

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФИЛИАЛ ФГБОУ ВО «ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ЭКОНОМИКИ И СЕРВИСА» В Г. НАХОДКЕ  
КАФЕДРА МЕНЕДЖМЕНТА И ЭКОНОМИКИ

## **ТЕХНОЛОГИЯ ПРОГРАММИРОВАНИЯ**

Рабочая программа дисциплины

по направлению подготовки

38.03.05 Бизнес-информатика

Рабочая программа дисциплины «Технология программирования» составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 38.03.05 «Бизнес-информатика» и «Порядком организации и осуществления образовательной деятельности по образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры» (утв. приказом Минобрнауки России от 5 апреля 2017 г. № 301).

Составители:


Васильев Б.К., к.х.н., доцент кафедры информационных технологий и систем,

Коломийцев А.К., преподаватель

Рассмотрена и утверждена на заседании кафедры менеджмента и экономики от «28» апреля 2018 года, протокол № 9

Заведующий кафедрой (разработчика)

«28» апреля 2018г.

  
\_\_\_\_\_

подпись

Прасолова В.С.

фамилия, инициалы

Заведующий кафедрой (выпускающей)

«28» апреля 2018г.

  
\_\_\_\_\_

подпись

Прасолова В.С.

фамилия, инициалы

## 1 Цель и задачи освоения дисциплины

Целью освоения дисциплины «Технология программирования» является теоретическая и практическая подготовка студентов в области технологий разработки программ в такой степени, чтобы при менеджменте программного проекта или в процессе участия в его реализации они могли выбирать необходимые технические, алгоритмические, программные и технологические решения, уметь объяснить принципы их функционирования и правильно их использовать. Иметь представление о каждом этапе жизненного цикла программы от проектирования до внедрения и сопровождения. Знать современные стандарты качества программного обеспечения и перспективные направления развития технологии разработки ПО.

Задачи освоения дисциплины состоят: в освоении основных положений технологии разработки ПО, формулировка практических рекомендаций по организации работы коллективов программистов, руководства такими коллективами, формировании у студентов знаний по дисциплине, связанных с процессом разработки ПО, включая связи с предметной областью, реализацию, организацию производства, контроль сроков исполнения и качества, ознакомлении с техническими программными и технологическими решениями, используемыми при разработке ПО; А также в приобретении практических навыков работы в коллективе программистов, умения находить правильные технологические решения по выбору средств разработки и структуры программного проекта, методов тестирования и контроля исполнения использование современных инструментальных и методологических средств.

## 2 Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

Планируемыми результатами обучения по дисциплине, являются знания, умения, навыки и/или опыт деятельности, характеризующие этапы/уровни формирования компетенций и обеспечивающие достижение планируемых результатов освоения образовательной программы в целом. Перечень компетенций, формируемых в результате изучения дисциплины, приведен в таблице 1.

Таблица 1 – Формируемые компетенции

Название ОПОП ВО (сокращенное название)	Компетенции	Название компетенции	Составляющие компетенции	
38.03.05 Бизнес-информатика (Б-БИ)	ОПК-3	способностью работать с компьютером как средством управления информацией, работать с информацией из различных источников, в том числе в глобальных компьютерных сетях	Знания:	основ технологий разработки программ, современных стандартов качества программного обеспечения и перспективных направлений развития технологии разработки программного обеспечения
			Умения:	выбирать необходимые технические, алгоритмические, программные и технологические решения и правильно их использовать

			Навыки:	владения основами технологий разработки программного обеспечения
	ПК-18	способность использовать соответствующий математический аппарат и инструментальные средства для обработки, анализа и систематизации информации по теме исследования	Знания:	инструментальных средства для обработки, анализа и систематизации информации по теме исследования
			Умения:	использовать инструментальные средства для обработки, анализа и систематизации информации по теме исследования
			Навыки:	владения основными инструментальными средствами для обработки, анализа и систематизации информации по теме исследования

### 3 Место дисциплины в структуре основной образовательной программы

Отнесение дисциплины к дисциплинам по выбору вариативной части ОПОП определяется спецификой и миссией ВГУЭС, а также особенностями взаимодействия ВГУЭС с рынком труда и региональными требованиями, выраженными в результатах образования и компетенциях.

Входными требованиями, необходимыми для освоения дисциплины, является наличие у обучающихся компетенций, сформированных при изучении дисциплин «Информатика и программирование модуль 1», «Компьютерный анализ данных», «Объектно-ориентированное программирование», «Операционные системы», «Основы алгоритмизации и языки программирования», «Теория принятия решений».

На данную дисциплину опираются дисциплины «Интеллектуальные информационные системы», «Информационные технологии управления персоналом», «Проектный практикум», «Теория информационных процессов и систем», «Хранилища данных».

### 4. Объем дисциплины

Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу с обучающимися (по видам учебных занятий) и на самостоятельную работу по всем формам обучения, приведен в таблице 2.

Таблица 2 – Общая трудоемкость дисциплины

Название ОПОП	Форма обучения	Цикл	Курс	Трудоемкость (З.Е.)	Объем контактной работы (час)					СРС	Форма аттестации	
					Всего	Аудиторная			Внеаудиторная			
				лек.		прак.	лаб.	ПА	КСР			
Б-БИ	ОЗФО	Б.1.ДВ.Ж.02	3	4	24	15	–	10	9	–	120	ДЗ

## 5 Структура и содержание дисциплины

### 5.1 Структура дисциплины

Тематический план, отражающий содержание дисциплины (перечень разделов и тем), структурированное по видам учебных занятий с указанием их объемов в соответствии с учебным планом, приведен в таблице 3.

Таблица 3 – Структура дисциплины

№	Название темы	Вид занятия
1	Основные понятия технологии программирования, особенности программного проекта. Способы преодоления сложностей при разработке.	Лекция
		Лабораторная работа
2	Значение предметной области. Различные модели процесса разработки ПО. АТД, ориентированные на предметную область, оценка осуществимости проекта, графики выполнения.	Лекция
		Лабораторная работа
3	Тестирование, обеспечение качества: Критерии качества и их метрики. Статическое и динамическое тестирование. Методы белого и черного ящиков. Создание тестовых наборов данных.	Лекция
		Лабораторная работа
4	Средства автоматизации при разработке синтаксических анализаторов. Понятия грамматики языка, лексического и синтаксического разбора. генераторы распознавателей yacc, bison. Лингвистический подход при разработке приложений.	Лекция
		Лабораторная работа
5	Групповая разработка, управление версиями. Параллельная и конкурентная разработка. Различные способы организации коллектива разработчиков. Основные и вспомогательные подразделения на предприятии и их задачи	Лекция
6	Сопровождение: Исправление ошибок, внесение дополнительной функциональности, повышение эффективности. Требования, предъявляемые к ПО и документации для реализации успешного сопровождения	Лекция
		Лабораторная работа
7	Разработка интерфейса пользователя: решаемые задачи и средства. Целесообразность и метафоричность интерфейса. Виды интерфейсов. Средства для разработки интерфейсов. Реинжиниринг программных систем: Перевод устаревших программ на новые языки и платформы, возвратное проектирование.	Лекция
		Лабораторная работа

### 5.2 Содержание дисциплины

**Тема 1.** Содержание темы: Понятие технологии программирования: Особенности промышленного программирования, "программирование для себя" (Just for fun) и "программирование на заказ". Языки программирования как средство выражения алгоритмов и как средство получения выполняемого кода. Распределение памяти в языке Си. Жизненный цикл программного обеспечения (ПО). Общая организация проекта. Модели разработки ПО. Основные технологические подходы: каскадный, каркасный, сборочный, адаптивный (экстремальное программирование). Трудности проектирование, их разновидности и преодоление. Примеры. Понятие АТД. АТД, ориентированные на предметную область (пример из геометрии).

Литература по теме: [1,2,6]

Формы и методы проведения занятий по теме: Лекция и лабораторная работа №1, Разработка приложения — игры в «крестики-нолики» на поле заданного размера, содержащее графический интерфейс и содержащее аналитическую часть для проверки завершающего хода. Используется построитель графических интерфейсов fluid.

Самостоятельно — изучение набора управляющих элементов библиотеки FLTK, использование fluid.

Форма текущего контроля: выполненная лабораторная.

**Тема 2.** Значение предметной области. Каскадная и спиралевидная модели процесса разработки ПО. Возникновение и исследование идеи. Иллюстрации на предыдущем примере АТД, оценка осуществимости: Как оценить сложность задачи? Реальность ее решения в заданные сроки при заданных рамочных ограничениях. Планирование: Сетевой и ленточный графики, треугольник – сроки, работы, ресурсы. Анализ требований и выработка спецификаций ПО. Проектирование архитектуры продукта. Выбор средств реализации. Инструментарий разработчика. Технологии RAD и готовые компоненты. Библиотеки алгоритмов и подпрограмм. .

Литература по теме: [1,2,5]

Формы и методы проведения занятий по теме: Лекция —лабораторная работа №2, Разработка приложения, считывающее структурированные данные из файла и заполняющее структуры данных. Используется внешняя библиотека для манипуляций с данными (3D-преобразования над множеством точек). Используется построитель графических интерфейсов fluid. Самостоятельно — модификация библиотеки работы с матрицами.

Форма текущего контроля: выполненная лабораторная.

**Тема 3.** Тестирование, обеспечение качества: Критерии качества и их метрики. Стандарты ISO 9000, 9001. Стандартизация информационных технологий. Статическое и динамическое тестирование. Методы белого и черного ящиков. Создание тестовых наборов данных. Примеры из конкретной предметной области. Групповая разработка (постановка задачи).

Литература по теме: [1,2]

Формы и методы проведения занятий по теме: Лекция —лабораторная работа №3, Разработка приложения, демонстрирующего многомерный куб (4 или 5 измерений), на 2-мерных проекциях. Изменить и протестировать библиотеку для работы с матрицами, добавив функции заполнения матриц поворота, переноса, масштабирования. Используется построитель графических интерфейсов fluid и результаты предыдущей работы. Самостоятельно — модификация библиотеки работы с матрицами, анализ ресурсов интернета, посвященных многомерным многогранникам.

Форма текущего контроля: выполненная лабораторная.

**Тема 4.** Средства автоматизации решения традиционно считающихся трудными задач. Разработка синтаксических анализаторов. Понятия грамматики языка, лексического и синтаксического разбора. Способы описания грамматики. Нотация БНФ, генератор распознавателей yacc, bison. Лингвистический подход при разработке приложений. Примеры малых языков и преимущества их использования.

Литература по теме: [6,7]

Формы и методы проведения занятий по теме: Лекция лабораторная работа №4, Реализовать программу, включающую графический интерфейс инженерного калькулятора с использованием специальных функций и «построителя графиков» функций одной и двух переменных по заданной формуле. Самостоятельно найти информацию о применении и вычислении специальных функций (гиперболических синусов, косинусов, функций Бесселя).

Форма текущего контроля: выполненная лабораторная.

**Тема 5.** Групповая разработка, управление версиями. Единый репозиторий проекта. Системы RCS, CVS. Методы нумерации версий. Параллельная и конкурентная разработка. Организация коллектива разработчиков: Матричный метод, метод главного специалиста, вертикальные и горизонтальные координации управления проектом. Основные и вспомогательные подразделения и их задачи.

Литература по теме: [1,3,4]

Формы и методы проведения занятий по теме: Лекция самостоятельно найти информацию об организации процесса разработки ПО на крупных предприятиях.

**Тема 6.** Сопровождение: Исправление ошибок, внесение дополнительной функциональности, повышение эффективности. Требования, предъявляемые к ПО и документации для реализации успешного сопровождения.

Литература по теме: [2,5,6]

Формы и методы проведения занятий по теме: Лекция лабораторная работа №5, Реализовать программу, включающую язык манипуляций с матрицами и матричные операции.

Форма текущего контроля: выполненная лабораторная.

**Тема 7.** Разработка интерфейса пользователя: решаемые задачи и средства. Целесообразность и метафоричность интерфейса. Виды интерфейсов. Средства для разработки интерфейсов. Реинжиниринг программных систем: Перевод устаревших программ на новые языки и платформы, возвратное проектирование – извлечение знаний из текста программы.

Литература по теме: [1,5,7]

Формы и методы проведения занятий по теме: Лекция лабораторная работа №6, Выполнить переработку графического интерфейса лабораторных работ 1-4. Реализовать функции контроля времени. Самостоятельно найти информацию о наиболее используемых библиотеках управляющих графических элементах.

Форма текущего контроля: выполненная лабораторная.

## **6. Методические указания для обучающихся по освоению дисциплины**

В ходе изучения дисциплины «Технология программирования» студенты могут посещать аудиторные занятия (лекции, лабораторные занятия, практические занятия, консультации). Особенность изучения дисциплины «Технология программирования» состоит в том, что студенты в ходе выполнения лабораторных работ разбирают и анализируют способы проектирования, изготовления, интеграции и тестирования многомодульного ПО.

Особое место в овладении частью тем данной дисциплины отводится самостоятельной работе, при этом во время аудиторных занятий могут быть рассмотрены и проработаны наиболее важные и трудные вопросы по той или иной теме дисциплины, а второстепенные и более легкие вопросы, а также вопросы, специфичные для направления подготовки, могут быть изучены студентами самостоятельно.

Для очной формы обучения в соответствии с учебным планом направления подготовки процесс изучения дисциплины может предусматривать проведение лекций, лабораторных занятий, консультаций, а также самостоятельную работу студентов. Обязательным является проведение лабораторных занятий в специализированных компьютерных аудиториях, оснащенных персональными компьютерами с установленными операционными системами и средствами разработки приложений различного назначения.

Для заочной формы обучения процесс изучения дисциплины может предусматривать проведение установочных и обзорных лекций в аудиториях с мультимедийным оборудованием и лабораторных занятий по ключевым практическим темам дисциплины в специализированных компьютерных аудиториях, а также проведение консультаций. Наибольшая часть учебного времени отводится на самостоятельную работу студентов, во время которой студентами заочной формы обучения должны быть выполнены контрольные работы.

Ниже перечислены предназначенные для самостоятельного изучения студентами очной формы обучения те вопросы из лекционных тем, которые во время проведения аудиторных занятий изучаются недостаточно или изучение которых носит обзорный характер.

В рамках общего объема часов, отведенных для изучения дисциплины, предусматривается выполнение следующих видов самостоятельных работ студентов (СРС): изучение теоретического материала при подготовке к защите лабораторных работ, итоговое повторение теоретического материала. Для самостоятельного изучения дисциплины

выносятся часть материала по всем темам дисциплины с самоконтролем по контрольным вопросам и возможностью консультации у ведущего преподавателя. Для закрепления материала и приобретения навыков расчета рекомендуется выполнение следующих задач:

1. Разработка многомодульного программного проекта «калькулятор с возможностью построения графиков произвольных функций одной и двух переменных»
2. Ознакомление с методами автоматической генерации синтаксических распознавателей (bison, Zubr, eli). Анализ грамматики известных языков программирования (C, Pascal, yacc, prolog).
3. Установка средств разработки программного обеспечения open source (компилятор gcc, отладчик), используя пакет исходных текстов («тарбол»).
4. Разработка плана работ и сетевого графика для собственного программного проекта.

Ниже приведены рекомендации по работе с литературой. В список рекомендованной литературы включены только издания, содержащие фундаментальные знания, не претерпевшие изменений за последние годы, и новые источники, отражающие современное состояние в рассматриваемой области.

Для изучения теоретического материала по дисциплине (темы 1-3, 5, 6) можно предложить классические учебники [1, 2].

Параллельно с ним для изучения тем 2, 4 рекомендуется использовать [6, 7, 8], где ряд вопросов рассмотрен несколько более подробно, а для темы 4 – [7, каталог /pub/TP/YACC] соответственно.

Для закрепления материала (приобретения практических навыков написания многомодульных программ и составления планов выполнения работ, документации к лаб. работам) предпочтительнее использовать примеры из [3, 4, 5].

При выполнении контрольных работ и индивидуальных заданий рекомендуется использовать дополнительную литературу [5-8].

Для выполнения лабораторных работ во ВГУЭС материалы в достаточном количестве изданы и размещены на сервере [7]. Эти материалы соответствуют перечню лабораторных работ раздела (распределены по каталогам).

## **7. Перечень учебно-методического обеспечения для самостоятельной работы**

В процессе самостоятельной работы при изучении дисциплины студенты могут использовать в специализированных аудиториях компьютеры с установленными операционными системами, виртуальные машины, размещенные в них, обеспечивающему доступ к аппаратным ресурсам, современному программному обеспечению, необходимому для изучения дисциплины, а также доступ через локальную сеть университета к студенческому файловому серверу и через информационно-телекоммуникационную сеть «Интернет» к электронной образовательной среде и к хранилищу полнотекстовых материалов, где в электронном виде располагаются учебно-методические и раздаточные материалы, которые могут быть использованы для самостоятельной работы при изучении дисциплины. При работе дома можно использовать на виртуальной машине предустановленную ОС со всем необходимым ПО (см. п.13), справочными материалами и книгами в электронном формате.

## **8. Фонд оценочных средств для проведения промежуточной аттестации**

В соответствии с требованиями ФГОС ВО для аттестации обучающихся на соответствие их персональных достижений планируемым результатам обучения по дисциплине созданы фонды оценочных средств (Приложение 1).



## **9. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины**

а) основная литература

- Антамошкин О.А. Программная инженерия. Теория и практика: учебник / О.А. Антамошкин. - Красноярск: Сиб. Федер. ун-т, 2012. - 247 с.  
[Электронный ресурс] Режим доступа: <http://znaniium.com/bookread2.php?book=492527>
- Иванова Г.С. Технология программирования: учебник для студентов вузов / Г.С. Иванова. - М.: КНОРУС, 2011. - 336 с.

б) дополнительная литература

- Рейнвотер, Дж.  
Как пасти котов: наставление для программистов, руководящих другими программистами: пер. с англ. / Д. Рейнвотер. - СПб.: Питер, 2006. - 256 с.: ил. - (Библиотека программиста). - Пер. изд.: Herding Cats: A Primer for Programmers Who Lead Programmers/ J.H. Rainwater, U.S, Apress, 2002.
- Лакман Макдауэлл Г. Карьера программиста. 6-е изд., Спб, Питер, 2016.
- Орам Э., Уилсон Г. Идеальная разработка ПО. Рецепты лучших программистов, Спб, Питер, 2012.
- Клейнберг Д., Тардос Е. Алгоритмы: разработка и применение, Спб, Питер, 2016.

## **10. Перечень ресурсов информационно - телекоммуникационной сети**

### **«Интернет»**

а) полнотекстовые базы данных:

1. Научная электронная библиотека eLIBRARY.RU: <http://www.eLIBRARY.RU>
2. Ресурс Цифровые учебные материалы <http://abc.vvsu.ru/>
3. ЭБС «Руконт»: <http://www.rucont.ru/>
4. ЭБС «Юрайт»: <http://www.biblio-online.ru/>

## **11. Перечень информационных технологий**

Для проведения лекционных и лабораторных занятий рекомендуется использовать программное обеспечение: операционная система Linux с ядром 3.2 и выше, обслуживающие программы и среды разработки программ, основанные на открытом исходном коде, по выбору преподавателей.

## **12. Электронная поддержка дисциплины**

При изучении дисциплины для проработки всех тем и выполнения заданий по всем темам студенты могут использовать различные учебно-методические материалы, размещаемые в электронном виде преподавателями на файловом ftp-сервере [7], в хранилище полнотекстовых материалов, а также в электронной образовательной среде, которая предполагает также возможность обмена информацией с преподавателем для подготовки заданий. Доступ студентов к студенческому файловому серверу, хранилищу полнотекстовых материалов, электронной образовательной среде осуществляется с использованием с использованием учетных записей студентов.

## **13. Материально-техническое обеспечение дисциплины**

Для проведения лекций по дисциплине используются специализированные аудитории с мультимедийным оборудованием или с возможностями подключения к такому оборудованию, позволяющему демонстрировать на большом экране приемы работы с персональным компьютером и другой лекционный материал (технические характеристики

компьютера, входящего в состав мультимедийного оборудования или используемого совместно с таким оборудованием, должны обеспечивать возможность работы с современными версиями операционной системы Linux и Unix, пакета LibreOffice 5 версии и старше, обслуживающих, прикладных программ и другого, в том числе и сетевого программного обеспечения). Обязательным является наличие компиляторов и средств разработки ПО.

Для проведения лабораторных занятий по дисциплине и для самостоятельной работы студентов используются специализированные аудитории, оснащенные терминалами и персональными компьютерами, подключенными к центральному серверу, обеспечивающему технические характеристики обслуживания терминалов или персональных компьютеров, позволяющие при проведении лабораторных занятий использовать современное программное обеспечение (операционные системы Linux, FreeBSD, а также обслуживающие программы и среды разработки программ на языках высокого уровня по выбору преподавателей). На ресурсе [7] размещены образы предустановленных и сконфигурированных 32- и 64-разрядных операционных систем для виртуальных машин (в каталоге /pub/DISTR/VM) со всем необходимым программным обеспечением, включая среды разработки и электронные варианты книг.