или) опыта деятельности, а также критерии и показатели, необходимые для оценки знаний, умений, навыков и характеризующие этапы формирования компетенций в процессе освоения образовательной программы, представлены в разделе IX.

## **V. СПИСОК УЧЕБНОЙ ЛИТЕРАТУРЫ И ИНФОРМАЦИОННО МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

### **Основная литература**

1. С/С++. Структурное и объектно-ориентированное программирование: практикум / Т.А. Павловская, Ю.А. Щупак.- СПб.: Питер, 2010.- 347 с.  
<http://lib.dvfu.ru:8080/lib/item?id=chamo:418970&theme=FEFU>
2. Керниган Б., Ритчи Д. Язык программирования С. Пер. с англ. — М.: Издательский дом "Вильямс", 2012. — 289 с.  
<http://lib.dvfu.ru:8080/lib/item?id=chamo:666721&theme=FEFU>
3. Подбельский В.В., Фомин С.С. Программирование на языке Си: Учеб. пособие для вузов. М.: Финансы и статистика, 2009. - 600 с..  
<http://lib.dvfu.ru:8080/lib/item?id=chamo:355876&theme=FEFU>
4. Язык программирования С++. Специальное издание: пер. с англ. / Бьерн Страуструп. - М.: Вильямс, 2011. — 1135с.  
<https://lib.dvfu.ru:8443/lib/item?id=chamo:418460&theme=FEFU>



5. C/C++. Программирование на языке высокого уровня: учебник для вузов для магистров и бакалавров / Т. А. Павловская. СПб.: Питер, 2011.-460 с. <https://lib.dvfu.ru:8443/lib/item?id=chamo:307692&theme=FEFU>
6. Программирование на языке C++: Учебное пособие / Т.И. Немцова, С.Ю. Голова, А.И. Терентьев; Под ред. Л.Г. Гагариной. - М.: ИД ФОРУМ: ИНФРА-М, 2012. - 512 с. <http://znanium.com/catalog.php?bookinfo=244875>
7. Дейл, Н. Программирование на C++ [Электронный ресурс] / Н. Дейл, Ч. Уимз, М. Хедингтон; Пер. с англ. - М.: ДМК Пресс, 2007. - 672 с. <http://znanium.com/catalog.php?bookinfo=407353>
8. Программирование на языке Си/ А.В.Кузин, Е.В.Чумакова - М.: Форум, НИЦ ИНФРА-М, 2015. - 144 с. <http://znanium.com/catalog.php?bookinfo=505194>
9. Царев, Р. Ю. Программирование на языке Си [Электронный ресурс] : учеб. пособие / Р. Ю. Царев. – Красноярск : Сиб. федер. ун-т, 2014. – 108 с. <http://znanium.com/catalog.php?bookinfo=510946>

#### Дополнительная литература

1. Полубенцева, М.И. C/C++. Процедурное программирование [Электронный ресурс] / М.И. Полубенцева. - СПб.: БХВ-Петербург, 2008. - 414 с.: ил. - (Внесерийная). - ISBN 978-5-9775-0145-3. <http://znanium.com/catalog.php?bookinfo=350407>
2. Дизайн и эволюция C++ : [пер. с англ.] / Бьерн Страуструп. Москва : ДМК Пресс, Санкт-Петербург : Питер, 2007. - 444 с. <https://lib.dvfu.ru:8443/lib/item?id=chamo:297090&theme=FEFU>
3. Павловская Т.А. C/C++. Программирование на языке высокого уровня- СПб.: Питер, 2006.-461 с. Режим доступа: <http://lib.dvfu.ru:8080/lib/item?id=chamo:237589&theme=FEFU>
4. Хабибуллин И.Ш. Программирование на языке высокого уровня C/C++. - СПб.: БХВ-Петербург, 2006. - 499 с. - ISBN 5-94157-559-9. <http://znanium.com/catalog.php?bookinfo=356906>
5. Романов Е.Л. Практикум по программированию на C++: уч. пособие. СПб: БХВ-Петербург; Новосибирск: Изд-во НГТУ, 2004. - 432 с.
6. Лукас П. C++ под рукой: Пер. с англ. - Киев: «ДиаСлфт», 1993. <http://lib.dvfu.ru:8080/lib/item?id=chamo:378744&theme=FEFU>
7. Холзнер С. Visual C++ 6. Учебный курс.- СПб.: Питер, 2007. — 570 с: ил.
8. Прата Стивен. Язык программирования C++ .- Лекции и упражнения (5-е изд.), М. : ООО "И.Д. Вильямс", 2007. 1171 с. - ISBN: 5-8459-1127-3, 0-672-32697-3.

9. Шилдт Г. Полный справочник по С++ . -ИД "Вильямс", 4-е изд, 2006 г, 791с.

### **Перечень ресурсов информационно-телекоммуникационной сети «Интернет»**

1. <https://code-live.ru/tag/cpp-manual/> Портал о программировании
2. <http://cppstudio.com/cat/274/> Язык программирования С++
3. <http://www.c-cpp.ru/> Программирование на С и С++. Онлайн справочник программиста
4. <http://www.proklondike.com/books/cpp.html> Программирование
5. <http://purecodecpp.com/> основы программирования на С++ для начинающих
6. <http://progbook.ru/c/> Книги по С++
7. <http://mexalib.com/view/2267> Компьютерная литература. Шилдт Г. Самоучитель С++ ВHV - Санкт - Петербург, 2003. - 688 стр.
8. <http://www.programmersclub.ru/main/> Клуб программистов
9. <http://info-comp.ru/> Информационный портал. Все о компьютере и программировании для начинающих

### **Перечень информационных технологий и программного обеспечения**

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используется следующее программное обеспечение:

1. Пакет прикладных программ Microsoft Office / Open Office.
2. Интегрированные среды разработки программ на языке С++.
3. Программное обеспечение электронного ресурса сайта ДВФУ, включая ЭБС ДВФУ.

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используются следующие информационно-справочные системы:

1. Научная электронная библиотека eLIBRARY.
2. Электронно-библиотечная система IPRbooks.
3. Доступ к электронному заказу книг в библиотеке ДВФУ, доступ к нормативным документам ДВФУ, расписанию, рассылке писем.

Лабораторные занятия проводятся в специализированном компьютерном классе.

## **VI. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ**

Дисциплина «Компьютерный практикум» изучается в следующих организационных формах: лабораторное занятие; самостоятельное изучение теоретического материала; индивидуальные и групповые консультации.

### **Лабораторные работы**

В результате выполнения лабораторных работ студент должен научиться формализовать поставленную задачу, составлять и оформлять программы на языке программирования C++, тестировать и отлаживать программы, использовать для разработки программ современные интегрированные среды разработки программ.

При выполнении лабораторной работы необходимо следовать методическим рекомендациям по ее выполнению. Результатом лабораторной работы является программа и отчет (документация), которые демонстрируются преподавателю в конце работы. Студент должен уметь отвечать на вопросы преподавателя, поясняя процесс создания программ и документа и выполнения работы.

Индивидуальные задания необходимо выполнять согласно рассмотренной технологии:

1. изучить словесную постановку задачи;
2. сформулировать математическую постановку задачи;
3. выбрать метод решения задачи, если это необходимо;
4. разработать схему алгоритма и спроектировать требуемые классы;
5. записать разработанный алгоритм на языке C++;
6. разработать контрольные тесты программы;
7. отладить программу;
8. написать отчет.

### **Содержание отчета**

1. Титульный лист.
2. Неформальная постановка задачи (НПЗ).
3. Формальная постановка задачи (ФПЗ).
4. Проекты классов
5. Текст программы с комментариями.
6. Тестовые условия.

### **Работа с литературой**

В процессе подготовки к лабораторным работам, студентам необходимо обратить особое внимание на поиск и на самостоятельное

изучение рекомендованной учебно-методической (а также научной и популярной) литературы. Самостоятельная работа с учебниками, учебными пособиями, научной, справочной и популярной литературой, материалами периодических изданий и Интернета, статистическими данными является наиболее эффективным методом получения знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у студентов свое отношение к конкретной проблеме. Более глубокому раскрытию вопросов способствует знакомство с дополнительной литературой, рекомендованной преподавателем по каждой теме практического занятия, что позволяет студентам проявить свою индивидуальность в рамках выполнения индивидуального проекта, выявить широкий спектр мнений по изучаемой проблеме.

### **Самостоятельная работа студента**

Основными формами самостоятельной работы студента являются:

- подготовка к лабораторным занятиям, зачету,
- изучение обязательной и дополнительной литературы,
- поиск информации по изучаемым темам в периодических изданиях и Интернете,
- изучение в рамках программы курса тем, не выносимых на лабораторные работы,
- оформление отчетов по лабораторным работам.

Контроль за выполнением работы студента производится в виде контроля каждого этапа работы (см. приложение 1).

Студент должен планировать график самостоятельной работы по дисциплине и придерживаться его.

## **VII. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

Лекции проводятся с использованием проектора и внутренней системы портала ДВФУ. Лабораторные занятия проходят в аудиториях, оборудованных компьютерами типа Lenovo C360G-i34164G500UDK с лицензионными программами MicrosoftOffice 2013 и аудио-визуальными средствами проектор Panasonic DLPProjectorPT-D2110XE, плазма LG FLATRON M4716CCBAM4716CJ. Для выполнения самостоятельной работы студенты в жилых корпусах ДВФУ обеспечены Wi-Fi.

## **VIII. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ**

Самостоятельная работа состоит в выполнении на компьютере индивидуальных практических заданий по изучаемым темам (72 часа).

### План-график выполнения самостоятельной работы по дисциплине

№ п/п	Дата/сроки выполнения	Виды СРС	Всего часов	Форма контроля
1.	1-2 неделя обучения	Изучение теоретического материала к ЛР-1 по литературным источникам	4	Собеседование
		Оформление отчета по ЛР-1	8	Проверка отчета
2.	3-4 неделя обучения	Изучение теоретического материала к ЛР-2 по литературным источникам	4	Собеседование
		Оформление отчета по ЛР-2	8	Проверка отчета т
3.	5-6 неделя обучения	Изучение теоретического материала к ЛР-3 по литературным источникам	4	Собеседование
		Оформление отчета по ЛР-3	8	Проверка отчета
4.	7-12 неделя обучения	Изучение теоретического материала к ЛР-4 по литературным источникам	4	Собеседование
		Оформление отчета ЛР-4	8	Проверка отчета
5.	13-15 неделя обучения	Изучение теоретического материала к ЛР-5 по литературным источникам	4	Собеседование
		Оформление отчета ЛР-6	8	Проверка отчета
6.	16-18 неделя обучения	Изучение теоретического материала к ЛР-6 по литературным источникам	4	Собеседование
		Оформление отчета ЛР-6	8	Проверка отчета
Итого:			72	

### Рекомендации по работе с литературой

Для более эффективного освоения и усвоения материала рекомендуется ознакомиться с теоретическим материалом по той или иной теме до проведения лабораторного занятия. Всю учебную литературу желательно изучать «под конспект».

Цель написания конспекта по дисциплине – сформировать навыки по поиску, отбору, анализу и формулированию учебного материала.

Работу с теоретическим материалом по теме можно проводить по следующей схеме:

- название темы;

- цели и задачи изучения темы;
- основные вопросы темы;
- характеристика основных понятий и определений, необходимых для усвоения данной темы;
- краткие выводы, ориентирующие на определенную совокупность сведений, основных идей, ключевых положений, систему доказательств, которые необходимо усвоить.

При работе над конспектом обязательно выявляются и отмечаются трудные для самостоятельного изучения вопросы, с которыми уместно обратиться к преподавателю при посещении консультаций, либо в индивидуальном порядке.

### **Подготовка к лабораторным работам**

Подготовку к каждой лабораторной работе каждый студент должен начать с изучения теоретического материала и ознакомления с планом, который отражает содержание предложенной темы. Все новые понятия по изучаемой теме необходимо выучить наизусть и внести в глоссарий, который целесообразно вести с самого начала изучения курса. Результат такой работы должен проявиться в способности студента свободно ответить на теоретические вопросы по теме задания, правильном выполнении лабораторной работы.

В процессе выполнения лабораторной работы студент должен создать требуемый документ (отчет) с помощью предлагаемого программного средства и выполнить требуемые в задании операции. Задание по лабораторной работе содержит методические указания по подготовке документа, который должен быть получен в результате выполнения работы. При подготовке к лабораторной работе следует их внимательно прочесть.

При формировании индивидуальных заданий по самостоятельной работе используется материал из работы: Программирование: метод. указания/ сост. Л.И.Прудникова - Владивосток: Изд-во ДВФУ, 2012. - 24 с.

### **Содержание отчета по лабораторной работе**

1. Титульный лист.
2. Неформальная постановка задачи (НПЗ).
3. Формальная постановка задачи (ФПЗ).
4. Блок-схема.
5. Текст программы с комментариями.
6. Тестовые условия.

### **Критерии оценки отчетов по лабораторным работам (проектов)**

– 100-86 баллов выставляется, если содержание и составляющие части соответствуют выданному заданию. Продемонстрировано владение навыками подготовки документа по теме. Фактических ошибок, связанных с пониманием структуры и содержания задания нет.

– 85-76 - баллов выставляется, если при выполнении задания допущено не более одной ошибки. Продемонстрировано владение навыками подготовки документа по теме. Фактических ошибок, связанных с пониманием структуры и содержания задания нет.

– 75-61 балл выставляется, если при выполнении задания допущено не более двух ошибок. Продемонстрировано навыки подготовки документа по теме. Допущено не более 2 ошибок, связанных с пониманием структуры и содержания задания.

60-50 баллов - если структура и содержание задания не соответствуют требуемым

### **Шкала оценивания**

Менее 60 баллов	незачтено	неудовлетворительно
От 61 до 75 баллов	зачтено	удовлетворительно
От 76 до 85 баллов	зачтено	хорошо
От 86 до 100 баллов	зачтено	отлично

## IX. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

### Паспорт ФОС

Код и формулировка компетенции	Этапы формирования компетенции	
ОПК-3. Способен применять современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения	Знает	основные положения и концепции в прикладного и системного программирования, архитектуры компьютеров и сетей (в том числе и глобальных), современные языки программирования, технологии создания и эксплуатации программных продуктов и программных комплексов
	Умеет	использовать их в профессиональной деятельности
	Владеет	Имеет практические навыки разработки программного обеспечения
ОПК-4. Способен участвовать в разработке технической документации программных продуктов и программных комплексов	Знает	основные стандарты, нормы и правила разработки технической документации программных продуктов и программных комплексов
	Умеет	использовать их при подготовке технической документации программных продуктов
	Владеет	Имеет практические навыки подготовки технической документации
ОПК-5. Способен устанавливать и сопровождать программное обеспечение для информационных систем и баз данных, в том числе отечественного производства	Знает	методику установки и администрирования информационных систем и баз данных.
	Умеет	реализовывать техническое сопровождение информационных систем и



		баз данных
	Владеет	Имеет практические навыки установки и инсталляции программных комплексов
ПК-8. Способен использовать основные концептуальные положения функционального, логического, объектно- ориентированного и визуального направлений программирования, методы, способы и средства разработки программ в рамках этих направлений.	Знает	основные концептуальные положения функционального, логического, объектно- ориентированного и визуального направлений программирования, методы, способы и средства разработки программ в рамках этих направлений
	Умеет	программировать в рамках этих направлений
	Владеет	Имеет практический опыт разработки программ в рамках этих направлений

№ п/п	Контролируемые разделы/темы дисциплины	Коды и этапы формирования компетенций	Оценочные средства - наименование		
			текущий контроль	промежуточная аттестация	
1	Основы языка C++. Операторы.	ОПК-3 ОПК-4 ОПК-5 ПК-8	Знает Умеет Владеет	УО1 собеседование ПР2 контрольная работа ПР-6 лабораторная работа 1	Зачет, вопросы 1-3
2	Работа с данными. Типы данных C++.	ОПК-3 ОПК-4 ОПК-5 ПК-8	Знает Умеет Владеет	УО1 собеседование ПР2 контрольная работа ПР-6 лабораторная работа 2	Зачет, вопросы 4
3	Составные типы данных. Массивы. Структуры.	ОПК-3 ОПК-4 ОПК-5 ПК-8	Знает Умеет Владеет	УО1 собеседование ПР2 контрольная работа ПР-6 лабораторная	Зачет, вопросы 9-12

				работа 3	
4	Циклы. Выражения отношений.	ОПК-3 ОПК-4 ОПК-5 ПК-8	Знает Умеет Владеет	УО1 собеседование ПР2 контрольная работа ПР-6 лабораторная работа 4	Зачет, вопросы 6
5	Операторы ветвления. Логические операторы.	ОПК-3 ОПК-4 ОПК-5 ПК-8	Знает Умеет Владеет	УО1 собеседование ПР2 контрольная работа ПР-6 лабораторная работа 5	Зачет, вопросы 5,7
6	Функции. Рекурсия.	ОПК-3 ОПК-4 ОПК-5 ПК-8	Знает Умеет Владеет	УО1 собеседование ПР2 контрольная работа ПР-6 лабораторная работа 6	Зачет, вопросы 8, 13-17

### Шкала оценивания уровня сформированности компетенций

Код и формулировка компетенции	Этапы формирования компетенции		критерии	показатели
ОПК-3. Способен применять современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения	знает (пороговый уровень)	Синтаксис и семантику основных конструкций алгоритмических языков С и С++	Знание синтаксиса и семантики языков С и С++	Способность дать ответы на вопросы
	умеет (продвинутый)	использовать методы записи алгоритмов на языках С и С++	Умение записать алгоритм с использованием языков С и С++	Наличие созданных при выполнении лабораторных работ программ
	владеет (высокий)	методами, способами и программными средствами для разработки программ на языках С и С++	Владение способами программирования в среде языковых процессоров С и С++	Наличие компьютерных программ для языкового процессора/, созданных при выполнении лабораторных работ

ПК-8. Способен использовать основные концептуальные положения функционального, логического, объектно-ориентированного и визуального направлений программирования, методы, способы и средства разработки программ в рамках этих направлений.	знает (пороговый уровень)	Методы проектирования программ на языках С и С++	Знание методов проектирования структур данных и процедур	Способность дать ответы на вопросы
	умеет (продвинутый)	Проектировать требуемые структуры данных и процедуры при создании программ на языках С и С++	Умение использовать методы проектирования программ на языках С и С++	Наличие созданных проектов
	владеет (высокий)	Навыками разработки программных средств по проекту	Владение методами разработки программ в некоторой среде программирования	Наличие созданных программ
ОПК-5. Способен устанавливать и сопровождать программное обеспечение для информационных систем и баз данных, в том числе отечественного производства	знает (пороговый уровень)	Методы оценки качества проекта	Знание методов проектирования тестовых ситуаций и тестов	Способность дать ответы на вопросы
	умеет (продвинутый)	Проектировать требуемый набор тестов для оценки качества программ	Умение определить тестовые ситуации и необходимый набор тестов для проверки качества создаваемых программ	Наличие проектов тестов
	владеет (высокий)	Навыками создания пакета тестов для оценки качества программ	Умение записать тестовые данные средствами языка	Наличие созданных тестов

ОПК-4 Способен участвовать в разработке технической документации программных продуктов и программных комплексов	знает (пороговый уровень)	Типы данных и операции с ними в языках С и С++	Знание стандартных типов данных языков С и С++, операций работы с ними	Способность дать ответы на вопросы
	умеет (продвинутый)	использовать стандартные типы данных, конструкторы типов и операции работы с данными разных типов при создании программ на языках С и С++	Умение проектировать программы, используя стандартные типы данных или определяя требуемые	Наличие проектов
	владеет (высокий)	методами проектирования и разработки программ, требующих использования разных структур данных	Владение методами создания программ на языках С и С++	Наличие созданных программ

### Методические рекомендации, определяющие процедуры оценивания результатов освоения дисциплины

**Промежуточная аттестация студентов.** Промежуточная аттестация студентов проводится в соответствии с локальными нормативными актами ДВФУ, и является обязательной.

По дисциплине предусмотрен зачет, который проводится в устной либо письменной форме.

### Оценочные средства для промежуточной аттестации

#### ЗАДАЧИ К ЗАЧЕТУ

1. Какие из следующих последовательностей символов не являются идентификаторами? Почему?

$x$	$x_1$	$X1$	$_1$
$1x2x3$	$\sin$	$\sin x$	$A_1$
$\sin(x)$	$A-1$	$2a$	$mass$

2. Ниже приводятся выражения, записанные в традиционной математической форме, справа от которых даны некоторые выражения, записанные по правилам языка Паскаля. Расстановкой минимального количества скобок добиться того, чтобы выражения слева и справа стали эквивалентны.

a.  $a/b*c/d*e/f*h$

$$\frac{a}{b * \frac{c}{a+b/x-2*y/f*h}}$$

b.  $b) \frac{a+b}{x-2y}$

c.  $c) a + \frac{b}{x-2} * y$

$$a+b/x-2*y$$

3. Какие из следующих выражений не содержат ошибки? Почему?

a)  $-xy^2$

b)  $1+|y|$

c)  $_2xy$

d)  $-2*xy$

e)  $-abs(x)+siny$

f)  $x+y*-z$

g)  $0,5(x+7)*(ax-b+1)/3$

4. Переписать следующие выражения, записанные по правилам языка Паскаль, в традиционной математической форме:

a)  $sqrt(a+b)-sqrt(a-b)$

b)  $a+b/(c+d)-(a+b)/c+d$

c)  $a*b/(c+d)-(c-d)/b*(a+b)$

d)  $1+sqrt(cos((x+y)/2))$

5. Какие числа и переменные содержатся в следующих выражениях, записанных по правилам языка Паскаль?

a)  $2+x-y-1.7$

b)  $2*xy$

c)  $sin(2+a)-1/3$

d)  $1/2+1/3-0.2/x-sqrt(-y)+yx_y$

6. Записать в соответствии с правилами языка Паскаль следующие выражения:

a)  $\frac{x^2 + y^2}{1 - \frac{x^2 - y^2}{2}}$

b)  $\frac{1 - \sqrt{1 + |\sin(x)|}}{2}$

c)  $\frac{a+b}{c+d} - 2,5 : \frac{c+a}{a+b:c}$

7. Какие из следующих последовательностей символов являются операторами, записанными по правилам языка Паскаль? Почему?

a.  $a:=b;$

b.  $a:=2c+1;$

c.  $a*x+b:=0;$

d.  $z:=z+1,2;$

e.  $z:=z+1;$

f.  $if\ x>y\ then\ x:=0;\ y:=0$   
 $else\ write(z);$

g.  $if\ a<b\ then\ 100\ else\ z:=5;$

h.  $if\ x<y\ then\ z:=z+1;\ else\ z:=5;$

i.  $if\ x<y\ then\ x:=0\ else\ xy:=a+b;$

8. Убрать из записи выражения лишние скобки (скобка считается лишней, если при ее удалении значение выражения не изменяется)

a)  $(x/x^2)*y$

b)  $(\sqrt{p}*q)/r$

c)  $b+(a-(c/3))$

d)  $(a*(b/(c*(d/(1+f))))))$

e)  $(a \text{ or } b) \text{ or } ((c \text{ and } b) \text{ and } a)$

9. Укажите, какие из следующих операторов содержат ошибки. Почему?

a) *while True do Read(x);*

b) *for I:=10 to 2 do X:=Sqrt(X);*

c) *while Z do begin Z:=(not X)>0;X:=X+1 end;*

d) *for X:=N downto N-2 do Read(X);*

e) *repeat Z:=X=Y; X:=X+1 until Z;*

f) *for X:=0.1 to 2 do Write(X);*

g) *while X<1 while Z<>0 do Read(X, Z);*

h) *for X:=1 to N/2 do N:=X+1;*

10. Проанализируйте завершаемость следующих циклов:

11. *Repeat X:=X-1 until X=0;*

12. *While X mod 10 <> x div 10 do X:= X + 1;*

13. *Repeat Z:=X; X:=Y; Y:=Z div 2 until X<Y;*

14. *While N>1 do if not (N mod 2=0) then N:=N div 2+1 else N:=N+1;*

15. Дано  $x$ . Вычислить  $y$ , если

$$y = \begin{cases} 0, & \text{если } x \geq 10 \\ x^{\frac{1}{2}} - x, & \text{если } 0 < x \leq 1 \\ x^2 - \sin(\pi x^2), & \text{иначе} \end{cases}$$

$$y = \begin{cases} 0, & \text{если } 5 \leq x < 15 \\ |x|, & \text{если } -1 < x \leq 0 \\ \sin(x), & \text{иначе} \end{cases}$$

$$y = \begin{cases} x^2, & \text{если } -2 \leq x \leq 2 \\ -\cos(x), & \text{если } -10 \leq x < -5 \\ 2, & \text{иначе} \end{cases}$$

$$y = \begin{cases} 0, & \text{если } x \leq -5 \\ \sin(x), & \text{если } 0 < x \leq 1 \\ x^4, & \text{иначе} \end{cases}$$

16. Дано натуральное  $n$ . Написать программу которая вычисляет сумму  $n$ -первых членов ряда:

- a)  $-\frac{1}{3^2} + \frac{1}{5^2} + \dots + \frac{(-1)^n}{(2n+1)^2}$
- b)  $-\frac{2}{1!} + \frac{3}{2!} + \dots + \frac{(-1)^n(n+1)}{n!}$
- c)  $(1 + \frac{1}{3^2})(3 + \frac{1}{5^2}) \dots (2n-1 + \frac{1}{(2n+1)^2})$
- d)  $\frac{1!}{1} + \frac{2!}{1+1/2} + \dots + \frac{n!}{1+1/2+\dots+1/n}$

17. Дана непустая последовательность натуральных чисел, за которой следует ноль. Составить программу нахождения членов последовательности, которые

- a) совпадают с последними цифрами записи их квадратов (например,  $6^2=36$ ,  $25^2=625$ )
- b) являются простыми числами
- c) являются степенями двойки
- d) можно преобразовать в число 1, повторяя не более  $n$  раз следующие действия: если число четно, то разделить его на 2, иначе умножить на 3 и прибавить 1

18. В файле содержится последовательность идентификаторов, после каждого из которых следует ровно один пробел. Написать процедуру (функцию), которая печатает каждый идентификатор на отдельной строке

- a) удаляя из него все цифры;
- b) заменяя в нем все буквы на симметричные относительно середины латинского алфавита;
- c) удаляя из него все символы;
- d) заменяя в нем все цифры на симметричные относительно середины диапазона 0..9.

19. Не используя стандартных процедур (функций) работы со строками. Написать процедуру (функцию), которая в заданной строке удаляет

- a) группы символов, расположенных между круглыми скобками, вместе с этими скобками (предполагается, что внутри скобок нет других скобок);
- b) все символы предшествующие первой точке, не являющиеся буквами;
- c) все следующие за первой запятой четные цифры;
- d) все запятые предшествующие второй точке (если двух точек нет, то все запятые).

20. Дана последовательность слов. Написать процедуру (функцию), которая печатает слова, удовлетворяющие условию:

- a) слово симметрично;
- b) буквы слова упорядочены по алфавиту;

- c) в слове нет повторяющихся букв;
- d) слово является возрастающей последовательность цифр.

### **ВОПРОСЫ К ЗАЧЕТУ**

1. Переменные, константы.
2. Выражения.
3. Приоритеты в C++.
4. Типы данных.
5. Операторы выбора.
6. Операторы цикла.
7. Операторы ветвления.
8. Функции.
9. Массивы.
10. Строки.
11. Структуры и объединения.
12. Глобальные переменные
13. Функции. Их параметры.
14. Функция main().
15. Перегрузка функций.
16. Рекурсивные функции
17. Стандартные библиотеки

**Текущая аттестация студентов.** Текущая аттестация проводится в форме собеседования (устного опроса) для проверки теоретических знаний, а также в форме защиты проекта, выполняемого в рамках самостоятельной работы параллельно с лабораторными работами и осуществляется ведущим преподавателем.

Объектами оценивания выступают:

- степень усвоения теоретических знаний - оценивается в форме собеседования и контрольных работ;
- уровень овладения практическими умениями и навыками – оценивается в форме защиты индивидуального заданий (проектов), выполняемых в рамках лабораторных работ.

Студенты получают индивидуальные задания. В процессе их выполнения должны быть разработаны: формальная постановка задачи, алгоритмы ее решения, написана программа на языке программирования C++, для которой созданы тесты. Правильность работы программы демонстрируется с помощью созданных тестов. Преподаватель вправе задать свои значения входных данных.



## Критерии оценки устного ответа

– **100-85 баллов** - если ответ показывает прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность, явлений, процессов, событий, делать выводы и обобщения, давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа; умение приводить примеры современных проблем изучаемой области.

– **85-76 баллов** - ответ, обнаруживающий прочные знания основных процессов изучаемой предметной области, отличается глубиной и полнотой раскрытия темы; владение терминологическим аппаратом; умение объяснять сущность, явлений, процессов, событий, делать выводы и обобщения, давать аргументированные ответы, приводить примеры; свободное владение монологической речью, логичность и последовательность ответа. Однако допускается одна - две неточности в ответе.

– **75-61 балл** - оценивается ответ, свидетельствующий в основном о знании процессов изучаемой предметной области, отличающийся недостаточной глубиной и полнотой раскрытия темы; знанием основных вопросов теории; слабо сформированными навыками анализа явлений, процессов, недостаточным умением давать аргументированные ответы и приводить примеры; недостаточно свободным владением монологической речью, логичностью и последовательностью ответа. Допускается несколько ошибок в содержании ответа; неумение привести пример развития ситуации, провести связь с другими аспектами изучаемой области.

– **60-50 баллов** - ответ, обнаруживающий незнание процессов изучаемой предметной области, отличающийся неглубоким раскрытием темы; незнанием основных вопросов теории, несформированными навыками анализа явлений, процессов; неумением давать аргументированные ответы, слабым владением монологической речью, отсутствием логичности и последовательности. Допускаются серьезные ошибки в содержании ответа; незнание современной проблематики изучаемой области.

## Оценочные средства для текущей аттестации

### Вопросы к собеседованию

1. Переменные, константы.

2. Выражения.
3. Приоритеты в C++.
4. Типы данных.
5. Операторы выбора.
6. Операторы цикла.
7. Операторы ветвления.
8. Функции.
9. Массивы.
10. Строки.
11. Структуры и объединения.
12. Глобальные переменные
13. Функции. Их параметры.
14. Функция main().
15. Перегрузка функций.
16. Рекурсивные функции
17. Стандартные библиотеки

### **Контрольные работы**

1. Разбить текст из входного файла input.txt на N колонок шириной K символов. В том случае, если слово не вмещается в ширину колонки, выдать соответствующее сообщение об ошибке и указать минимальную допустимую ширину колонки. Каждая из колонок в результате разбиения должна иметь выравнивание по ширине. Результат работы программы вывести в выходной файл output.txt.

2. Разбить текст из входного файла input.txt на N колонок шириной K символов. В том случае, если слово не вмещается в ширину колонки, выдать соответствующее сообщение об ошибке и указать минимальную допустимую ширину колонки. Каждая из колонок в результате разбиения должна иметь выравнивание по ширине. В качестве строкового типа можно использовать только char\*. Результат работы программы вывести в выходной файл output.txt.

3. Написать калькулятор строковых выражений. В выражениях допустимы скобки любого уровня вложенности. Множество допустимых операций: +, -, \*, /, ^. Исходное выражение считывается из входного файла и выводится в выходной файл.

4. Решить систему уравнений методом Гаусса. Исходная матрица коэффициентов считывается из входного файла, результат работы программы выводится в выходной файл.

5. Разработать эмулятор клавиатуры сотового телефона (12 кнопок). Приложение должно обладать оконным интерфейсом. Знаки препинания расположены на кнопке 1, пробел на 0.

### **Критерии оценки программы по лабораторным работам (проектов)**

– 100-86 баллов выставляется, если содержание и составляющие части соответствуют выданному заданию. Продемонстрировано владение навыками разработки, тестирования программ на языке программирования C++. Программа правильно работает на всех наборах входных данных. Текст программы содержит комментарии.

– 85-76 - баллов выставляется, если при выполнении задания допущено не более одной ошибки. Продемонстрировано владение навыками разработки программ на языке программирования C++. Программа правильно работает не на всех наборах входных данных (90%). Текст программы содержит комментарии.

– 75-61 балл выставляется, если при выполнении задания допущено не более двух ошибок. Продемонстрировано знание методов разработки программ на языке программирования C++. Программа правильно работает не на всех наборах входных данных (70%). В тексте программы комментарии отсутствуют.

60-50 баллов - если структура и содержание задания не соответствуют требуемым

### **Шкала оценивания**

Менее 60 баллов	незачтено	неудовлетворительно
От 61 до 75 баллов	зачтено	удовлетворительно
От 76 до 85 баллов	зачтено	хорошо
От 86 до 100 баллов	зачтено	отлично

### **Примерные задания к лабораторным работам**

#### **ЛАБОРАТОРНАЯ РАБОТА 1. ОСНОВЫ ЯЗЫКА C++. ОПЕРАТОРЫ.**

Получение практических навыков работы с операторами, потоками ввода-вывода

**Задание 1.** Напишите программу на C++, которая отобразит Ваше имя, фамилию и адрес (фиктивные данные).

**Задание 2.** Напишите программу на C++, которая выдает запрос на ввод расстояния в метрах и переводит это расстояние в ярды ( $1 \text{ м} = 1.0936133 \text{ ярда}$ ).

**Задание 3.** Напишите программу на C++, которая предлагает пользователю ввести свой возраст. Затем программа должна отобразить его в месяцах.

**Задание 4.** Напишите программу, в которой функция `main()` вызывает определяемую пользователем функцию, принимающую в качестве аргумента значение температуры по Цельсию и возвращающую эквивалентное значение температуры по Фаренгейту. Программа должна выдать запрос на ввод значения по Цельсию и отобразить следующий результат:

```
Please enter a Celsius value: 20
20 degrees Celsius is 68 degrees Fahrenheit.
```

**Задание 5.** Напишите программу, которая выдает запрос на ввод значений часов и минут. Функция `main ()` должна передать эти два значения функции, имеющей тип `void`, которая отобразит эти два значения в следующем виде:

```
Enter the number of hours: 9
Enter the number of minutes: 28
Time: 9:28
```

## **ЛАБОРАТОРНАЯ РАБОТА 2. РАБОТА С ДАННЫМИ. ТИПЫ ДАННЫХ C++**

Изучить встроенные типы данных языка C++, арифметические операции. Работа с константами.

**Задание 1.** Напишите короткую программу, которая запрашивает рост в футах и дюймах и вес в фунтах. (Для хранения этой информации используйте три переменных.) Программа должна выдать индекс массы тела (`body mass index` — `BMI`). Чтобы рассчитать `BMI`, сначала преобразуйте рост в футах и дюймах в рост в дюймах ( $1 \text{ фут} = 12 \text{ дюймов}$ ). Затем преобразуйте рост в дюймах в рост в метрах, умножив на  $0.0254$ . Далее преобразуйте вес в фунтах в массу в килограммах, разделив на  $2.2$ . После этого рассчитайте `BMI`, разделив массу в килограммах на квадрат роста в метрах. Для представления различных коэффициентов преобразования используйте символические константы.

**Задание 2.** Напишите программу, которая запрашивает широту в градусах, минутах и секундах, после чего отображает широту в десятичном формате. В одной минуте 60 угловых секунд, а в одном градусе 60 угловых минут; представьте эти значения с помощью символических констант. Для каждого вводимого значения должна использоваться отдельная переменная. Результат выполнения программы должен выглядеть следующим образом:

```
Enter a latitude in degrees, minutes, and seconds:  
First, enter the degrees: 37  
Next, enter the minutes of arc: 51  
Finally, enter the seconds of arc: 19  
37 degrees, 51 minutes, 19 seconds = 37.8553 degrees
```

**Задание 3.** Напишите программу, которая запрашивает количество секунд в виде целого значения (используйте тип `long` или `long long`, если последний доступен) и затем отображает эквивалентное значение в сутках, часах, минутах и секундах. Для представления количества часов в сутках, количества минут в часе и количества секунд в минуте используйте символические константы. Результат выполнения программы должен выглядеть следующим образом:

```
Enter the number of seconds: 31600000  
31600000 seconds = 365 days, 17 hours, 46 minutes, 40 seconds
```

**Задание 4.** Напишите программу, которая запрашивает текущую численность населения Земли и текущую численность населения России (или любой другой страны). Сохраните эту информацию в переменных типа `long long`. В качестве результата программа должна отображать процентное соотношение численности населения России (или выбранной страны) и всего мира. Результат выполнения программы должен выглядеть следующим образом:

```
Enter the world's population: 6898758899  
Enter the population of the Russia: 310783781  
The population of the Russia is 4.50492% of the world population.
```

**Задание 5.** Напишите программу, которая запрашивает текущую численность населения Земли и текущую численность населения России (или любой другой страны). Сохраните эту информацию в переменных типа `long long`. В качестве результата программа должна отображать процентное соотношение численности населения России (или выбранной страны) и всего

мира. Результат выполнения программы должен выглядеть следующим образом:

Enter the world's population: 7046000000

Enter the population of the Russia: 143500000

The population of the Russia is 2.036617% of the world population.

### **ЛАБОРАТОРНАЯ РАБОТА 3. СОСТАВНЫЕ ТИПЫ ДАННЫХ. МАССИВЫ. СТРУКТУРЫ**

Изучение составных типов данных языка C++: структуры, массивы, строки. Освоить практические навыки работы с ними. Указатели.

**Задание 1.** Напишите программу, которая приглашает пользователя ввести его имя и фамилию, а затем построит, сохранит и отобразит третью строку, состоящую из фамилии, за которой следует запятая, пробел и имя. Используйте объекты `string` и методы из заголовочного файла `string`. Пример запуска должен выглядеть так:

Enter your first name: Ivan

Enter your last name: Petrov

Here's the information in a single string: Petrov, Ivan

**Задание 2.** Структура `CandyBar` содержит три члена. Первый из них хранит название коробки конфет. Второй — ее вес (который может иметь дробную часть), а третий — число калорий (целое значение). Напишите программу, объявляющую эту структуру и создающую переменную типа `CandyBar` по имени `snack`, инициализируя ее члены значениями "Snack", 2.3 и 350, соответственно. Инициализация должна быть частью объявления `snack`. И, наконец, программа должна отобразить содержимое этой переменной.

**Задание 3.** Структура `CandyBar` включает три члена, как описано в предыдущем упражнении. Напишите программу, которая создает массив из трех структур `CandyBar`, инициализирует их значениями по вашему усмотрению и затем отображает содержимое каждой структуры.

**Задание 4.** Напишите программу, которая приглашает пользователя ввести три результата забега на 40 ярдов (или 40 метров, если желаете) и затем отображает эти значения и их среднее.

### **ЛАБОРАТОРНАЯ РАБОТА 4. ЦИКЛЫ. ВЫРАЖЕНИЯ ОТНОШЕНИЙ**

Получение практических навыков работы с циклами (`for`, `while do`, `do while`). Работа с вложенными циклами, двумерными массивами.

**Задание 1.** Напишите программу, запрашивающую у пользователя ввод двух целых чисел. Затем программа должна вычислить и выдать сумму всех целых чисел, лежащих между этими двумя целыми. Предполагается, что меньшее значение вводится первым. Например, если пользователь ввел 2 и 9, программа должна сообщить, что сумма всех целых чисел от 2 до 9 равна 44.

**Задание 2.** Напишите программу, которая приглашает пользователя вводить числа. После каждого введенного значения программа должна выдавать накопленную сумму введенных значений. Программа должна завершаться при вводе 0.

**Задание 3.** Дарья инвестировала 1000 руб под простые 10%. Другими словами, ежегодно инвестиция должна приносить 10% инвестированной суммы, т.е. 100 руб каждый год:

прибыль = 0,10 x исходный баланс

В то же время Гликерья инвестировала 1000 руб под сложные 5%. Это значит, что прибыль составит 5% от текущего баланса, включая предыдущую накопленную прибыль:

прибыль = 0,05 x текущий баланс

Клео зарабатывает 5% от 1000 руб в первый год, что дает ей 1050. На следующий год она зарабатывает 5% от 1050, что составляет 52,5 руб, и т.д. Напишите программу, которая вычислит, сколько лет понадобится для того, чтобы сумма баланса Гликерьи превысила сумму баланса Дарьи, с отображением значений обоих балансов за каждый год.

**Задание 4.** Предположим, что вы продаете книгу по программированию на языке C++ для начинающих. Напишите программу, которая позволит ввести ежемесячные объемы продаж в течение года (в количестве книг, а не в деньгах). Программа должна использовать цикл, в котором выводится приглашение с названием месяца, применяя массив указателей на char (или массив объектов string, если вы предпочитаете его), инициализированный строками — названиями месяцев, и сохраняя введенные значения в массиве int. Затем программа должна найти сумму содержимого массива и выдать общий объем продаж за год.

**Задание 5.** Выполните задание 4, но используя двумерный массив для сохранения данных о месячных продажах за 3 года. Выдайте общую сумму продаж за каждый год и за все годы вместе.

**Задание 6.** Разработайте структуру по имени `car`, которая будет хранить следующую информацию об автомобиле: название производителя в виде строки в символьном массиве или в объекте `string`, а также год выпуска автомобиля в виде целого числа. Напишите программу, которая запросит пользователя, сколько автомобилей необходимо включить в каталог. Затем программа должна применить `new` для создания динамического массива структур `car` указанного пользователем размера. Далее она должна пригласить пользователя ввести название производителя и год выпуска для наполнения данными каждой структуры в массиве. И, наконец, она должна отобразить содержимое каждой структуры. Пример запуска программы должен выглядеть подобно следующему:

Сколько автомобилей поместить в каталог? 2

Автомобиль #1:

Введите производителя: Toyota

Укажите год выпуска: 2007

Автомобиль #2:

Введите производителя: Lexus

Укажите год выпуска: 2010

Вот ваша коллекция:

2007 Toyota 2010 Lexus

**Задание 7.** Напишите программу, использующую вложенные циклы, которая запрашивает у пользователя значение количества строк для отображения. Затем она должна отобразить указанное число строк со звездочками, с одной звездочкой в первой строке, двумя — во второй и т.д. В каждой строке звездочкам должны предшествовать точки — в таком количестве, чтобы общее число символов в каждой строке было равно количеству строк. Пример запуска программы должен выглядеть следующим образом:

Введите количество строк: 5.

```
....*
...**
..***
.****
*****
```



**Задание 8.** Напишите программу, которая использует массив `char` и цикл для чтения по одному слову за раз до тех пор, пока не будет введено слово `done`. Затем программа должна сообщить количество введенных слов (исключая `done`).

Пример запуска должен быть таким:

Вводите слова (для завершения введите слово `done`): anteater birthday  
category dumpster envy finagle geometry done for sure

Вы ввели 7 слов .

## **ЛАБОРАТОРНАЯ РАБОТА 5. ОПЕРАТОРЫ ВЕТВЛЕНИЯ. ЛОГИЧЕСКИЕ ОПЕРАТОРЫ**

Получение практических навыков работы с оператором `if then else`, логическими операторами, операторами `switch`, а также `break` и `continue`. Работа с файлами.

**Задание 1.** Напишите программу, которая читает клавиатурный ввод до символа `@` и повторяет его, за исключением десятичных цифр, преобразуя каждую букву верхнего регистра в букву нижнего регистра и наоборот.

**Задание 2.** Напишите программу, читающую в массив `double` до 10 значений пожертвований. Программа должна прекращать ввод при получении нечисловой величины. Она должна выдавать среднее значение полученных чисел, а также количество значений в массиве, превышающих среднее.

**Задание 3.** Напишите предшественник программы, управляемой меню. Она должна отображать меню из четырех пунктов, каждый из них помечен буквой. Если пользователь вводит букву, отличающуюся от четырех допустимых, программа должна повторно приглашать его ввести правильное значение до тех пор, пока он этого не сделает. Затем она должна выполнить некоторое простое действие на основе пользовательского выбора. Работа программы должна выглядеть примерно так:

Please enter one of the following choices:

c) carnivore            p) pianist

t) tree                    g) game

f

Please enter a c, p, t, or g: q

Please enter a c, p, t, or g: t

A maple is a tree.

**Задание 4.** Постройте программу, которая отслеживает пожертвования в Общество Защиты Влиятельных Лиц. Она должна запрашивать у пользователя количество меценатов, а затем приглашать вводить их имена и суммы пожертвований от каждого. Информация должна сохраняться в динамически выделяемом массиве структур. Каждая структура должна иметь два члена: символьный массив (или объект string) для хранения имени и переменную-член типа double — для хранения суммы пожертвования. После чтения всех данных программа должна отображать имена и суммы пожертвований тех, кто не пожалел 10 000 и более. Этот список должен быть озаглавлен меткой “Grand Patrons”. После этого программа должна выдать список остальных жертвователей. Он должен быть озаглавлен “Patrons”. Если в одной из двух категорий не окажется никого, программа должна напечатать “none”. Помимо отображения двух категорий, никакой другой сортировки делать не нужно.

**Задание 5.** Напишите программу, которая читает слова по одному за раз, пока не будет введена отдельная буква q. После этого программа должна сообщить количество слов, начинающихся с гласных, количество слов, начинающихся с согласных, а также количество слов, не попадающих ни в одну из этих категорий.

**Задание 6.** Напишите программу, которая открывает текстовый файл, читает его символ за символом до самого конца и сообщает количество символов в файле.

**Задание 7.** Выполните задание 4, но измените его так, чтобы данные можно было получать из файла. Первым элементом файла должно быть количество меценатов, а остальная часть состоять из пар строк, в которых первая строка содержит имя, а вторая — сумму пожертвования.

## ЛАБОРАТОРНАЯ РАБОТА 6. ФУНКЦИИ. РЕКУРСИЯ

Получение практических навыков работы в области написания собственных пользовательских функций а также рекурсивных функций

**Задание 1.** Напишите программу, которая многократно запрашивает у пользователя пару чисел до тех пор, пока хотя бы одно из этой пары не будет равно 0. С каждой парой программа должна использовать функцию для вычисления среднего гармонического этих чисел. Функция должна возвращать ответ main () для отображения результата. Среднее

гармоническое чисел — это инверсия среднего значения их инверсий; она вычисляется следующим образом:

$$\text{среднее гармоническое} = 2.0 * x y / (x+y)$$

**Задание 2.** Напишите программу, которая запрашивает у пользователя 10 результатов игры в гольф, сохраняя их в массиве. При этом необходимо обеспечить возможность прекращения ввода до ввода всех 10 результатов. Программа должна отобразить все результаты в одной строке и сообщить их среднее значение. Реализуйте ввод, отображение и вычисление среднего в трех отдельных функциях, работающих с массивами.

**Задание 3.** Пусть имеется следующее объявление структуры:

```
struct box {  
    char maker[40]; float height; float width; float length; float volume;  
};
```

- а. Напишите функцию, принимающую структуру `box` по значению и отображающую все ее члены.
- б. Напишите функцию, принимающую адрес структуры `box` и устанавливающую значение члена `volume` равным произведению остальных трех членов.
- в. Напишите простую программу, которая использует эти две функции.

**Задание 4.** Определите рекурсивную функцию, принимающую целый аргумент и возвращающую его факториал. Вспомните, что факториал 3 записывается, как 3! и вычисляется как  $3 \times 2!$  и т.д., причем 0! равно 1. В общем случае, если  $n$  больше нуля, то  $n! = n * (n - 1)!$ . Протестируйте функцию в программе, использующей цикл, где пользователь может вводить различные значения, для которых программа вычисляет и отображает факториалы.

**Задание 5.** Напишите программу, использующую описанные ниже функции.

`Fill_array ()` принимает в качестве аргумента имя массива элементов типа `double` и размер этого массива. Она приглашает пользователя ввести значения `double` для помещения их в массив. Ввод прекращается при наполнении массива либо когда пользователь вводит нечисловое значение и возвращает действительное количество элементов.

`Show array ()` принимает в качестве аргументов имя массива значений `double`, а также его размер, и отображает содержимое массива.

`Reverse_array ()` принимает в качестве аргумента имя массива значений `double`, а также его размер, и изменяет порядок его элементов на противоположный.

Программа должна использовать эти функции для наполнения массива, обращения порядка его элементов, кроме первого и последнего, с последующим отображением.

**Задание 6.** Вернитесь к заданию 5 и замените три функции обработки массивов версиями, которые работают с диапазонами значений, заданными парой указателей. Функция `fill_array ()` вместо возврата действительного количества прочитанных значений должна возвращать указатель на место, следующее за последним введенным элементом; прочие функции должны использовать его в качестве второго аргумента для идентификации конца диапазона данных.

**Задание 7.** Спроектируйте функцию `calculate ()`, которая принимает два значения типа `double` и указатель на функцию, принимающую два аргумента `double` и возвращающую значение `double`. Функция `calculate ()` также должна иметь тип `double` и возвращать значение, вычисленное функцией, которая задана указателем, используя аргумент `double` функции `calculate ()`.

Например, предположим, что имеется следующее определение функции `add ()`: `double add(double x, double y)`

```
{  
    return x + y;  
}
```

Приведенный ниже вызов функции должен заставить `calculate ()` передать значения 2.5 и 10.4 функции `add ()` и вернуть ее результат (12.9):

```
double q = calculate (2.5, 10.4, add);
```

Используйте в программе эти функции и еще хотя бы одну дополнительную, которая подобна `add ()`. В программе должен быть организован цикл, позволяющий пользователю вводить пары чисел. Для каждой пары `calculate ()` должна вызвать `add ()` и хотя бы еще одну функцию такого рода. Если вы чувствуете себя уверенно, попробуйте создать массив указателей на функции, подобные `add ()`, и организуйте цикл, применяя `calculate ()` для вызова этих функций по их указателям.

Подсказка: вот как можно объявить массив из трех таких указателей:

```
double (*pf [3]) (double, double);
```

Инициализировать такой массив можно с помощью обычного синтаксиса инициализации массивов и имен функций в качестве адресов.