

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования «Кабардино-Балкарский государственный университет  
им. Х.М. Бербекова» (КБГУ)

ИНСТИТУТ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА И ЦИФРОВЫХ ТЕХНОЛОГИЙ

КАФЕДРА КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ И ИНФОРМАЦИОННОЙ  
БЕЗОПАСНОСТИ

СОГЛАСОВАНО  
Руководитель образовательной  
программы \_\_\_\_\_ О.Л. Бозиев

УТВЕРЖДАЮ  
Директор ИИИЦТ  
\_\_\_\_\_ А.Х. Шапсигов

« \_\_\_\_ » \_\_\_\_\_ 2022 г.

« \_\_\_\_ » \_\_\_\_\_ 2022 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

«Технологии программирования»

Направление подготовки

**09.03.01 Информатика и вычислительная техника**

Профиль

Автоматизированные системы обработки информации и управления

Квалификация (степень) выпускника  
бакалавр

Форма обучения  
очная

Нальчик 2022

Рабочая программа дисциплины «Технологии программирования» /сост. Е. А. Акбашева – Нальчик: ФГБОУ КБГУ, 2022г. – 35 с.

Рабочая программа предназначена для преподавания дисциплины по выбору студентам очной формы обучения по направлению подготовки 09.03.01 Информатика и вычислительная техника в 4 семестре 2 курса.

Рабочая программа дисциплины составлена с учетом Федерального государственного образовательного стандарта высшего образования по направлению подготовки 09.03.01 Информатика и вычислительная техника, утвержденного приказом Министерства образования и науки Российской Федерации от «19» сентября 2017 г. №929 (зарегистрировано в Минюсте России 10 октября 2017 г. № 48489).

## СОДЕРЖАНИЕ

1. ЦЕЛЬ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ .....	4
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО.....	4
3. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ.....	5
4. СОДЕРЖАНИЕ И СТРУКТУРА ДИСЦИПЛИНЫ .....	6
5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ТЕКУЩЕГО И РУБЕЖНОГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ .....	12
6. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ.....	26
7. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ .....	27
8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ.....	32
9. ЛИСТ ПЕРЕУТВЕРЖДЕНИЯ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ .....	34
ПРИЛОЖЕНИЕ .....	35

## 1. ЦЕЛЬ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ

**Цель** освоения дисциплины:

Систематизация знаний и навыков в области теории, методов, средств и современных технологий разработки программного обеспечения.

**Задачи:**

Изучение методов анализа, проектирования, реализации и тестирования программных систем.

Изучение дисциплины направлено на подготовку специалистов, способных решать проблемы, возникающие при эксплуатации автоматизированных систем обработки информации и управления с учетом области, типов и задач профессиональной деятельности в соответствии с профессиональными стандартами:

- 06.001 – «Программист», утвержденный приказом Министерства труда и социальной защиты Российской Федерации от 18 ноября 2013 г. № 679н (зарегистрирован Министерством юстиции Российской Федерации 18 декабря 2013 г., регистрационный № 30635), с изменением, внесенным приказом Министерства труда и социальной защиты Российской Федерации от 12 декабря 2016 г. № 727н (зарегистрирован Министерством юстиции Российской Федерации 13 января 2017 г., регистрационный № 45230);
- 06.022 – «Системный аналитик», утвержденный приказом Минтруда России от 28.10.2014 № 809н (зарегистрирован в Минюсте России 24.11.2014 № 34882).

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Дисциплина относится к части, формируемой участниками образовательных отношений, дисциплинам по выбору студентов учебного плана по направлению 09.03.01 Информатика и вычислительная техника, профиль «Автоматизированные системы обработки информации и управления» – Б1.В.ДВ.01.01.

Для изучения данной учебной дисциплины необходимы следующие знания, умения и навыки, формируемые предшествующими дисциплинами:

***Из курса «Языки и методы программирования», «Структуры и алгоритмы обработки данных»:***

**Знания:** ядро языка программирования высокого уровня, его синтаксис и семантику; основы проектирования программ: типовые алгоритмы.

**Умения:** описывать разработанные программы посредством блок-схем, тестировать и отлаживать разработанные программы; реализовывать на языке программирования высокого уровня типовые алгоритмы: табуляцию функций, формирование таблиц, нахождение сумм, среднего и т.п.; поиск экстремума, работу с датчиком случайных чисел, ввод и вывод одномерных и двумерных массивов, поиск элементов в массиве, обработку массивов с выводом таблиц, сортировку, ввод и вывод текстов, сравнение фрагментов текста, изменение фрагмента текста по определенному правилу, запись информации в файл, чтение информации из файла, поиск и изменение информации в файле по заданному условию.

**Владения:** приемами работы в среде программирования (составление, отладка и тестирование программ; разработка и использование интерфейсных объектов).

В результате освоения дисциплины обучающийся сможет частично продемонстрировать следующие обобщенные трудовые функции (ОТФ):

- Разработка требований и проектирование программного обеспечения (профессиональный стандарт 06.001 – «Программист», код D, уровень квалификации – 6).
- Концептуальное, функциональное и логическое проектирование систем среднего и крупного масштаба и сложности (профессиональный стандарт 06.022 – «Системный аналитик», код C, уровень квалификации – 6).

### **3. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ**

В совокупности с другими дисциплинами профиля «Автоматизированные системы обработки информации и управления (АСОИиУ)» дисциплина «Технологии программирования» направлена на формирование следующих компетенций в соответствии с ФГОС ВО и ОПОП ВО по направлению подготовки 09.03.01 – Информатика и вычислительная техника (ИиВТ) (уровень бакалавриата):

профессиональные компетенции:

- способен разрабатывать требования и проектировать программное обеспечение (ПКС-1).

#### **Коды и наименования индикаторов достижения компетенции:**

31 ПКС-1.1.

Знать: методики поиска, сбора и обработки информации, метод системного анализа

32 ПКС-1.1.

Знать: формальные методы, технологии и инструменты разработки программного обеспечения; концепции и стратегии проектирования и конструирования программного обеспечения

У1 ПКС-1.2.

Уметь: применять методики поиска, сбора, обработки информации, системный подход для решения поставленных задач и осуществлять критический анализ и синтез информации, полученной из актуальных российских и зарубежных источников

У2 ПКС-1.2.

Уметь: конструировать программное обеспечение, разрабатывать основные программные документы, работать с современными системами программирования

В1 ПКС-1.3.

Владеть: методами поиска, сбора и обработки, критического анализа и синтеза информации, методикой системного подхода для решения поставленных задач

В2 ПКС-1.3.

Владеть: методами конструирования программного обеспечения и проектирования человеко-машинного интерфейса; навыками разработки и отладки программ на алгоритмических языках программирования

В результате освоения дисциплины обучающийся должен:

**Знать:**

- современные методы и средства программирования, СУБД, интегрированные среды, возможности и особенности их применения при разработке программного обеспечения;
- принципы организации и построения баз данных, баз знаний, экспертных систем, пути, методы и средства интеллектуализации информационных систем;

- принцип, модели, средства описания информационных систем и их элементов, объектно-ориентированные модели предметных областей, средства спецификации функциональных задач и проектных решений;
- современные методы и средства разработки информационных систем, тенденции их развития, связь со смежными областями;

**Уметь:**

- организовать процесс разработки ПО;
- грамотно выполнить системный анализ, проектирование, кодирование, отладку и тестирование, документирование и выпуск программного продукта;
- осуществлять коллективную разработку
- оценивать основные критерии качества созданного программного продукта.

**Владеть:**

- навыками проектирования, кодирования и отладки разрабатываемого ПО;
- приемами тестирования и документирования ПО;
- приемами работы при осуществлении коллективной разработки.

#### 4. СОДЕРЖАНИЕ И СТРУКТУРА ДИСЦИПЛИНЫ

**Таблица 1**

**Содержание дисциплины**

№ раздела	Наименование раздела	Содержание раздела	Код контролируемой компетенции (или ее части)	Форма текущего контроля
1	Введение в дисциплину «Технологии программирования». Основные понятия и определения.	Введение в дисциплину «Технологии программирования». Основные понятия и определения.	ПКС-1	ТК, К, Т
2	Этапы решения задач на ЭВМ.	Этапы решения задач на ЭВМ.	ПКС-1	ТК, К, Т
3	Жизненный цикл программ.	Жизненный цикл программы. Модели жизненного цикла ПО. Изменение жизненного цикла программного обеспечения при использовании CASE-технологий.	ПКС-1	ТК, К, Т
4	Метрология и качество программного обеспечения.	Метрология и качество программного обеспечения. Модели качества ПО. Метрики качества программного обеспечения. Измерения и оценка качества ПО. Стандартный метод	ПКС-1	ТК, К, Т

		оценки значений показателей качества. Управление качеством ПС.		
5	Диалоговые программы. Дружественность.	Диалоговые программы. Дружественность.	ПКС-1	ТК, К, Т
6	Спецификация программных средств.	Спецификация программных средств. Определение требований к программному средству. Спецификация программных средств. Спецификация качества программного средства. Функциональная спецификация. Методы контроля внешнего описания программного средства.	ПКС-1	ТК, К, Т
7	Способы записи алгоритмов.	Способы записи алгоритмов. Представление основных структур алгоритмов. Псевдокоды. Flow-формы. Диаграммы Насси-Шнейдермана.	ПКС-1	ТК, К, Т
8	Структура данных программы.	Структура данных программы. Стандартные типы данных. Операции над структурами данных. Общая классификация логических структур данных. Простые и интегрированные структуры данных. Типы данных, определяемые пользователем. Связные динамические структуры данных. Классификация видов оперативных структур данных по их логическому устройству. Файловые структуры. Физическая организация файлов. Логическая организация файлов. Документирование файлов. Программирование рекурсивных алгоритмов.	ПКС-1	ТК, К, Т, ЛР
9	Способы конструирования программ.	Способы конструирования программ. Модульные программы. Модули и их свойства. Сцепление модулей. Связность модулей. Нисходящая и восходящая разработка программного обеспечения. Программирование «с защитой от ошибок». Сквозной структурный контроль.	ПКС-1	ТК, К, Т
10	Основные подходы программирования.	Основные подходы программирования. «Стихийное программирование». Структурный подход к программированию. Объектный подход к программированию. Компонентный подход и CASE-технологии.	ПКС-1	ТК, К, Т, ЛР
11	Императивное и декларативное программирование.	Императивное и декларативное программирование. Процедурное программирование. Функциональное	ПКС-1	ТК, К, Т

		программирование. Логическое программирование. Объектно-ориентированное программирование.		
12	Методы, технология и инструментальные средства при структурном подходе к программированию.	Спецификация программного обеспечения при структурном подходе. Диаграммы переходов состояний. Функциональные диаграммы. Диаграммы потоков данных. Моделирование управляющих процессов с помощью диаграмм потоков данных. Структуры данных и диаграммы отношений компонентов данных. Диаграммы Джексона. Скобочные диаграммы Орра. Сетевая модель данных. Математические модели задач, разработка или выбор методов решения.	ПКС-1	ТК, К, Т, ЛР
13	Проектирование программного обеспечения при структурном подходе к программированию.	Разработка структурной и функциональной схем. Структурная схема разрабатываемого программного обеспечения. Функциональная схема. Метод пошаговой детализации для проектирования структуры программного обеспечения. Структурные карты Константайна. Проектирование структур данных. Представление данных в оперативной памяти. Представление данных во внешней памяти. Проектирование программного обеспечения, основанное на декомпозиции данных. Методика Джексона. CASE-технологии, основанные на структурных методологиях анализа и проектирования.	ПКС-1	ТК, К, Т, ЛР
14	Методы, технология и инструментальные средства при объектном подходе к программированию.	UML – стандартный язык описания разработки программных продуктов с использованием объектного подхода. Определение «вариантов использования». Диаграммы вариантов использования. Построение концептуальной модели предметной области. Диаграмма последовательностей системы. Системные события и операции. Диаграммы деятельности.	ПКС-1	ТК, К, Т, ЛР
15	Проектирование программного обеспечения при объектном подходе.	Разработка структуры ПО при объектном подходе. Определение отношений между объектами. Диаграммы последовательностей этапа проектирования. Диаграммы	ПКС-1	ТК, К, Т, ЛР



		кооперации. Уточнение отношений классов. Интерфейсы. Проектирование классов. Проектирование методов класса. Компоновка программных компонентов. Проектирование размещения программных компонентов для распределенных программных систем. Особенность спиральной модели разработки. Реорганизация проекта.		
16	Основы доказательства правильности программ.	Основы доказательства правильности программ. Методы доказательства правильности программ. Метод индуктивных утверждений Флойда. Метод Хоора.	ПКС-1	ТК, К, Т
17	Тестирование программных продуктов.	Виды контроля качества разрабатываемого ПО. Формирование тестовых наборов. Ручной контроль программного обеспечения. Инспекции исходного текста. Сквозные просмотры. Оценка программ. Структурное тестирование. Функциональное тестирование. Тестирование модулей и комплексное тестирование. Восходящее тестирование. Нисходящее тестирование. Оценочное тестирование.	ПКС-1	ТК, К, Т, ЛР
18	Отладка программного обеспечения.	Классификация ошибок. Методы отладки ПО. Метод ручного тестирования. Метод индукции. Метод дедукции. Метод обратного прослеживания. Методы и средства получения дополнительной информации.	ПКС-1	ТК, К, Т, ЛР
19	Документирование и стандартизация.	Виды программных документов. Основные правила оформления программной документации.	ПКС-1	ТК, К, Т, ЛР
20	Технологический цикл разработки программных систем.	Основные инженерные подходы к созданию программ. Классификация технологических подходов к созданию программ.	ПКС-1	ТК, К, Т
21	Автоматизация проектирования программных продуктов.	Особенности и компоненты CASE-средств. Объектно-ориентированные CASE-средства анализа и проектирования. Структурные CASE-средства. CASE-средства компании IBM Rational Software.	ПКС-1	ТК, К, Т
22	Принципы построения, структура и технология	Системы автоматизированного проектирования и их место среди других автоматизированных систем.	ПКС-1	ТК, К, Т

	использования САПР ПО.	Структура САПР. Разновидности САПР. Понятие о CALS-технологиях.		
--	---------------------------	--	--	--

Таблица 2

## Структура дисциплины

Вид работы	Трудоемкость, часы
	3 семестр
<b>Общая трудоемкость (в зачетных единицах)</b>	<b>4</b>
<b>Контактная работа (в часах):</b>	68
Лекции (Л)	34
Практические занятия (ПЗ)	—
Семинарские занятия (СЗ)	—
Лабораторные работы (ЛР)	34
<b>Самостоятельная работа (в часах ), в том числе контактная работа:</b>	67
Курсовой проект (КП), курсовая работа (КР)	—
Расчетно-графическое задание (РГЗ)	—
Реферат (Р)	—
Эссе (Э)	—
Самостоятельное изучение разделов	67
Контрольная работа (К)	—
<b>Подготовка и прохождение промежуточной аттестации</b>	9
Вид промежуточной аттестации	зачет

Таблица 3

## Лекции

№ раз-дела	Наименование разделов
1	Введение в дисциплину «Технологии программирования». Основные понятия и определения.
2	Этапы решения задач на ЭВМ.
3	Жизненный цикл программ.
4	Метрология и качество программного обеспечения.
5	Диалоговые программы. Дружественность.
6	Спецификация программных средств.
7	Способы записи алгоритмов.
8	Структура данных программы.
9	Способы конструирования программ.
10	Основные подходы программирования.

11	Императивное и декларативное программирование.
12	Методы, технология и инструментальные средства при структурном подходе к программированию.
13	Проектирование программного обеспечения при структурном подходе к программированию.
14	Методы, технология и инструментальные средства при объектном подходе к программированию.
15	Проектирование программного обеспечения при объектном подходе.
16	Основы доказательства правильности программ.
17	Тестирование программных продуктов.
18	Отладка программного обеспечения.
19	Документирование и стандартизация.
20	Технологический цикл разработки программных систем.
21	Автоматизация проектирования программных продуктов.
22	Принципы построения, структура и технология использования САПР ПО.

Таблица 4

#### Лабораторные работы

№ занятия	№ раздела	Наименование лабораторных работ
1	8	Структура данных программ. Динамические структуры данных.
2	10	Объектно-ориентированное программирование.
3	12	Этапы разработки программного обеспечения при структурном подходе к программированию. Стадия «Техническое задание»
4	12	Структурный подход к программированию. Стадия «Эскизный проект».
5	13	Структурный подход к программированию. Стадия «Технический проект».
6	13	Этапы разработки программного обеспечения. Стадия «Реализация».
7	17	Тестирование программ методами «белого ящика».
8	10	Использование технологий OLE, COM и ActiveX.
9	14-15	Проектирование программной системы при объектном подходе к программированию.

#### Практические занятия (семинары)

Не предусмотрено

**Курсовой проект (курсовая работа)****Не предусмотрено****Таблица 5****Самостоятельное изучение разделов дисциплины**

№ раздела	Вопросы, выносимые на самостоятельное изучение
1	Стандарты проектирования.
2	Диаграммы «сущность-связь». Нотация ERD. Основные понятия.
3	Методологии IDEF1 и IDEF1X.
4	Матрицы списков событий. Диаграммы структур данных.
5	Диаграммы процессов системы. Принципы построения, примеры.
6	Технологии COM и DCOM.
7	Технология OLE-automation.
8	Технология ActiveX.
9	Технологии MTS и MIDAS.
10	Технология CORBA.
11	Case-средства компании IBM Rational Software, средство визуального моделирования Rational Rose.
12	Системы автоматизированного проектирования и их место среди других автоматизированных систем.
13	Структура САПР.
14	Разновидности САПР.
15	Понятие о CALS-технологиях.

## **5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ТЕКУЩЕГО И РУБЕЖНОГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ**

Формы контроля текущих, рубежных и промежуточных знаний студентов по дисциплине определяются в соответствии с учебным планом образовательной программы и в соответствии с действующим Положением о балльно-рейтинговой системе оценки успеваемости студентов КБГУ.

От обучающихся требуется посещение занятий, выполнение лабораторных работ, знакомство с рекомендованной литературой.

При аттестации обучающихся оценивается качество работы на занятиях (умение вести дискуссию, способность четко и ёмко формулировать свои мысли), уровень подготовки к самостоятельной деятельности, качество выполнения заданий (презентаций, докладов, выполнение лабораторных работ и др.).

Конечными результатами освоения программы дисциплины являются сформированные когнитивные дескрипторы «знать», «уметь», «владеть», расписанные по отдельным компетенциям. Формирование этих дескрипторов происходит в течение всего семестра по этапам в рамках различного вида занятий и самостоятельной работы.

В ходе изучения дисциплины предусматриваются текущий, рубежный контроль и промежуточная аттестация.

### 5.1. Оценочные материалы для текущего контроля

Цель текущего контроля – оценка результатов работы в семестре и обеспечение своевременной обратной связи, для коррекции обучения, активизации самостоятельной работы обучающегося. Объектом текущего контроля являются конкретизированные результаты обучения (учебные достижения) по дисциплине.

Текущий контроль успеваемости обеспечивает оценивание хода освоения дисциплины «Технологии программирования» и включает: отчет по результатам выполнения лабораторных работ, самостоятельное выполнение заданий с отчетом (защитой) в установленный срок.

Оценка качества подготовки на основании выполненных заданий ведется преподавателем (с обсуждением результатов), баллы начисляются в зависимости от сложности задания.

#### Критерии формирования оценок (оценивания) устного опроса

Устный опрос является одним из основных способов учёта знаний обучающегося по дисциплине «Технологии программирования». Развёрнутый ответ должен представлять собой связное, логически последовательное сообщение на заданную тему, показывать его умение применять определения.

В результате устного опроса знания, обучающегося оцениваются по следующей шкале:

Таблица 6

3 балла	2 балла	1 балл	0 баллов
<p>ставится, если обучающийся:</p> <p>1) полно излагает изученный материал, даёт правильное определенное экономических понятий;</p> <p>2) обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только по учебнику, но и самостоятельно составленные;</p> <p>3) излагает материал последовательно и правильно с точки зрения норм литературного языка.</p>	<p>ставится, если обучающийся даёт ответ, удовлетворяющий тем же требованиям, что и для балла «1», но допускает 1-2 ошибки, которые сам же исправляет, и 1-2 недочёта в последовательности и языковом оформлении излагаемого.</p>	<p>ставится, если обучающийся обнаруживает знание и понимание основных положений данной темы, но:</p> <p>1) излагает материал неполно и допускает неточности в определении понятий;</p> <p>2) не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры;</p> <p>3) излагает материал непоследовательно и допускает ошибки в языковом оформлении излагаемого.</p>	<p>ставится, если обучающийся обнаруживает незнание большей части соответствующего раздела изучаемого материала, допускает ошибки в формулировке.</p>

Баллы «1», «2», «3» могут ставиться не только за единовременный ответ, но и за рассредоточенный во времени, т.е. за сумму ответов, данных на протяжении занятия. начисляются в зависимости от сложности задания.

### **5.1.1. Оценочные материалы для самостоятельной работы обучающегося**

Рабочая программа предусматривает проведение лекционных, лабораторных занятий, а также самостоятельную работу обучающихся. В ФГБОУ ВО «Кабардино-Балкарский государственный университет» действует балльно-рейтинговая система оценки учебных достижений обучающихся по образовательным программам, реализуемым на основании федеральных государственных образовательных стандартов. Балльно-рейтинговая система оценки знаний является одной из составляющих системы управления качеством образовательной деятельности в университете.

#### **Темы для самостоятельной работы**

1. Стандарты проектирования.
2. Диаграммы «сущность-связь». Нотация ERD. Основные понятия.
3. Методологии IDEF1 и IDEF1X.
4. Матрицы списков событий. Диаграммы структур данных.
5. Диаграммы процессов системы. Принципы построения, примеры.
6. Технологии COM и DCOM.
7. Технология OLE-automation.
8. Технология ActiveX.
9. Технологии MTS и MIDAS.
10. Технология CORBA.
11. Case-средства компании IBM Rational Software, средство визуального моделирования Rational Rose.
12. Системы автоматизированного проектирования и их место среди других автоматизированных систем.
13. Структура САПР.
14. Разновидности САПР.
15. Понятие о CALS-технологиях.

#### **Критерии формирования оценок по заданиям для самостоятельной работы студента**

*«Отлично»* (5 баллов) – обучающийся показал глубокие знания материала по поставленным вопросам, грамотно, логично его излагает, структурировал и детализировал информацию, избегая простого повторения информации из текста, информация представлена в переработанном виде. Свободно использует необходимые формулы при решении задач;

*«Хорошо»* (4 балла) – обучающийся твердо знает материал, грамотно его излагает, не допускает существенных неточностей в процессе решения задач;

*«Удовлетворительно»* (3 балла) – обучающийся имеет знания основного материала по поставленным вопросам, но не усвоил его деталей, допускает отдельные неточности при решении задач;

*«Неудовлетворительно»* (менее 3 баллов) – обучающийся допускает грубые ошибки в ответе на поставленные вопросы и при решении задач

## 5.2. Оценочные материалы для рубежного контроля

Рубежный контроль осуществляется по более или менее самостоятельным разделам – учебным модулям курса и проводится по окончании изучения материала модуля в заранее установленное время. Рубежный контроль проводится с целью определения качества усвоения материала учебного модуля в целом. В течение семестра проводится три таких контрольных мероприятия по графику.

В качестве форм рубежного контроля используется тестирование (письменное или компьютерное), проведение коллоквиума.

Рубежный контроль освоения студентом дисциплины осуществляется в рамках балльно-рейтинговой системы. Распределение баллов в соответствии с действующим Положением о балльно-рейтинговой системе оценки успеваемости студентов КБГУ приведено в таблице 7.

Таблица 7

Распределение баллов в соответствии с действующим Положением о балльно-рейтинговой системе

№ рейтинговой точки	Коллоквиум	Лаб.практикум	Посещаемость	Тестирование	Итого
1	7	8	3	5	23
2	7	8	3	5	23
3	7	8	4	5	24

Таблица 8

### Критерии оценки

Вид мероприятия	Критерии оценки	Баллы
Коллоквиум (устный опрос по теме)	- ясность, четкость и доказательность изложения ответов на вопросы; - владение специальными терминами; - системность знаний по тематике	0-21 балл
Лабораторное занятие	- понимание цели и задач работы - выполнение заданий и обработка результатов - отчет и защита лабораторной работы	0-24 балла
Компьютерное тестирование по разделам дисциплины	Результаты тестирования (Количество баллов = $5 \cdot \varphi$ , $\varphi$ - доля правильно отвеченных тестов по теме).	0-15 баллов
Посещение занятий	При более 3 пропусках без уважительной причины занятий аннулируются баллы	0-10 баллов
Зачет	ясность, четкость и доказательность изложения ответов на вопросы; - владение специальными терминами; - системность знаний по тематике дисциплины в целом	0-30 баллов
Итоговая оценка		0-100 баллов

### 5.2.1. Оценочные материалы для проведения коллоквиума

#### Перечень вопросов, выносимых на рейтинговый контроль первой точки (контролируемые компетенции ПКС-1):

1. Основные понятия и определения дисциплины «Технологии программирования».
2. Этапы решения задач на ЭВМ.
3. Определение жизненного цикла программного средства. Состав жизненного цикла. Стандарты, регламентирующие состав жизненного цикла.

4. Жизненный цикл программного средства. Структура жизненного цикла в соответствии со стандартом ISO/IEC 12207.
5. Модели жизненного цикла. Каскадная модель, ее недостатки и достоинства.
6. Модели жизненного цикла. Модель с промежуточным контролем, ее недостатки и достоинства.
7. Модели жизненного цикла. Спиральная модель, ее недостатки и достоинства.
8. Изменение жизненного цикла программного обеспечения при использовании CASE-технологий.
9. Метрология программного обеспечения. Концепция качества программного обеспечения.
10. Модели качества программного обеспечения.
11. Метрики качества программного обеспечения. Типы метрик.
12. Измерения и оценка качества ПО. Стандартный метод оценки значений показателей качества.
13. Диалог. Диалоговые программы. Типы диалога.
14. Диалог. Типы диалога. Формы диалога.
15. Фразовая форма диалога, ее недостатки и достоинства.
16. Директивная форма диалога, ее недостатки и достоинства.
17. Табличная форма диалога, ее недостатки и достоинства.
18. Понятие спецификации программных средств, внешнее описание ПС.
19. Определение требований к программному средству.
20. Способы определения требований к программному средству.
21. Спецификация качества программного средства.
22. Функциональная спецификация программного средства.
23. Методы контроля внешнего описания программного средства.
24. Способы записи алгоритмов.
25. Представление основных структур алгоритмов. ГОСТ 19.701-90.
26. Псевдокод. Определение, основные обозначения, привести пример.
27. Flow-формы. Определение, основные обозначения, привести пример.
28. Диаграммы Насси-Шнейдермана. Определение, основные обозначения, привести пример.
29. Информационные системы. Определение, назначение, применение.
30. Техническое задание. Основные разделы технического задания. ГОСТ 19.201-78.
31. Принципиальные решения начальных этапов проектирования.
32. Понятие архитектуры программного обеспечения.
33. Типы пользовательского интерфейса.
34. Технологии работы с документами.
35. Принципиальные решения начальных этапов проектирования. Подход к разработке.
36. Группы языков программирования.
37. Специализированные языки разработчика и специализированные языки пользователя.
38. Языки высокого уровня и языки низкого уровня.
39. Среда программирования. Понятие визуальной среды программирования.
40. Методология функционального моделирования SADT.
41. Состав функциональной модели.
42. Моделирование потоков данных.
43. Иерархия диаграмм потоков данных, ее построение.
44. Моделирование данных. Метод Баркера.
45. Методология IDEF1.
46. Определение понятий: методология, технология и инструментальные средства проектирования.
47. Понятие структурного подхода к разработке ИС. Сущность подхода. Декомпозиция.
48. Состав функциональной модели. Уровни детализации. Блоки и дуги.



49. Типы связей на функциональной диаграмме.
50. Диаграммы структур данных. Принципы построения, примеры.

**Перечень вопросов, выносимых на рейтинговый контроль второй точки  
(контролируемые компетенции ПКС-1):**

1. Структура данных. Классификация структур данных.
2. Простые структуры данных.
3. Статические структуры данных.
4. Полустатические структуры данных.
5. Динамические структуры данных.
6. Файловые структуры. Физическая организация файлов.
7. Логическая организация файлов.
8. Документирование файлов.
9. Программирование рекурсивных алгоритмов.
10. Модульные программы.
11. Модули.
12. Сцепление модулей.
13. Связность модулей.
14. Библиотеки ресурсов.
15. Восходящий подход к разработке программного обеспечения.
16. Нисходящий подход к разработке программного обеспечения.
17. Программирование «с защитой от ошибок».
18. Способы проявления ошибок.
19. Проверки правильности выполнения операций ввода-вывода.
20. Проверка допустимости промежуточных результатов.
21. Обработка исключений.
22. Сквозной структурный контроль.
23. Основные подходы программирования. «Стихийное» программирование.
24. Основные подходы программирования. Структурный подход к программированию.
25. Основные подходы программирования. Объектный подход к программированию.
26. Основные подходы программирования. Компонентный подход и CASE-технологии.
27. Процедурное программирование.
28. Императивные (процедурные) языки.
29. Функциональное программирование.
30. Декларативные языки программирования.
31. Объектно-ориентированное программирование.
32. Объектно-ориентированные языки.
33. Сущность объектно-ориентированного подхода.
34. Унифицированный язык моделирования UML.
35. Варианты использования.
36. Диаграмма вариантов использования.
37. Диаграммы классов.
38. Механизм пакетов.
39. Диаграмма пакетов.
40. Диаграммы взаимодействия.
41. Диаграммы последовательности.
42. Кооперативные диаграммы.
43. Диаграммы состояний.
44. Диаграммы деятельности.
45. Диаграммы компонентов.
46. Диаграммы размещения.
47. Сущность объектно-ориентированного подхода, его отличие от структурного подхода.
48. Основные элементы объектной модели.

49. Понятие языка моделирования, нотации, процесса.
50. Набор диаграмм, предлагаемый UML.
51. Понятие варианта использования. Диаграммы вариантов использования.
52. Действующие лица, варианты использования и связи между ними.
53. Связи «использование» и «расширение».
54. Диаграммы классов. Основные понятия и определения.
55. Аспекты построения диаграмм классов (концептуальный, спецификации, реализации).
56. Связи между экземплярами классов для различных аспектов построения диаграмм классов.
57. Понятие атрибута.
58. Понятие операции.
59. Определение обобщения.
60. Ограничения на диаграмме классов.
61. Стереотипы в UML.
62. Классификация, типы и супертипы в UML.
63. Агрегация и композиция.
64. Класс ассоциаций.
65. Механизм пакетов в UML. Диаграмма пакетов.
66. Диаграммы взаимодействия. Типы диаграмм взаимодействия.
67. Диаграммы последовательности. Особенности построения диаграмм последовательности.
68. Параллельные процессы, активизации и асинхронные сообщения на диаграмме последовательности.
69. Кооперативные диаграммы. Особенности построения.
70. Диаграмма состояний в языке UML. Состояния и суперсостояния.
71. Диаграмма деятельностей. Построение диаграммы деятельностей.
72. Диаграмма компонентов, ее назначение.
73. Диаграмма размещения, ее назначение.
74. Взаимосвязь между объектно-ориентированным и структурным подходами.
75. Недостатки и преимущества объектно-ориентированного подхода.

**Перечень вопросов, выносимых на рейтинговый контроль третьей точки  
(контролируемые компетенции ПКС-1):**

1. Виды контроля качества разрабатываемого программного обеспечения.
2. Формирование тестовых наборов, основные подходы.
3. Ручной контроль программного обеспечения, методы ручного контроля.
4. Структурное тестирование, критерии формирования тестовых наборов.
5. Функциональное тестирование, методы формирования тестовых наборов.
6. Тестирование модулей и комплексное тестирование.
7. Оценочное тестирование.
8. Отладка программного обеспечения.
9. Классификация ошибок программного обеспечения.
10. Методы отладки программного обеспечения.
11. Методы и средства получения дополнительной информации об ошибках.
12. Общая методика отладки программного обеспечения.
13. Документирование и стандартизация.
14. Виды программных документов.
15. Основные правила оформления программной документации.
16. Основные инженерные подходы к созданию программ.
17. Классификация технологических подходов к созданию программ.

18. Классификация технологических подходов к созданию программ, подходы со слабой формализацией.
19. Классификация технологических подходов к созданию программ, строгие каскадные подходы.
20. Классификация технологических подходов к созданию программ, строгие каркасные подходы.
21. Классификация технологических подходов к созданию программ, генетические подходы.
22. Классификация технологических подходов к созданию программ, подходы на основе формальных преобразований.
23. Классификация технологических подходов к созданию программ, ранние подходы быстрой разработки.
24. Классификация технологических подходов к созданию программ, адаптивные технологические подходы.
25. Классификация технологических подходов к созданию программ, подходы исследовательского программирования.
26. Особенности и компоненты CASE-средств.
27. Объектно-ориентированные CASE-средства анализа и проектирования.
28. Структурные CASE-средства анализа и проектирования.

#### **Критерии формирования оценок по контрольным точкам (коллоквиум)**

4 балла ставится за работу, выполненную полностью без ошибок и недочетов; обучающийся демонстрирует знание теоретического и практического материала по теме практической работы, решено 100% задач;

3 балла ставится за работу, выполненную полностью, но при наличии в ней не более одной негрубой ошибки и одного недочета, не более трех недочетов. Обучающийся демонстрирует знание теоретического и практического материала по теме практической работы, допуская незначительные неточности при решении задач, решено 70% задач;

2 балла ставится за работу, если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой. Обучающийся затрудняется с правильной оценкой предложенной задачи, дает неполный ответ, решено 55% задач

1 балл ставится за работу, если число ошибок и недочетов превысило норму или правильно выполнено менее 2/3 всей работы. Обучающийся дает неверную оценку ситуации, решено менее 50 % задач.

#### **5.2.2. Оценочные материалы: тестирование**

Тестирование обучающихся проводится в онлайн-режиме согласно расписанию в ЭИОС [open.kbsu.ru](http://open.kbsu.ru)

#### **Примерные тестовые задания (контролируемые компетенции ПКС-1)**

**Полный перечень тестовых заданий представлен в ЭОИС**

##### **V1: Основные понятия и определения**

I:

S: Совокупность всей информации, данных и программ, которые обрабатываются компьютерными системами

+: программное обеспечение

-: программный продукт

-: программное изделие

-: программная система

I:

S:Система инженерных принципов для создания программного обеспечения

+ : технология разработки ПО

- : технологическая операция

- : технический проект

I:

S:Совокупность методов и средств, используемых в процессе разработки программного обеспечения

+ : технология программирования

- : технологическая операция

- : программа

I:

**V1:Жизненный цикл программного обеспечения**

**V2:Жизненный цикл программного обеспечения**

I:

S:Совокупность процессов, отражающая различные состояния программного средства (ПС), начиная с момента принятия решения о необходимости его создания и заканчивая его полным изъятием из эксплуатации

+ : жизненный цикл ПС

- : технологическая операция

- : проектная операция

- : технология

I:

S:Состав процессов жизненного цикла регламентируется международным стандартом

- : ГОСТ 19.101-77

+ : ISO/IEC 12207: 1995

- : ISO/IEC 13207

- : ГОСТ 19.404-79

I:

S:Создание текстов программ на языках программирования, их отладка с тестированием ПС

+ : кодирование

- : разработка

- : сопровождение

- : эксплуатация

**V2:Модели жизненного цикла**

I:

S:Структура, определяющая последовательность выполнения стадий, и их взаимосвязи на протяжении жизненного цикла

+ : модель жизненного цикла

- : аспект жизненного цикла

- : этап жизненного цикла

- : технология

I:

S:Схема разработки программного обеспечения, предполагающая, что переход на следующую стадию осуществляется после того, как полностью будут завершены проектные операции предыдущей стадии и получены все исходные данные для следующей стадии

+ : каскадная модель

- : модель с промежуточным контролем

- : итерационная модель

- : спиральная модель

**V1:Метрология и качество программного обеспечения**

**V2:Качество ПО**

I:

S:Дисциплина, изучающая проблемы оценивания метрических характеристик качества ПО на этапах от разработки спецификаций до завершения отладки и тестирования программного продукта – это ###

+: метрология;

I:

S:Международный стандарт, в соответствии с которым проводится окончательная оценка качества программного обеспечения

-: ISO 9126-01

+: ISO 15504-98

-: ISO/IEC 12207: 1995

-: ISO/IEC 13207

I:

S:Анализ и проверка различных представлений системы и ПО (спецификаций, архитектурных схем, диаграмм, исходного кода и др.), выполняющийся на всех этапах ЖЦ разработки ПО

+: инспекция ПО

-: верификация

-: валидация

-: аттестация

I:

S:Процесс обеспечения правильной реализации программного обеспечения (в соответствии со спецификациями), выполняющийся на протяжении всего жизненного цикла

-: инспекция ПО

+: верификация

-: валидация

-: аттестация

**V1:Спецификация программных средств**

I:

S:Внешнее описание программного средства состоит из

+: спецификация качества

+: функциональная спецификация

-: функция защиты от несанкционированного доступа

-: требования к ПС

I:

S:Мера, характеризующая приемлемость величины погрешности в выдаваемых программами ПС результатах с точки зрения предполагаемого их использования

+: точность

-: П-документированность

-: автономность

-: завершенность

I:

S:Свойство, характеризующее способность ПС выполнять предписанные функции без помощи или поддержки других компонент программного обеспечения

+: автономность

-: С-документированность

-: завершенность

-: точность

I:

S:Мера, характеризующая способность ПС выполнять возложенные на него функции при определенных ограничениях на используемую память

- : временная эффективность
- +: эффективность по памяти
- : эффективность по устройствам
- : П-документированность

### **V1: Структура данных программы**

I:

S: Структура данных, состоящая из фиксированного количества компонентов одного типа, называется:

- +: Массив
- : Запись
- : Объединение
- : Перечислимый тип

I:

S: Структуру данных, состоящую из фиксированного количества компонентов, которые могут быть различных типов, называют:

- +: Запись (структура)
- : Объединение
- : Множество
- : Список

### **V1: Способы конструирования программ**

#### **V2: Модули**

I:

S: Автономно компилируемая программная единица называется ###

- +: модуль

I:

S: Мера взаимозависимости модулей, которая определяет, насколько хорошо модули отделены друг от друга – это ###

- +: сцепление

I:

S: Сцепление, предполагающее, что модули обмениваются данными, представленными скалярными значениями

- +: по данным
- : по образцу
- : по управлению
- : по содержимому

I:

S: Сцепление, предполагающее, что один модуль посылает другому некоторый информационный объект (флаг), предназначенный для управления внутренней логикой модуля

- : по данным
- : по образцу
- +: по управлению
- : по содержимому

### **V1: Основные подходы программирования**

I:

S: В основе структурного подхода лежит ### сложных систем с целью последующей реализации в виде отдельных небольших подпрограмм

- +: декомпозиция

I:

S: Подход, предполагающий построение программного обеспечения из отдельных компонентов физически отдельно существующих частей программного обеспечения,

которые взаимодействуют между собой через стандартизованные двоичные интерфейсы, называют ###

+: компонентный

#### **V1:Императивное и декларативное программирование**

I:

S:Языки программирования, управляемые командами или операторами языка это:

+: Императивные языки

-: Языки функционального программирования

-: Декларативные языки

-: Объектно-ориентированные языки

I:

S:Языки, в которых вычисления производятся путем применения функций к заданному набору данных это:

-: Императивные языки

+: Языки функционального программирования

-: Декларативные языки

-: Объектно-ориентированные языки

### **Критерии формирования оценок по тестовым заданиям**

Выполнение тестирования оценивается согласно проценту правильных ответов. Максимально возможное количество баллов за тестирование – 5.

### **5.3. Оценочные материалы для промежуточной аттестации**

Целью промежуточных аттестаций по дисциплине является оценка качества освоения дисциплины обучающимися.

Промежуточная аттестация предназначена для объективного подтверждения и оценивания достигнутых результатов обучения после завершения изучения дисциплины. Осуществляется в конце семестра и представляет собой итоговую оценку знаний по дисциплине «Технологии программирования» в виде проведения зачета.

Промежуточная аттестация может проводиться в устной, письменной форме, и в форме тестирования. На промежуточную аттестацию отводится до 30 баллов.

#### **Экзаменационные вопросы (контролируемые компетенции ПКС-1)**

1. Сущность и актуальность дисциплины «Технологии программирования», основные понятия и определения дисциплины.
2. Жизненный цикл программного средства.
3. Модели жизненного цикла ПО.
4. Каскадная модель жизненного цикла ПО.
5. Модель жизненного цикла ПО с промежуточным контролем.
6. Спиральная модель жизненного цикла ПО.
7. Изменение жизненного цикла программного обеспечения при использовании CASE-технологий.
8. Качество программного обеспечения.
9. Модели качества ПО.
10. Метрики качества программного обеспечения.
11. Измерение и оценка качества ПО, стандартный метод оценки значений показателей качества.
12. Управление качеством ПС.
13. Диалоговые программы, типы диалога, формы диалога.
14. Спецификация ПС.

15. Определение требований к программному средству.
16. Спецификация качества программного средства.
17. Функциональная спецификация программного средства.
18. Методы контроля внешнего описания программного средства.
19. Способы записи алгоритмов.
20. Представление основных структур алгоритмов.
21. Псевдокоды.
22. Flow-формы.
23. Диаграммы Насси-Шнейдермана.
24. Классификация структур данных.
25. Файловые структуры, физическая организация файлов.
26. Логическая организация файлов.
27. Документирование файлов.
28. Модульные программы, модули и их свойства.
29. Сцепление и связность модулей.
30. Нисходящая и восходящая разработка программного обеспечения.
31. Программирование «с защитой от ошибок».
32. Основные подходы программирования, «стихийное» программирование.
33. Основные подходы программирования, структурный подход к программированию.
34. Основные подходы программирования, объектный подход к программированию.
35. Основные подходы программирования, компонентный подход и CASE-технологии.
36. Процедурное (императивное) программирование.
37. Функциональное программирование.
38. Декларативное программирование.
39. Объектно-ориентированное программирование.
40. Объектно-ориентированные языки программирования.
41. Спецификация программного обеспечения при структурном подходе.
42. Диаграммы переходов состояний.
43. Функциональные диаграммы.
44. Диаграммы потоков данных.
45. Моделирование управляющих процессов с помощью диаграмм потоков данных.
46. Структуры данных и диаграммы отношений компонентов данных.
47. Диаграммы Джексона.
48. Скобочные диаграммы Орра.
49. Сетевая модель данных.
50. Проектирование программного обеспечения при структурном подходе к программированию.
51. Разработка структурной и функциональной схем.
52. Структурная схема разрабатываемого программного обеспечения.
53. Функциональная схема.
54. Метод пошаговой детализации для проектирования структуры программного обеспечения.
55. Структурные карты Константайна.
56. Проектирование структур данных.
57. Представление данных в оперативной памяти.
58. Представление данных во внешней памяти.
59. Проектирование программного обеспечения, основанное на декомпозиции данных, методика Джексона.
60. Проектирование программного обеспечения, основанное на декомпозиции данных, методика Варнье-Орра.
61. CASE-технологии, основанные на структурных методологиях анализа и проектирования.



62. UML – стандартный язык описания разработки программных продуктов с использованием объектного подхода.
63. Определение «вариантов использования».
64. Диаграммы вариантов использования.
65. Построение концептуальной модели предметной области.
66. Диаграмма последовательностей системы.
67. Системные события и операции.
68. Диаграммы деятельности.
69. Проектирование программного обеспечения при объектном подходе.
70. Разработка структуры ПО при объектном подходе.
71. Определение отношений между объектами.
72. Диаграммы последовательностей этапа проектирования.
73. Диаграммы кооперации.
74. Уточнение отношений классов.
75. Интерфейсы в UML.
76. Проектирование классов.
77. Проектирование методов класса.
78. Компоновка программных компонентов.
79. Проектирование размещения программных компонентов для распределенных программных систем.
80. Методы доказательства правильности программ.
81. Метод индуктивных утверждений Флойда.
82. Метод Хора доказательства правильности программ.
83. Виды контроля качества разрабатываемого программного обеспечения.
84. Формирование тестовых наборов, основные подходы.
85. Ручной контроль программного обеспечения, методы ручного контроля.
86. Структурное тестирование, критерии формирования тестовых наборов.
87. Функциональное тестирование, методы формирования тестовых наборов.
88. Тестирование модулей и комплексное тестирование.
89. Оценочное тестирование.
90. Отладка программного обеспечения.
91. Классификация ошибок программного обеспечения.
92. Методы отладки программного обеспечения.
93. Методы и средства получения дополнительной информации об ошибках.
94. Общая методика отладки программного обеспечения.
95. Документирование и стандартизация.
96. Виды программных документов.
97. Основные правила оформления программной документации.
98. Основные инженерные подходы к созданию программ.
99. Классификация технологических подходов к созданию программ.
100. Классификация технологических подходов к созданию программ, подходы со слабой формализацией.
101. Классификация технологических подходов к созданию программ, строгие каскадные подходы.
102. Классификация технологических подходов к созданию программ, строгие каркасные подходы.
103. Классификация технологических подходов к созданию программ, генетические подходы.
104. Классификация технологических подходов к созданию программ, подходы на основе формальных преобразований.
105. Классификация технологических подходов к созданию программ, ранние подходы быстрой разработки.

106. Классификация технологических подходов к созданию программ, адаптивные технологические подходы.
107. Классификация технологических подходов к созданию программ, подходы исследовательского программирования.
108. Особенности и компоненты CASE-средств.
109. Объектно-ориентированные CASE-средства анализа и проектирования.
110. Структурные CASE-средства анализа и проектирования.
111. Case-средства компании IBM Rational Software, средство визуального моделирования Rational Rose.
112. Системы автоматизированного проектирования и их место среди других автоматизированных систем.
113. Структура САПР.
114. Разновидности САПР.
115. Понятие о CALS-технологиях.

### **Критерии формирования оценок по промежуточной аттестации**

**«Зачтено»** получают обучающиеся, которые

- свободно ориентируются в материале и отвечают без затруднений. Обучающийся способен к выполнению сложных заданий, постановке целей и выборе путей их реализации. Работа выполнена полностью без ошибок, решено 100% задач;
- относительно полно ориентируются в материале, отвечают без затруднений, допускают незначительное количество ошибок. Обучающийся способен к выполнению сложных заданий. Работа выполнена полностью, но имеются не более одной негрубой ошибки и одного недочета, не более трех недочетов. Допускаются незначительные неточности при решении задач, решено 70% задач;
- недостаточно высок уровень владения материалом. В процессе ответа на зачете допускаются ошибки и затруднения при изложении материала. Обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой. Обучающийся затрудняется с правильной оценкой предложенной задачи, дает неполный ответ, решено 55% задач;

**«Не зачтено»** получают обучающиеся, которые допускают значительные ошибки. Обучающийся имеет лишь начальную степень ориентации в материале. В работе число ошибок и недочетов превысило норму для оценки 3 или правильно выполнено менее 2/3 всей работы. Обучающийся дает неверную оценку ситуации, решено менее 50% задач.

## **6. МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ, ОПРЕДЕЛЯЮЩИЕ ПРОЦЕДУРЫ ОЦЕНИВАНИЯ ЗНАНИЙ, УМЕНИЙ, НАВЫКОВ И (ИЛИ) ОПЫТА ДЕЯТЕЛЬНОСТИ**

Максимальная сумма (61 балл), набираемая студентом по дисциплине включает две составляющие:

- первая составляющая – оценка регулярности, своевременности и качества выполнения студентом учебной работы по изучению дисциплины в течение периода изучения дисциплины (семестра, или нескольких семестров) (сумма – не более 70 баллов). Баллы, характеризующие успеваемость студента по дисциплине, набираются им в течение всего периода обучения за изучение отдельных тем и выполнение отдельных видов работ.

- вторая составляющая – оценка знаний студента по результатам промежуточной аттестации (до 61 балла).

Критерием оценки уровня сформированности компетенций в рамках учебной дисциплины «Технологии программирования» в 3 семестре является зачет.

Общий балл текущего и рубежного контроля складывается из следующих составляющих (Приложение).

Целью промежуточных аттестаций по дисциплине является оценка качества освоения дисциплины обучающимися.

Таблица 9

#### Результаты освоения учебной дисциплины, подлежащие проверке

Результаты обучения (компетенции)	Основные показатели оценки результатов обучения	Вид оценочного материала, обеспечивающего формирование компетенций
ПКС-1 – способен разрабатывать требования и проектировать программное обеспечение	<p><b>ИД-1<sub>ПКС-1</sub>. Знать:</b></p> <ul style="list-style-type: none"> <li>– Возможности существующей программно-технической архитектуры</li> <li>– Возможности современных и перспективных средств разработки программных продуктов, технических средств</li> <li>– Методологии разработки программного обеспечения и технологии программирования</li> </ul> <p><b>ИД-2<sub>ПКС-1</sub>. Уметь:</b></p> <ul style="list-style-type: none"> <li>– Проводить анализ исполнения требований</li> <li>– Вырабатывать варианты реализации требований</li> <li>– Проводить оценку и обоснование рекомендуемых решений</li> <li>– Осуществлять коммуникации с заинтересованными сторонами</li> </ul> <p><b>ИД-3<sub>ПКС-1</sub>. Владеть:</b></p> <ul style="list-style-type: none"> <li>– Навыками анализа возможностей реализации требований к программному обеспечению</li> <li>– Навыками оценки времени и трудоемкости реализации требований к программному обеспечению</li> <li>– Навыками согласования требований к программному обеспечению с заинтересованными сторонами</li> </ul>	Типовые оценочные материалы для устного опроса, типовые тестовые задания, лабораторные работы, задания для самостоятельной работы (раздел 5)

## 7. УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### 7.1. Нормативно-правовая база

1. ГОСТ «Единая система программной документации».
2. ГОСТ Р ИСО/МЭК ТО 15271-2002 «Руководство по применению ГОСТ Р ИСО/МЭК 12270 (Процессы жизненного цикла программных средств)».
3. ГОСТ Р ИСО/МЭК ТО 12182-2010 «Классификация программных средств».
4. ISO/IEC 14764:2006 «Разработка программного обеспечения. Процессы жизненного цикла программного обеспечения. Сопровождение».
5. ГОСТ Р ИСО/МЭК 12119-2000 «Информационная технология. Пакеты программ. Требования к качеству и тестирование»
6. ГОСТ 28195-89 «Оценка качества программных средств. Общие положения».

7. ISO/IEC 25000:2005 «Технология программного обеспечения. Требования и оценка качества программного продукта. Руководство».
8. ISO/IEC 25001:2014 «Программирование. Требования к качеству программного продукта и его оценка. Планирование и менеджмент».
9. ISO/IEC 25010:2011 «Проектирование систем и разработка программного обеспечения. Требования к качеству систем и программного обеспечения и их оценка (SQuaRE). Модели качества систем и программного обеспечения».
10. ISO/IEC 25012:2008 «Программная инженерия – Требования к качеству и оценке программного обеспечения. Модель качества данных».
11. ISO/IEC 25020:2007 «Разработка программного обеспечения. Требования к качеству и оценка качества программного продукта. Измерительная эталонная модель и руководство».

## **7.2. Основная литература**

1. Анашкина Н.В. Технологии и методы программирования (1-е изд.), учебное пособие, 2012 г. Библиотека КБГУ, 6 экз.
2. Кнут Д.Э. Искусство программирования: Учебное пособие. 3-е изд. – М.: Вильямс. В 3 т. 2012 г. Библиотека КБГУ, 10 экз.
3. Кулямин В.В. Технологии программирования. Компонентный подход [Электронный ресурс] / В.В. Кулямин. — 2-е изд. — Электрон. текстовые данные. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 590 с. — 5-9556-0067-1. — Режим доступа: <http://www.iprbookshop.ru/73733.html>
4. Кручинин В.В. Технологии программирования [Электронный ресурс] : учебное пособие / В.В. Кручинин. — Электрон. текстовые данные. — Томск: Томский государственный университет систем управления и радиоэлектроники, 2013. — 271 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/72195.html>
5. Букунов С.В. Основы объектно-ориентированного программирования [Электронный ресурс] : учебное пособие / С.В. Букунов, О.В. Букунова. — Электрон. текстовые данные. — СПб. : Санкт-Петербургский государственный архитектурно-строительный университет, ЭБС АСВ, 2017. — 196 с. — 978-5-9227-0713-8. — Режим доступа: <http://www.iprbookshop.ru/74339.html>
6. Калентьев А.А. Новые технологии в программировании [Электронный ресурс] : учебное пособие / А.А. Калентьев, Д.В. Гарайс, А.Е. Горяинов. — Электрон. текстовые данные. — Томск: Томский государственный университет систем управления и радиоэлектроники, Эль Контент, 2014. — 176 с. — 978-5-4332-0185-9. — Режим доступа: <http://www.iprbookshop.ru/72142.html>

## **7.3. Дополнительная литература**

1. Мейер Б. Объектно-ориентированное программирование и программная инженерия [Электронный ресурс] / Б. Мейер. — 3-е изд. — Электрон. текстовые данные. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Эр Медиа, 2018. — 285 с. — 978-5-4486-0513-0. — Режим доступа: <http://www.iprbookshop.ru/79706.html>
2. Киселева Т.В. Программная инженерия. Часть 1 [Электронный ресурс] : учебное пособие / Т.В. Киселева. — Электрон. текстовые данные. — Ставрополь: Северо-Кавказский федеральный университет, 2017. — 137 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/69425.html>

3. Орлов С.А. Технологии разработки программного обеспечения. – Питер, 2012.
4. Кунву Ли. Основы САПР CAD/CAM/CAE. – СПб.: Питер, 2005.
5. Ковалевская Е.В., Комлева Н.В. Методы программирования. – М.: Евразийский открытый институт, 2011 г.

#### **7.4. Периодические издания (газета, вестник, бюллетень, журнал)**

1. Журнал «Объектно-ориентированное программирование для профессионалов».
2. Журнал «Компьютеры & Программы».
3. Журнал «Программирование».

#### **7.5. Интернет-ресурсы**

1. С.В. Зыков. Технологии и средства разработки корпоративных систем. Интуит.ру, <http://www.intuit.ru/department/itmngt/techcorpsys/>
2. Т.С. Васючкова, М.А. Держо, Н.А. Иванчева, Т.П. Пухначева. Управление проектами с использованием Microsoft Project. Интуит.ру, <http://www.intuit.ru/department/itmngt/pmusemspr/>
3. Д.В. Андреев. Организация процессов разработки программного обеспечения с использованием Team Foundation Server 2010. Интуит.ру, <http://www.intuit.ru/department/se/vdevtfds2010/>
4. А.В. Марчуков, А.О. Савельев. Работа в Microsoft Visual Studio. Интуит.ру, <http://www.intuit.ru/department/itmngt/workinmsvistudio/>
5. В.А. Петрухин, Е.М. Лаврищева. Методы и средства инженерии программного обеспечения. Интуит.ру, <http://www.intuit.ru/department/se/swebok/>
6. [habr.com](http://habr.com)
7. <http://www.intuit.ru>
8. <http://citforum.ru>

#### **7.6. Современные профессиональные базы данных**

1. База данных Science Index (РИНЦ) <http://elibrary.ru>
2. Национальная электронная библиотека РГБ <https://нэб.рф>
3. Крупнейшая единая база данных, содержащая аннотации и информацию о цитируемости рецензируемой научной литературы, со встроенными инструментами отслеживания, анализа и визуализации данных. [www.scopus.com](http://www.scopus.com)
4. Самая полная математическая база данных, охватывающая материалы с конца 19 века. zbMath содержит документы, журналы и книги по математике, статистике, информатике, а также машиностроению, физике, естественным наукам и др. [www.zbmath.org](http://www.zbmath.org) (доступ открытый).

#### **7.7. Методические указания к лабораторным занятиям**

1. Акбашева Е.А., Акбашева Г.А. Технологии программирования. – Нальчик: Каб.-Балк. ун-т, 2009.

#### **7.8. Методические указания по проведению различных учебных занятий и самостоятельной работы**

##### **Методические рекомендации по работе с литературой**

Всю литературу можно разделить на учебники и учебные пособия, оригинальные научные монографические источники, научные публикации в периодической печати. Из них можно выделить литературу основную (рекомендуемую), дополнительную и литературу для углубленного изучения дисциплины.

Изучение дисциплины следует начинать с учебника, поскольку учебник – это книга, в которой изложены основы научных знаний по определенному предмету в соответствии с целями и задачами обучения, установленными программой.

При работе с литературой необходимо учитывать, что имеются различные виды чтения, и каждый из них используется на определенных этапах освоения материала.

Предварительное чтение направлено на выявление в тексте незнакомых терминов и поиск их значения в справочной литературе. В частности, при чтении указанной литературы необходимо подробнейшим образом анализировать понятия.

Сквозное чтение предполагает прочтение материала от начала до конца. Сквозное чтение литературы из приведенного списка дает возможность студенту сформировать свод основных понятий из изучаемой области и свободно владеть ими.

Выборочное – наоборот, имеет целью поиск и отбор материала. В рамках данного курса выборочное чтение, как способ освоения содержания курса, должно использоваться при подготовке к практическим занятиям по соответствующим разделам.

Аналитическое чтение – это критический разбор текста с последующим его конспектированием. Освоение указанных понятий будет наиболее эффективным в том случае, если при чтении текстов студент будет задавать к этим текстам вопросы. Часть из этих вопросов сформулирована в ФОС в перечне вопросов для собеседования. Перечень этих вопросов ограничен, поэтому важно не только содержание вопросов, но сам принцип освоения литературы с помощью вопросов к текстам.

Целью изучающего чтения является глубокое и всестороннее понимание учебной информации. Есть несколько приемов изучающего чтения:

1. Чтение по алгоритму предполагает разбиение информации на блоки: название; автор; источник; основная идея текста; фактический материал; анализ текста путем сопоставления имеющихся точек зрения по рассматриваемым вопросам; новизна.
2. Прием постановки вопросов к тексту имеет следующий алгоритм:
  - медленно прочитать текст, стараясь понять смысл изложенного;
  - выделить ключевые слова в тексте;
  - постараться понять основные идеи, подтекст и общий замысел автора.
3. Прием тезирования заключается в формулировании тезисов в виде положений, утверждений, выводов.

К этому можно добавить и иные приемы: прием реферирования, прием комментирования.

Важной составляющей любого солидного научного издания является список литературы, на которую ссылается автор. При возникновении интереса к какой-то обсуждаемой в тексте проблеме всегда есть возможность обратиться к списку относящейся к ней литературы. В этом случае вся проблема как бы разбивается на составляющие части, каждая из которых может изучаться отдельно от других. При этом важно не терять из вида общий контекст и не погружаться чрезмерно в детали, потому что таким образом можно не увидеть главного.

Подготовка к экзамену должна проводиться на основе лекционного материала, материала практических занятий с обязательным обращением к основным учебникам по

курсу. Это позволит исключить ошибки в понимании материала, облегчит его осмысление, прокомментирует материал многочисленными примерами.

### **Методические рекомендации по организации самостоятельной работы**

Самостоятельная работа обучающихся – способ активного, целенаправленного приобретения обучающимися новых для него знаний и умений без непосредственного участия в этом процесса преподавателей. Повышение роли самостоятельной работы обучающихся при проведении различных видов учебных занятий предполагает:

- оптимизацию методов обучения, внедрение в учебный процесс новых технологий обучения, повышающих производительность труда преподавателя, активное использование информационных технологий, позволяющих обучающемуся в удобное для него время осваивать учебный материал;
- широкое внедрение компьютеризированного тестирования;
- совершенствование методики проведения практик и научно-исследовательской работы обучающихся, поскольку именно эти виды учебной работы в первую очередь готовят обучающихся к самостоятельному выполнению профессиональных задач;
- модернизацию системы курсового и дипломного проектирования, которая должна повышать роль обучающихся в подборе материала, поиске путей решения задач.

Самостоятельная работа приводит обучающихся к получению нового знания, упорядочению и углублению имеющихся знаний, формированию у него профессиональных навыков и умений. Самостоятельная работа выполняет ряд функций:

- развивающую;
- информационно-обучающую;
- ориентирующую и стимулирующую;
- воспитывающую;
- исследовательскую.

В рамках курса выполняются следующие виды самостоятельной работы:

- Проработка учебного материала (по конспектам, учебной и научной литературе);
- Выполнение разноуровневых заданий;
- Работа с тестами и вопросами для самопроверки;
- Выполнение итоговой контрольной работы.

Обучающимся рекомендуется с самого начала освоения курса работать с литературой и предлагаемыми заданиями в форме подготовки к очередному аудиторному занятию. При этом актуализируются имеющиеся знания, а также создается база для усвоения нового материала, возникают вопросы, ответы на которые обучающийся получает в аудитории.

Необходимо отметить, что некоторые задания для самостоятельной работы по курсу имеют определенную специфику. При освоении курса обучающийся может пользоваться библиотекой вуза, которая в полной мере обеспечена соответствующей литературой. Значительную помощь в подготовке к очередному занятию может оказать имеющийся в учебно-методическом комплексе краткий конспект лекций. Он же может использоваться и для закрепления полученного в аудитории материала. Самостоятельная работа обучающихся предусмотрена учебным планом и выполняется в обязательном порядке. Задания предложены по каждой изучаемой теме и могут готовиться индивидуально или в группе. По необходимости обучающийся может обращаться за консультацией к преподавателю. Выполнение заданий контролируется и оценивается преподавателем.

Для успешной организации самостоятельной работы все активнее применяются разнообразные образовательные ресурсы в сети Интернет: системы тестирования по различным областям, виртуальные лекции, лаборатории, при этом пользователю достаточно

иметь компьютер и подключение к Интернету для того, чтобы связаться с преподавателем, решать вычислительные задачи и получать знания. Использование сетей усиливает роль самостоятельной работы обучающихся и позволяет кардинальным образом изменить методику преподавания.

Обучающийся может получать все задания и методические указания через сервер, что дает ему возможность привести в соответствие личные возможности с необходимыми для выполнения работ трудозатратами. Обучающийся имеет возможность выполнять работу дома или в аудитории. Большое воспитательное и образовательное значение в самостоятельном учебном труде обучающийся имеет самоконтроль. Самоконтроль возбуждает и поддерживает внимание и интерес, повышает активность памяти и мышления, позволяет обучающемуся своевременно обнаружить и устранить допущенные ошибки и недостатки, объективно определить уровень своих знаний, практических умений. Самое доступное и простое средство самоконтроля с применением информационно-коммуникационных технологий - это ряд тестов «on-line», которые позволяют в режиме реального времени определить свой уровень владения предметным материалом, выявить свои ошибки и получить рекомендации по самосовершенствованию.

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

### **8.1. Требования к материально-техническому обеспечению**

Минимально необходимый для реализации ОПОП перечень материально-технического обеспечения включает в себя: лекционные аудитории (оборудованные видеопроекционным оборудованием для презентаций, средствами звуковоспроизведения, экраном и имеющие выход в сеть Интернет), помещения для проведения семинарских и практических занятий (оборудованные учебной мебелью), компьютерные классы и др. (в соответствии с ФГОС и учебным планом).

Во время самостоятельной работы студенты используют компьютерные классы института информатики, электроники и компьютерных технологий, электронные читальные залы КБГУ и домашние компьютеры.

#### **Лицензионное программное обеспечение, используемое для проведения лекционных и лабораторных занятий**

1. Microsoft Windows 10.
2. Microsoft Office 2016.
3. Visual Studio 2019.

### **8.2. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья**

Для студентов с ограниченными возможностями здоровья созданы специальные условия для получения образования. В целях доступности получения высшего образования по образовательным программам инвалидами и лицами с ограниченными возможностями здоровья университетом обеспечивается (аудитория для самостоятельной работы и коллективного пользования специальными техническими средствами для обучения инвалидов и лиц с ОВЗ в КБГУ, аудитория № 145 Главный корпус КБГУ):

1. Альтернативная версия официального сайта в сети «Интернет» для слабовидящих;
2. Для инвалидов с нарушениями зрения (слабовидящие, слепые):
  - присутствие ассистента, оказывающего обучающемуся необходимую помощь, дублирование вслух справочной информации о расписании учебных занятий; наличие средств для усиления остаточного зрения, брайлевской компьютерной техники, видеовеличителей, программ не визуального доступа к информации, программ-синтезаторов речи и других технических средств приема-передачи учебной информации в доступных формах для студентов с нарушениями зрения;
  - задания для выполнения на экзамене зачитываются ассистентом;



- письменные задания выполняются на бумаге, надиктовываются ассистенту обучающимся;

3. Для инвалидов и лиц с ограниченными возможностями здоровья по слуху (слабослышащие, глухие):

- на зачете/экзамене присутствует ассистент, оказывающий студенту необходимую техническую помощь с учетом индивидуальных особенностей (он помогает занять рабочее место, передвигаться, прочитать и оформить задание, в том числе записывая под диктовку);
- зачет/экзамен проводится в письменной форме;

4. Для инвалидов и лиц с ограниченными возможностями здоровья, имеющих нарушения опорно-двигательного аппарата, созданы материально-технические условия, обеспечивающие возможность беспрепятственного доступа обучающихся в учебные помещения, объекты питания, туалетные и другие помещения университета, а также пребывания в указанных помещениях (наличие расширенных дверных проемов, поручней и других приспособлений).

- письменные задания выполняются на компьютере со специализированным программным обеспечением или надиктовываются ассистенту;
- по желанию студента экзамен проводится в устной форме.

Обучающиеся из числа лиц с ограниченными возможностями здоровья обеспечены электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

## 9. ЛИСТ ПЕРЕУТВЕРЖДЕНИЯ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ

Рабочая программа:

одобрена на 2022/2023 учебный год. Протокол № \_\_\_\_\_ заседания \_\_\_\_\_ кафедры \_\_\_\_\_ от  
« \_\_\_\_ » \_\_\_\_\_ 20\_\_ г.

В рабочую программу внесены следующие изменения:

---

---

---

---

Разработчик программы \_\_\_\_\_

Зав. кафедрой \_\_\_\_\_

## ПРИЛОЖЕНИЕ

№п/п	Вид контроля	Сумма баллов			
		Общая сумма	1-я точка	2-я точка	3-я точка
1	Посещение занятий	до 10 б.	до 3 б.	до 3б.	до 4б.
2	Текущий контроль:	до 28 б.	до 9 б.	до 9 б.	до 10 б.
	Выполнение лабораторных работ	до 18 б.	до 6 б.	до 6 б.	до 6 б.
	Выполнение самостоятельных заданий	от 0 до 10 б.	от 0 до 3 б.	от 0 до 3 б.	от 0 до 4 б.
3	Рубежный контроль	до 27 баллов	до 9 б.	до 9 б.	до 9 б.
	тестирование	от 0 до 15 б.	от 0 до 5 б.	от 0 до 5 б.	от 0 до 5 б.
	коллоквиум	от 0 до 12 б.	от 0 до 4 б.	от 0 до 4 б.	от 0 до 4 б.
4	Итого сумма текущего и рубежного контроля	до 70 б.	до 23 б.	до 23 б.	до 24 б.
5	Первый этап (базовый уровень) – оценка «удовлетворительно»	не менее 36 б.	не менее 12 б.	не менее 12 б.	не менее 12 б.
6	Второй этап (продвинутый уровень) – оценка «хорошо»	51-60 б.	менее 23 б.	менее 23 б.	менее 24 б.
7	Третий этап (высокий уровень) – оценка «отлично»	61-70 б.	не менее 23 б.	не менее 23 б.	не менее 24 б.