



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

**ИНСТИТУТ МАТЕМАТИКИ И КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ**

«СОГЛАСОВАНО»

Руководитель ОП

Сухомлинов А.И.

«УТВЕРЖДАЮ»

Директор департамента информационных и компьютерных систем

Пустовалов Е.В.

« 01 » марта 2022 г.



**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**  
Технология программирования  
**Направление подготовки 09.03.03 Прикладная информатика**  
(Прикладная информатика в управлении)  
**Форма подготовки очная**

курс  2  семестр  4

лекции  36  час.

практические занятия  не предусмотрены

лабораторные работы  36  час.

в том числе с использованием МАО лек.  18  / пр.  -  / лаб.  36  час.

всего часов аудиторной нагрузки  72  час.

в том числе с использованием МАО  36  час.

самостоятельная работа  108  час.

в том числе на подготовку к экзамену  36  час.

контрольные работы (количество)  не предусмотрены

курсовая работа / курсовой проект  не предусмотрены

зачет  не предусмотрен

экзамен 4 семестр

Рабочая программа составлена в соответствии с требованиями Федерального государственного образовательного стандарта по направлению подготовки 09.03.03 Прикладная информатика утвержденного приказом Министерства образования и науки РФ от 19 сентября 2017 г. № 922 (с изменениями и дополнениями)

Рабочая учебная программа обсуждена на заседании департамента информационных и компьютерных систем, протокол № 7 от «25» февраля 2022 г.

Директор департамента информационных и компьютерных систем Пустовалов Е.В.

Составитель (ли): ст. преподаватель Березкина Г.Л.

Владивосток  
2022

## Оборотная сторона титульного листа РПД

### **I. Рабочая программа пересмотрена на заседании департамента:**

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

Директор департамента \_\_\_\_\_

(подпись)

(И.О. Фамилия)

### **II. Рабочая программа пересмотрена на заседании департамента:**

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

Директор департамента \_\_\_\_\_

(подпись)

(И.О. Фамилия)

### **III. Рабочая программа пересмотрена на заседании департамента:**

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

Директор департамента \_\_\_\_\_

(подпись)

(И.О. Фамилия)

### **IV. Рабочая программа пересмотрена на заседании департамента:**

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

Директор департамента \_\_\_\_\_

(подпись)

(И.О. Фамилия)

## Цели и задачи освоения дисциплины:

Цель: обучение бакалавров теоретическим основам и практическим навыкам применения технологии программирования для разработки программного продукта с целью достижения его высокого качества, а также процесса его создания.

Задачи дисциплины заключаются:

в освоении бакалаврами фундаментальных теоретических положений современных подходов, методов и технологий программирования,

в формировании у бакалавров интегрированного восприятия стратегии деятельности, организации предприятия и его информационных технологий,

в приобретении компетенций применения перспективных подходов, методов, средств технологии программирования и организации проведения работ по созданию конечного продукта:

в развитии умений проведения анализа существующих подходов и средств технологии программирования.

В результате изучения данной дисциплины у обучающихся формируются следующие профессиональные компетенции:

Тип задач	Код и наименование профессиональной компетенции (результат освоения)	Код и наименование индикатора достижения компетенции
производственно-технологический	<b>ПК-3</b> Способен изготавливать компоненты информационных систем, включая программные комплексы, базы данных и интерфейсы "человек - электронно-вычислительная машина", использовать современные инструментальные средства разработки, и программно-технологические платформы информационных систем	ПК-3.1 Использует современные технологии программирования, системы баз данных, сетевые технологии, методы, формализмы и стандарты представления диалогов и экранных форм интерфейсов «человек – ЭВМ», инструментальные средства разработки и программно-технологические платформы информационных систем
		ПК-3.2 Осуществляет изготовление спецификаций программного обеспечения, определяет и устанавливает параметры используемых коммерческих программных пакетов, контролирует качество, создаваемого программного обеспечения, создает базы данных, инсталлирует программное обеспечение и осуществляет загрузку баз данных, разработку технической документации, проводит начальное обучение и консультирование пользователей

Тип задач	Код и наименование профессиональной компетенции (результат освоения)	Код и наименование индикатора достижения компетенции
		ПК-3.3 Применяет системные и прикладные программно-технологические платформы, стандарты визуального представления интерфейсов и электронных форматов данных, средства CASE, инструментальные средства разработки информационных систем и программирования

Код и наименование индикатора достижения компетенции	Наименование показателя оценивания (результата обучения по дисциплине)
ПК-3.1 Использует современные технологии программирования, системы баз данных, сетевые технологии, методы, формализмы и стандарты представления диалогов и экранных форм интерфейсов «человек – ЭВМ», инструментальные средства разработки и программно-технологические платформы информационных систем	<p>Знать: современные методологии и технологии проектирования и использования баз данных, технологию программирования, системы баз данных, сетевые технологии, методы, формализмы и стандарты представления диалогов и экранных форм интерфейсов «человек – ЭВМ», инструментальные средства разработки и программно-технологические платформы информационных систем</p> <p>Уметь: применять современные методы и средства проектирования и реализации программного обеспечения, структур данных, баз данных, программных интерфейсов.</p> <p>Владеть: навыками проектирования и реализации структур данных, баз данных и программных интерфейсов</p>
ПК-3.2 Осуществляет изготовление спецификаций программного обеспечения, определяет и устанавливает параметры используемых коммерческих программных пакетов, контролирует качество, создаваемого программного обеспечения, создает базы данных, устанавливает программное обеспечение и осуществляет загрузку баз	Знать: архитектуру, принципы функционирования, элементную базу современных компьютеров, вычислительных и телекоммуникационных систем; терминологию, основные руководящие и регламентирующие документы в области ЭВМ, комплексов и систем; принципы организации процессора компьютера, памяти компьютера, компьютерных интерфейсных систем, иметь представление о параллельных компьютерных архитектурах; основы совместного программирования на ассемблере и языках высокого уровня; технические характеристики, показатели качества систем, методы

Код и наименование индикатора достижения компетенции	Наименование показателя оценивания (результата обучения по дисциплине)
данных, разработку технической документации, проводит начальное обучение и консультирование пользователей	их оценки и пути совершенствования
	Уметь: выбирать архитектуры и структуры систем, оценивать эффективность архитектурно-технических решений, реализованных при построении систем; представлять данные на машинном уровне; писать программы с использованием ассемблерных вставок; осуществлять сбор, обработку, анализ и систематизацию научно-технической информации в области систем с применением современных информационных технологий.
	Владеть: методиками оценки показателей качества и эффективности систем; навыки работы с различными типами информационных систем и технологий; знаниями по особенностям архитектуры вычислительных машин различных классов.
ПК-3.3 Применяет системные и прикладные программно-технологические платформы, стандарты визуального представления интерфейсов и электронных форматов данных, средства CASE, инструментальные средства разработки информационных систем и программирования	Знать: основные языки программирования и работу с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем и технологий
	Уметь: реализовывать программные приложения разной сложности на любых языках программирования с созданием баз данных; создавать программные прототипы решения прикладных задач;
	Владеть: системными и прикладными программно-технологическими платформами; объектно-ориентированными методами алгоритмизации и программирования.

## 2. Трудоёмкость дисциплины и видов учебных занятий по дисциплине

Общая трудоёмкость дисциплины составляет 5 зачётные единицы 108 академических часа).

(1 зачетная единица соответствует 36 академическим часам)

Видами учебных занятий и работы обучающегося по дисциплине являются:

Обозначение	Виды учебных занятий и работы обучающегося
Лек	Лекции
Лаб	Лабораторные работы

СР	Самостоятельная работа обучающегося в период теоретического обучения
Контроль	Самостоятельная работа обучающегося и контактная работа обучающегося с преподавателем в период промежуточной аттестации

### Структура дисциплины:

Форма обучения – очная.

№	Наименование раздела дисциплины	Семестр	Количество часов по видам учебных занятий и работы обучающегося						Формы промежуточной аттестации
			Лек	Лаб	Пр	ОК	СР	Контроль	
1	Раздел 1. Введение в технологию программирования	4	2		-				УО-1; УО-3
2	Раздел 2. Основные методологии	4	4	4		-	72	36	
3	Раздел 3. Технологии программирования	4	12	8					
4	Раздел 4. Технологические подходы	4	18	24					
	Итого:		36	36		-	36	36	

## I. СТРУКТУРА И СОДЕРЖАНИЕ ТЕОРЕТИЧЕСКОЙ ЧАСТИ КУРСА

### Лекционные занятия (36 час.)

#### Раздел 1. Введение в технологию программирования. (2 часа)

##### Тема 1. Базовые понятия технологий программирования (2 часа)

Определение основных понятий, используемых в курсе "Технология программирования". Определение алгоритма. Свойства алгоритма. Понятия кибернетики, связанные с понятием алгоритма. Модель. Моделирование. Жизненный цикл программного обеспечения. Жизненного цикла программного обеспечения. Проектирование. Проект.

##### Раздел 2. Основные методологии (4 часов)

##### Тема 1. Обзор методологий, методология императивного и объектно-ориентированного программирования (1 часа).

Обзор методологий императивного программирования, объектно-ориентированного программирования, функционального программирования, логического программирования, программирования в ограничениях и другие.

Методология императивного программирования. Методология объектно-ориентированного программирования.

##### Тема 2. Методология функционального программирования. . Методология логического программирования. (1 час)

Определение, происхождение методы и концепции, синтаксис и семантика. Класс задач.

##### Тема 3. Остальные методологии. (1 час)

Методология программирования в ограничениях. Методология структурного императивного программирования. Методология императивного параллельного программирования. Методологии программирования, управляемого потоками данных. Методологии доступ - ориентированного программирования. Методологии нейро-сетевого программирования (2 часа))

### **Раздел 3. Технологии программирования (4 часа)**

#### **Тема 1 Технологии программирования (1 часа)**

Технологии программирования – основные составляющие и понятия. Определение жизненного цикла программы. Определение технологии программирования. Определение процесса. Определение технологического подхода. Стадии жизненного цикла программного обеспечения. Классификация технологических подходов. Классификация технологических процессов. Процессы жизненного цикла, определяемые международным стандартом ISO 12207 [ISO/IEC 12207:1995]. Основные направления развития и совершенствования технологических процессов.

#### **Тема 2. Классические технологические процессы. (1 часа)**

Возникновение и исследование идеи. Управление.

#### **Тема 3. Анализ требований и проектирование. (1 часа)**

Проектирование в большом (проектирование архитектуры).

#### **Тема 4. Стиль программирования (кодирования) программы. (1 часа).**

Определения стиля программирования. Правило стандартизации стиля. Составляющие стиля программирования. Защитное программирование. Выбор языка программирования.

#### **Тема 5. Тестирование и отладка. (1 часа)**

Типы тестирования программ в процессе разработки. Эксплуатация и сопровождение.

#### **Тема 6. Стандартные технологические процессы (1 часа)**

Основные технологические процессы. Приобретение. Поставка. Разработка. Эксплуатация. Сопровождение. Управление. Усовершенствование. Обучение. Управление. Вспомогательные процессы. Документирование. Управление конфигурацией. Обеспечение качества. Верификация. Аттестация. Совместная оценка. Аудит. Разрешение проблем. Организационные процессы. Управление. Создание инфраструктуры. Усовершенствование. Обучение. Взаимосвязь между процессами.

### **Раздел 4. Технологические подходы (8 часов)**

#### **Тема 1. Основные стадии технологических подходов (1 часа)**

Стадии технологических подходов. Стадия возникновения и исследования идеи. Стадия планирования проекта. Стадия анализа и проектирования. Стадия реализации. Стадия версии разработчика. Стадия оценки жизнеспособности продукта. Стадия альфа - версии. Стадия бета-версии. Стадия версии первой поставки пользователю.

#### **Тема 2. Основные технологические подходы. (1 часа)**

Ранние технологические подходы. Каскадные технологические подходы. Спиральная модель. Каркасные технологические подходы. Рациональный

унифицированный процесс. Синтезирующее программирование.

**Тема 3. Генетические технологические подходы. (1 часа)**

Сборочное (расширяемое) программирование. Конкретизирующее программирование. Подходы на основе формальных преобразований. Технология стерильного цеха.

**Тема 4. Формальные генетические подходы. (1 часа)**

Группа ранних подходов быстрой разработки. Эволюционное прототипирование. Итеративная разработка. Постадийная разработка. Адаптивные технологические подходы. Экстремальное программирование. Адаптивная разработка. Подходы исследовательского программирования. Компьютерный дарвинизм.

**Тема 5. Технологии коллективной разработки. (1 часа)**

Авторская разработка. Коллективная разработка. Бригада главного программиста. Общинная модель разработки.

**Тема 6. Качество программного обеспечения (1 часа)**

Подходы к качеству программного обеспечения. Характеристики качества программного обеспечения. Функциональность. Надежность. Удобство. Эффективность. Сопровождаемость. Добротность. Оценка качества процесса разработки. Модель зрелости процесса разработки программного обеспечения. Стандартизация информационных технологий.

Использование активных форм обучения.

Для данного курса лекции проводятся в форме лекции беседы, лекция консультация, лекция пресс-конференция. Более 60 процентов лекционных занятий проводятся с использованием активных форм обучения

Во время лекции у бакалавров должен быть раздаточный материал, который они должны активно использовать.

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

### **Лабораторные работы (36 часов)**

**Лабораторная работа 1.** Функциональное программирование. Лисп (4 часа).

Тестирование и отладка программ на языке Лисп.

**Лабораторная работа 2.** Логическое программирование. Пролог (8 часов).

Тестирование и отладка программ на языке ПРОЛОГ.

**Лабораторная работа 3.** Программирование на языке С# (8 часов).

Тестирование и отладка программ на языке С#.

**Лабораторная работа 4.** Программирование на языке JAVA (8 часов).

Тестирование и отладка программ на языке JAVA. Программирование с использованием языка JAVA

**Лабораторная работа 5.** Программирование на языке PHP (8 часов).

Тестирование и отладка программ на языке PHP. Программирование с



использованием языка PHP

Разработка программ с элементами объектно-ориентированного программирования

Форма проведения занятий – коллективное занятие с постановкой и решением проблемного задания, закрепляющего знания, полученные на лекции, и навыки, полученные на практических занятиях.

В начале занятия один из обучаемых выполняет задание у доски совместно с преподавателем и другими обучаемыми, в дальнейшем все обучаемыми получают индивидуальные задания.

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

Учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине включает в себя:

- план-график выполнения самостоятельной работы по дисциплине, в том числе примерные нормы времени на выполнение по каждому заданию;
- требования к представлению и оформлению результатов самостоятельной работы;
- критерии оценки выполнения самостоятельной работы.

#### **План-график выполнения самостоятельной работы по дисциплине**

<b>№ п/п</b>	<b>Дата/сроки выполнения</b>	<b>Вид самостоятельной работы</b>	<b>Примерные нормы времени на выполнение</b>	<b>Форма контроля</b>
1	В течение семестра	Подготовка к лабораторным занятиям, изучение литературы	18 часов	Работа на лабораторных занятиях (ПР-6)
2	1-6 неделя семестра	Изучение тем: Основные методологии	6 часов	УО-1 (собеседование/устный опрос)
3	7-12 неделя семестра	Изучение тем: Технологии программирования	10 часов	УО-1 (собеседование/устный опрос)
4	13-18 неделя семестра	Изучение тем: Технологические подходы	36 часов	УО-1 (собеседование/устный опрос)
7	16-18 неделя семестра	Подготовка к экзамену	36 часов	экзамен
Итого:			108 часа	

Методические указания по самостоятельной работе студентов

1. Работа над конспектом лекции: лекции – основной источник информации по многим предметам, позволяющий не только изучить материал, но и получить представление о наличии других источников, сопоставить разные взгляды на основные проблемы данного курса. Лекции предоставляют возможность «интерактивного» обучения, когда есть возможность задавать преподавателю вопросы и получать на них ответы. Поэтому имеет смысл находить время для хотя бы беглого просмотра информации по материалу лекций (учебники, справочники и пр.) и непонятные, а также дискуссионные моменты обсуждать с преподавателем, другими студентами;

- Подготовка к практическому/лабораторному занятию: производится, как правило, с использованием методических пособий, состоит в теоретической подготовке (особенно для семинаров) и выполнении практических заданий (решение задач, ответы на вопросы и т.д.). В данном курсе используются следующие формы практических занятий:

- - лабораторные занятия с использованием вычислительной техники;

2. Доработка конспекта лекции с применением учебника, методической литературы, дополнительной литературы: этот вид самостоятельной работы студентов особенно важен в том случае, когда изучаемый предмет содержит много неоднозначно трактуемых вопросов, проблем. Тогда преподаватель заведомо не может успеть изложить различные точки зрения, и студент должен самостоятельно ознакомиться с ними по имеющейся литературе. Кроме того, рабочая программа предметов предполагает рассмотрение некоторых относительно несложных тем только во время самостоятельных занятий, без чтения лектором;

3. Подбор, изучение, анализ и конспектирование рекомендованной литературы;

4. Самостоятельное изучение отдельных тем, параграфов;

5. Консультации по сложным, непонятным вопросам лекций, семинаров, зачетов;

6. Подготовка к экзамену: один из самых ответственных видов самостоятельной работы, и в то же время возможность сэкономить большое количество времени в период сессии, если эту подготовку начинать заблаговременно. Одно из главных правил – представлять себе общую логику предмета, что достигается проработкой планов лекций, составлением опорных конспектов, схем, таблиц. Фактически основной вид подготовки к экзамену – «свертывание» большого объема информации в компактный вид, а также тренировка в ее «развертывании» (примеры к теории, выведение одних закономерностей из других и т.д.). Владение этими технологиями обеспечивает, пожалуй, более половины успеха. Тем более что преподаватель обычно замечает в течение семестра целенаправленную подготовку такого студента и может поощрить его тем или иным способом. Надо также правильно распределить силы, не только готовясь к самому экзамену, но и позаботившись о допуске к нему (часто это хорошее посещение занятий, выполнение в

назначенный срок практических заданий, активность на семинарах). Наконец, необходимо выяснить условия проведения самого экзаменационного испытания, используя для этой цели, прежде всего консультацию (хотя преподаватель обычно касается этой темы заранее), - количество и характер вопросов, форму проведения (устно или письменно), возможность использования при подготовке различных материалов и пособий (таблицы, схемы, тетради для практических занятий и т.д.).

7. Используется следующая форма научной самостоятельной работы (долговременная): подготовка доклада к конференции: часто студенты для выступлений на научных и научно-практических конференциях используют материалы курсовых работ. Это вполне оправдано, но тогда возникают два вопроса: как обеспечить этим материалам качество научного доклада, который должен решать определенную проблему, иметь новизну и актуальность: как быть первокурсникам, еще не защитивших ни одну курсовую работу. Видимо, каждый студент должен определиться с первой научной темой уже в первые месяцы учебы, что позволит расширить круг интересов, приобретать важные навыки педагога - исследователя, необходимые в дальнейшем совершенствовании в своей профессии. Отсюда следует полезность раннего начала знакомства с литературой, что является вторым этапом любой научной (и методической) работы (первый этап – определение проблемы, темы и гипотезы исследования). Следующий очень важный шаг – правильно спроектировать и осуществить практическую реализацию. Один из самых ответственных этапов – обобщение результатов реализации, что сопровождается анализом качества проекта и анализом затрат на его реализацию. Последнее – формулировка выводов, содержащих данные о решении проблемы предметной области или исследования, положительном или отрицательном (в чем нет ничего страшного) результате. В заключении часто намечают основные пути расширения работы, ее продолжения. Обычно доклад иллюстрируется наглядными презентациями, которые необходимо заранее подготовить.

Таким образом, все виды самостоятельной работы взаимосвязаны и взаимообусловлены, ведущее место занимает учебная самостоятельная деятельность. Все они направлены на повышение как личностных, так и компетентностных качеств будущего специалиста.

#### **IV. КОНТРОЛЬ ДОСТИЖЕНИЯ ЦЕЛЕЙ КУРСА**

№ п/п	Контролируемые разделы / темы дисциплины	Код и наименование индикатора достижения		Оценочные средства	
				текущий контроль	промежуточная аттестация
1.	Разделы 1-3	ПК-3.1	Использует современные технологии программирования, системы	Контрольная работа 1-3, собеседование	Вопросы к экзамену 1 – 163, тестовые вопросы

			баз данных, сетевые технологии, методы, формализмы и стандарты представления диалогов и экранных форм интерфейсов «человек – ЭВМ», инструментальные средства разработки и программно-технологические платформы информационных систем		
		ПК-3.2	Осуществляет изготовление спецификаций программного обеспечения, определяет и устанавливает параметры используемых коммерческих программных пакетов, контролирует качество, создаваемого программного обеспечения, создает базы данных, устанавливает программное обеспечение и осуществляет загрузку баз данных, разработку технической документации,	Контрольная работа 1-3, собеседование	Вопросы к экзамену 1 – 163, тестовые вопросы

			проводит начальное обучение и консультирование пользователей		
		ПК-3.3.	Применяет системные и прикладные программно-технологические платформы, стандарты визуального представления интерфейсов и электронных форматов данных, средства CASE, инструментальные средства разработки информационных систем и программирования	Контрольная работа 1-3, собеседование	Вопросы к экзамену 1 – 63, тестовые вопросы

Типовые контрольные задания, методические материалы, определяющие процедуры оценивания знаний, умений и навыков и (или) опыта деятельности, а также качественные критерии оценивания, которые описывают уровень сформированности компетенций, представлены в разделе VIII.

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

### Основная литература

1. Влацкая И.В. Проектирование и реализация прикладного программного обеспечения [Электронный ресурс]: учебное пособие / И.В. Влацкая, Н.А. Заельская, Н.С. Надточий. — Электрон. текстовые данные. — Оренбург: Оренбургский государственный университет, ЭБС АСВ, 2015. — 119 с. — 978-5-7410-1238-3. <http://www.iprbookshop.ru/54145.html>
2. Сорокин А.А. Объектно-ориентированное программирование [Электронный ресурс] : учебное пособие. Курс лекций / А.А. Сорокин. — Электрон. текстовые данные. — Ставрополь: Северо-Кавказский федеральный университет, 2014. — 174 с. — 2227-8397. <http://www.iprbookshop.ru/63110.html>
3. Котов О.М. Язык C#. Краткое описание и введение в технологии

программирования [Электронный ресурс]: учебное пособие / О.М. Котов. — Электрон. текстовые данные. — Екатеринбург: Уральский федеральный университет, ЭБС АСВ, 2014. — 208 с. — 978-5-7996-1094-4. <http://www.iprbookshop.ru/68524.html>

4. Зыков С.В. Введение в теорию программирования [Электронный ресурс] — 2-е изд. — Электрон. текстовые данные. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 188 с. — 5-9556-0009-4 <http://www.iprbookshop.ru/73675.htm>

5. Александров Э.Э., Афонин В.В. Программирование на языке C в Microsoft Visual Studio 2010 [Электронный ресурс], — 2-е изд. — Электрон. текстовые данные. — М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 570 с. — 2227-8397. <http://www.iprbookshop.ru/73712.htm>

6. Ашарина И.В. Объектно-ориентированное программирование в C++: лекции и упражнения. - "Горячая линия - Телеком" Издательство: 978-5-9912-7001-4 ISBN: 2012: 2-е изд., стереотип. Издание: 320 стр. [http://e.lanbook.com/books/element.php?pl1\\_cid=25&pl1\\_id=5115](http://e.lanbook.com/books/element.php?pl1_cid=25&pl1_id=5115)

7. Бабушкина И.А. Окулов С.М. Практикум по объектно-ориентированному программированию - "Бином. Лаборатория знаний" Издательство: 978-5-9963-0954-2 ISBN: 2012:3-е изд. (эл.) Издание: 366 стр. [http://e.lanbook.com/books/element.php?pl1\\_cid=25&pl1\\_id=8781](http://e.lanbook.com/books/element.php?pl1_cid=25&pl1_id=8781)

8. Цуканова Н.И. Дмитриева Т.А. Теория и практика логического программирования на языке Visual Prolog 7. Учебное пособие для вузов. Москва Издательство: "Горячая линия-Телеком" : ISBN: 978-5-9912-0194-0 2013 - 232 с. [http://e.lanbook.com/books/element.php?pl1\\_cid=25&pl1\\_id=11847](http://e.lanbook.com/books/element.php?pl1_cid=25&pl1_id=11847)

9. Кубенский А.А. Функциональное программирование. С.Петербург: Издательство: СПбНИУ ИТМО (Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики) Ж ISBN: 2010 Год: 251 стр. [http://e.lanbook.com/books/element.php?pl1\\_cid=25&pl1\\_id=40771](http://e.lanbook.com/books/element.php?pl1_cid=25&pl1_id=40771)

#### Дополнительная литература

1. Вендров А.М. Практикум по проектированию программного обеспечения экономических информационных систем - Учебное пособие.-М: Финансы и статистика. 2004.-192с.

2. Вендров А.М. Проектирование программного обеспечения экономических информационных систем. Учебник - М.: Финансы и статистика. 2004.-352с

3. Калянов Г.Н. CASE- технологии. Консалтинг при автоматизации бизнес-процессов. 2-е изд. перераб. и доп. – М.: Горячая линия - Телеком, 2004.- 320с.

4. Леоненков А.В. Объектно-ориентированный анализ и

проектирование с использованием UML и IBM Rational Rose: Учебное пособие  
М.: Интернет - Университет информационных технологий; БИНОМ,  
Лаборатория знаний, 2006. – 320с.

5. Макконнелл С. Сколько стоит программный проект. – М.: «Русская редакция», СПб.: Питер, 2007 – 297с.

6. Л. Басс, П. Клеменс, Р. Кацман Архитектура программного обеспечения на практике – С.Петербург: изд-во Питер, 2006. - 578с.

7. Герман О.В., Герман Ю.О. Программирование на Java и C# для студента + CD С.Петербург: Издательство BHV, ISBN: 5-94157-710-9: 2005 – 272 с.

8. Мартынов Н.Н. C# для начинающих. Москва: Издательство: Кудиц-Пресс: ISBN: 5-91136-023, 2007, - 704с.

9. Беллиназо М., Ватсон К. C#. С.Петербург: Издательство: Питер ISBN: 5-85582-228-1, 2006 - 862 с.

10. Монахов В.С. Язык программирования Java и среда NetBeans. С.Петербург: Издательство: БХВ-Петербург: ISBN 978-5-9775-0671-7; 2011 г. 862

11. Березкина Г.Л. Автоматизированная разработка АСОИУ: Учебно-методический комплекс. - Владивосток. Изд-во ДВГТУ, 2007. – 208с.

12. Березкина Г.Л. Технология программирования: Учебно-методический комплекс.- Владивосток. Изд-во ДВГТУ, 2008. – 112с.

13. Бобровский С.И. Технологии Delphi. Разработка приложений для бизнеса. Учебный курс. – СПб: Питер.- 2007 – 380с.

14. Бобровский С.И. Технологии Borland. Разработка приложений для бизнеса. Учебный курс. – СПб: Питер.- 2007 – 380с.

15. Леонтьев Б.К. Microsoft Office Visio 2003 не для дилетантов: Построение проектов, диаграмм и бизнес – схем в операционной системе Microsoft Windows XP.- М.: ЗАО «Новый издательский дом», 2005. – 384с.

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

1. Интернет-библиотека образовательных изданий: <http://www.iqlib.ru>
2. Интернет университет информационных технологий: <http://www.intuit.ru/>
3. Научная электронная библиотека eLIBRARY.RU: <https://elibrary.ru/defaultx.asp>
4. Информационная система «Единое окно доступа к образовательным ресурсам»: <http://window.edu.ru/window/library>
5. Электронно-библиотечная система Znanium.com (ООО "Знаниум"): <http://znanium.com/>

6. Электронно-библиотечная система «ЮРАЙТ» <https://www.biblio-online.ru/>
7. Электронно-библиотечная система «Консультант студента» <http://www.studentlibrary.ru/>
8. Электронно-библиотечная система «РУКОНТ» <https://lib.rucont.ru/>
9. Электронно-библиотечная система «IPRBOOKS» <http://www.iprbookshop.ru/>

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

Для выполнения практикума и оформления отчёта используется программное обеспечение:

1. MS Word
2. Online трансляторы с языков программирования
3. Visual Studio

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

**Планирование и организация времени, отведенного на изучение дисциплины.** Приступить к освоению дисциплины следует незамедлительно в самом начале учебного семестра. Рекомендуется изучить структуру и основные положения Рабочей программы дисциплины. Обратите внимание, что кроме аудиторной работы (лекции, лабораторные занятия) планируется самостоятельная работа, итоги которой влияют на окончательную оценку по итогам освоения учебной дисциплины. Все задания (аудиторные и самостоятельные) необходимо выполнять и предоставлять на оценку в соответствии с графиком.

В процессе изучения материалов учебного курса предлагаются следующие формы работ: чтение лекций, лабораторные занятия, задания для самостоятельной работы.

*Лекционные занятия* ориентированы на освещение вводных тем в каждый раздел курса и призваны ориентировать студентов в предлагаемом материале, заложить научные и методологические основы для дальнейшей самостоятельной работы студентов.

*Лабораторные занятия* акцентированы на наиболее принципиальных и проблемных вопросах курса и призваны стимулировать выработку практических умений.

Особо значимой для профессиональной подготовки студентов является *самостоятельная работа* по курсу. В ходе этой работы студенты отбирают необходимый материал по изучаемому вопросу и анализируют его. Студентам



необходимо ознакомиться с основными источниками, без которых невозможно полноценное понимание проблематики курса.

Освоение курса способствует развитию навыков обоснованных и самостоятельных оценок фактов и концепций. Поэтому во всех формах контроля знаний, особенно при сдаче зачета, внимание обращается на понимание проблематики курса, на умение практически применять знания и делать выводы.

**Работа с литературой.** Рекомендуется использовать различные возможности работы с литературой: фонды научной библиотеки ДВФУ и электронные библиотеки (<http://www.dvfu.ru/library/>), а также доступные для использования другие научно-библиотечные системы.

**Подготовка к экзамену.** К сдаче экзамена допускаются обучающиеся, выполнившие все задания (лабораторные, самостоятельные), предусмотренные учебной программой дисциплины, посетившие не менее 85% аудиторных занятий.

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

Для обеспечения учебного процесса по дисциплине «Технология программирования» используется следующее материально-техническое обеспечение: компьютеры, операционная система Windows, Интернет, текстовый редактор MS Word, табличный процессор MS Excel, компьютерный класс, LMS Blackboard, LMS Blackboard Collaborate, Visual Studio 2013, персональные компьютеры студентов, а также программное обеспечение, разработанное преподавателем.

## **VIII. ФОНДЫ ОЦЕНОЧНЫХ СРЕДСТВ**

Для дисциплины «Технология программирования» используются следующие оценочные средства:

Устный опрос:

1. Собеседование (УО-1)

Письменные работы:

1. Лабораторная работа (ПР-6)

**Устный опрос**

Устный опрос позволяет оценить знания и кругозор студента, умение логически построить ответ, владение монологической речью и иные коммуникативные навыки.

Обучающая функция состоит в выявлении деталей, которые по каким-то

причинам оказались недостаточно осмысленными в ходе учебных занятий и при подготовке к зачёту.

Собеседование (УО-1) – средство контроля, организованное как специальная беседа преподавателя с обучающимся на темы, связанные с изучаемой дисциплиной, и рассчитанное на выяснение объема знаний обучающегося по определенному разделу, теме, проблеме и т.п.

### **Письменные работы**

Письменный ответ приучает к точности, лаконичности, связности изложения мысли. Письменная проверка используется во всех видах контроля и осуществляется как в аудиторной, так и во внеаудиторной работе.

Лабораторная работа (ПР-6) – средство для закрепления и практического освоения материала по определенному разделу.

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

### **Критерии оценки устных ответов**

- 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.

### **Критерии оценки лабораторных работ**

**Оценка «зачтено»** - Студент выполняет лабораторную работу в полном объеме с соблюдением необходимой последовательности проведения измерений, правильно самостоятельно определяет цель работы; самостоятельно, рационально выбирает необходимое оборудование для получения наиболее точных результатов проводимой работы. Грамотно и логично описывает ход работы, правильно формулирует выводы, точно и аккуратно выполняет все записи, таблицы, рисунки, чертежи, графики, вычисления и т.п., умеет обобщать фактический материал. Допускается два/три недочёта или одна негрубая ошибка и один недочёт. Работа соответствует требованиям и выполнена в срок.

**Оценка «не зачтено»** - Студент выполнил работу не полностью, объём выполненной части не позволяет сделать правильные выводы; не определяет самостоятельно цель работы; в ходе работы допускает одну и более грубые ошибки, которые не может исправить, или неверно производит наблюдения, измерения, вычисления и т.п.; не умеет обобщать фактический материал. Лабораторная работа не выполнена.

### **Критерии оценки выполнения практических занятий**

- 100-86 баллов - если ответ показывает глубокое и систематическое знание всего программного материала и структуры конкретного вопроса, а также основного содержания и новаций лекционного курса по сравнению с учебной литературой. Студент демонстрирует отчетливое и свободное владение концептуально-понятийным аппаратом, научным языком и терминологией соответствующей научной области. Знание основной литературы и знакомство с дополнительно рекомендованной литературой. Логически корректное и убедительное изложение ответа.

- 85-76 - баллов - знание узловых проблем программы и основного содержания лекционного курса; умение пользоваться концептуально-понятийным аппаратом в процессе анализа основных проблем в рамках данной темы; знание важнейших работ из списка рекомендованной литературы. В целом логически корректное, но не всегда точное и аргументированное изложение ответа.

- 75-61 - балл – фрагментарные, поверхностные знания важнейших разделов программы и содержания лекционного курса; затруднения с использованием научно-понятийного аппарата и терминологии учебной дисциплины; неполное знакомство с рекомендованной литературой; частичные затруднения с выполнением предусмотренных программой заданий; стремление логически определенно и последовательно изложить ответ.

- 60-50 баллов – незнание, либо отрывочное представление о данной проблеме в рамках учебно-программного материала; неумение использовать понятийный аппарат; отсутствие логической связи в ответе.

#### **. Методические указания по проведению аттестации студентов**

Промежуточная аттестация студентов по дисциплине «Технология программирования» проводится в соответствии с локальными нормативными актами ДВФУ и является обязательной.

Промежуточная аттестация предусмотрена в устной форме с использованием такого оценочного средства, как устный опрос в форме собеседования.

Устный опрос в форме собеседования (УО-1) по ранее известному кругу вопросов позволяет оценить не только знания, но и кругозор обучающегося, навыки логического построения ответов. В ходе собеседования создаются условия, при которых обучающийся имеет возможность показать владение научной лексикой, продемонстрировать, насколько хорошо он ориентируется в предметной области, связанной с данной дисциплиной.

#### **Критерии оценивания решения тестовых заданий**

По результатам решения тестовых заданий количество правильно решенных заданий переводится в традиционные оценки посредством применения следующей шкалы:

- 86% правильно решенных заданий – «отлично»,
- 75% правильно решенных заданий – «хорошо»,
- 61% правильно решенных заданий – «удовлетворительно»,
- менее 61% - «неудовлетворительно».

#### **Вопросы для промежуточного и итогового контроля**

1. Определение алгоритма.
2. Свойства алгоритма.
3. Понятия кибернетики, связанные с понятием алгоритма.
4. Что такое модель.
5. Что такое моделирование.
6. Что такое жизненный цикл программного обеспечения.
7. Перечислите этапы жизненного цикла программного обеспечения.
8. Что такое проектирование.
9. Что такое проект.
10. Перечислите основные методологии программирования.
11. Дайте определение методологии императивного программирования.
12. Методы и концепции методологии императивного программирования.
13. Вычислительная модель методологии императивного программирования.
14. Основные языки методологии императивного программирования.
15. Перечислите класс задач методологии императивного программирования.
16. Дайте определение методологии ОО программирования.

17. Методы и концепции методологии ОО программирования.
18. Вычислительная модель методологии ОО программирования.
19. Дайте определение методологии функционального программирования.
20. Основные языки методологии функционального программирования.
21. Класс задач методологии функционального программирования.
22. Методы и концепции методологии функционального программирования.
23. Вычислительная модель методологии функционального программирования.
24. Дайте определение методологии логического программирования.
25. Методы и концепции методологии логического программирования.
26. Вычислительная модель методологии логического программирования.
27. Основные языки методологии логического программирования.
28. Класс задач методологии логического программирования
29. Дайте определение методологии программирования в ограничениях.
30. Методы и концепции методологии программирования в ограничениях.
31. Вычислительная модель методологии программирования в ограничениях.
32. Основные языки методологии программирования в ограничениях.
33. Класс задач методологии программирования в ограничениях.
34. Дайте определение методологии структурного императивного программирования.
35. Методы и концепции методологии структурного императивного программирования.
36. Основные языки методологии структурного императивного программирования.
37. Класс задач методологии структурного императивного программирования.
38. Вычислительная модель методологии структурного императивного программирования.
39. Дайте определение методологии императивного параллельного программирования.
40. Методы и концепции методологии императивного параллельного программирования.
41. Вычислительная модель методологии императивного параллельного программирования.
42. Основные языки методологии императивного параллельного программирования.
43. Класс задач методологии императивного параллельного программирования

44. Определение методологии программирования, управляемого потоками данных.
45. Определение методологии доступ - ориентированного программирования.
46. Определение методологии нейросетевого программирования.
47. Определение жизненного цикла программы.
48. Определение технологии программирования.
49. Дать определение процесса.
50. Что такое технологический подход.
51. Что такое стадия жизненного цикла программного обеспечения.
52. Классификация технологических подходов.
53. Классификация технологических процессов.
54. Процессы жизненного цикла, определяемые международным стандартом ISO 12207 [ISO/IEC 12207:1995].
55. Назовите основные направления развития и совершенствования технологических процессов.
56. Составляющие классического процесса «Возникновение и исследование идеи».
57. Составляющие классического процесса «Управление»
58. Перечислите особенности европейского менеджмента.
59. Перечислите особенности американского менеджмента.
60. Перечислите особенности японского менеджмента.
61. Перечислите особенности российского менеджмента.
62. Перечислите методики оценок времени и затрат.
63. Перечислите методы управления проектами.
64. Современные подходы к управлению проектами (практики/навыки).
65. Составляющие классического процесса «Анализ требований и проектирование»
66. Что такое спецификация программы.
67. Средства спецификаций.
68. Составляющие спецификаций
69. Виды спецификаций.
70. Способы представления спецификаций.
71. Что такое архитектура программного продукта
72. Основные структуры, описывающие программную архитектуру.
73. Перечислите архитектурные стили.
74. Проектирование архитектуры (в большом). Структурная методология.
75. Проектирование архитектуры (в большом). Объектно-ориентированная методология.
76. Проектирование архитектуры (в малом). Структурная методология.
77. Проектирование архитектуры (в малом). Объектно-ориентированная методология.
78. Методы анализа и построения спецификаций. Структурная методология.

79. Методы анализа и построения спецификаций. Объектно-ориентированная методология.
80. Основные подходы к ведению анализа и проектирования. Структурная методология.
81. Основные подходы к ведению анализа и проектирования. Объектно-ориентированная методология.
82. Определения стиля программирования.
83. Правило стандартизации стиля.
84. Составляющие стиля программирования.
85. Защитное программирование.
86. Выбор языка программирования.
87. Определение тестирования.
88. Определение отладки.
89. Основные стратегии тестирования.
90. Типичные ошибки.
91. Основные группы тестов.
92. Типы тестирования программ в процессе разработки.
93. Алгоритм отладки программ.
94. Способы доставки программ до пользователя.
95. Основные классы задач, решаемые на этапе эксплуатации и сопровождения.
96. Типы сопровождения программ.
97. Основные технологические процессы. Приобретение (основные действия).
98. Основные технологические процессы. Поставка (основные действия).
99. Основные технологические процессы. Разработка (основные действия).
100. Основные технологические процессы. Эксплуатация (основные действия).
101. Основные технологические процессы. Сопровождение (основные действия).
102. Основные организационные процессы. Управление (основные действия).
103. Основные организационные процессы. Усовершенствование (основные действия).
104. Основные организационные процессы. Обучение (основные действия).
105. Основные организационные процессы. Управление (основные действия).
106. Вспомогательные процессы. Документирование
107. Вспомогательные процессы. Управление конфигурацией
108. Вспомогательные процессы. Обеспечение качества
109. Вспомогательные процессы. Верификация
110. Вспомогательные процессы. Аттестация

111. Вспомогательные процессы. Совместная оценка
112. Вспомогательные процессы. Аудит
113. Вспомогательные процессы. Разрешение проблем
114. Организационные процессы. Управление
115. Организационные процессы. Создание инфраструктуры
116. Организационные процессы. Усовершенствование
117. Организационные процессы. Обучение
118. Организационные процессы. Взаимосвязь между процессами
119. Стадии технологических подходов
120. Стадия возникновения и исследования идеи.
121. Стадия планирования проекта.
122. Стадия анализа и проектирования.
123. Стадия реализации.
124. Стадия версии разработчика.
125. Стадия оценки жизнеспособности продукта.
126. Стадия альфа - версии.
127. Стадия бета-версии.
128. Стадия версии первой поставки пользователю.
129. Основные технологические подходы
130. Ранние технологические подходы
131. Каскадные технологические подходы
132. Спиральная модель
133. Каркасные технологические подходы. Рациональный унифицированный процесс
134. Генетические технологические подходы. Синтезирующее программирование
135. Генетические технологические подходы. Сборочное (расширяемое) программирование
136. Генетические технологические подходы. Конкретизирующее программирование
137. Подходы на основе формальных преобразований. Технология стерильного цеха
138. Подходы на основе формальных преобразований. Формальные генетические подходы
139. Подходы на основе формальных преобразований. Группа ранних подходов быстрой разработки
140. Подходы на основе формальных преобразований. Эволюционное прототипирование
141. Подходы на основе формальных преобразований. Итеративная разработка
142. Подходы на основе формальных преобразований. Постадийная разработка
143. Адаптивные технологические подходы. Экстремальное программирование
144. Адаптивные технологические подходы. Адаптивная разработка



- 145. Подходы исследовательского программирования. Компьютерный дарвинизм
  - 146. Технологии коллективной разработки. Авторская разработка
  - 147. Технологии коллективной разработки. Коллективная разработка
  - 148. Технологии коллективной разработки. Бригада главного программиста
  - 149. Технологии коллективной разработки. Общинная модель разработки
  - 150. Подходы к качеству программного обеспечения
  - 151. Характеристики качества программного обеспечения.
  - 152. Характеристики качества программного обеспечения.
- Функциональность
- 153. Характеристики качества программного обеспечения. Надежность
  - 154. Характеристики качества программного обеспечения. Удобство.
  - 155. Характеристики качества программного обеспечения.
- Эффективность.
- 156. Характеристики качества программного обеспечения.
- Сопровождаемость.
- 157. Характеристики качества программного обеспечения. Добротность
  - 158. Оценка качества процесса разработки
  - 159. Модель зрелости процесса разработки программного обеспечения
  - 160. Стандартизация информационных технологий