

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования



**Пермский национальный исследовательский
политехнический университет**

УТВЕРЖДАЮ

Проректор по образовательной
деятельности

 А.Б. Петроченков

« 28 » апреля 20 23 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Дисциплина: Языки и методы программирования 2
(наименование)

Форма обучения: очная
(очная/очно-заочная/заочная)

Уровень высшего образования: бакалавриат
(бакалавриат/специалитет/магистратура)

Общая трудоёмкость: 108 (3)
(часы (ЗЕ))

Направление подготовки: 01.03.02 Прикладная математика и информатика
(код и наименование направления)

Направленность: Математическое моделирование (СУОС)
(наименование образовательной программы)

1. Общие положения

1.1. Цели и задачи дисциплины

Основной целью курса является расширение и углубление знаний, умений и навыков будущих специалистов-математиков в области разработки прикладных программ на языках высокого уровня для современных вычислительных машин и комплексов, овладение принципами построения сложных программных систем.

Задачами изучения дисциплины является приобретение студентом следующих способностей:

- Углубление знаний в области построения алгоритмов обработки данных и методов их реализации на современных ЭВМ.
- Углубление знаний в области построения языков программирования, способов и механизмов управления данными.
- Углубление знаний в области современных языков программирования для решения различных классов задач.
- Формирование умений по разработке проблемно ориентированных алгоритмов для решения прикладных задач по профилю будущей деятельности.
- Формирование умений по написанию прикладных программ с использованием современных языков программирования высокого уровня и сред разработки.
- Расширение представлений студентов о способах и механизмах построения человеко-машинного интерфейса при разработке прикладных программ.
- Формирование владений основами алгоритмизации, навыками построения адекватных алгоритмических решений при решении прикладных задач по специальности.
- Формирование навыков использования языков программирования высокого уровня, методов трансляции и современных сред разработки для создания прикладных программ с реализацией человеко-машинного интерфейса.

1.2. Изучаемые объекты дисциплины

Алгоритмические системы, искусственные языки, языки программирования, современные языки (C/C++) и интегрированные среды программирования (MS Visual Studio), компиляторы и трансляторы.

1.3. Входные требования

Не предусмотрены

2. Планируемые результаты обучения по дисциплине

Компетенция	Индекс индикатора	Планируемые результаты обучения по дисциплине (знать, уметь, владеть)	Индикатор достижения компетенции, с которым соотнесены планируемые результаты обучения	Средства оценки
-------------	-------------------	---	--	-----------------

Компетенция	Индекс индикатора	Планируемые результаты обучения по дисциплине (знать, уметь, владеть)	Индикатор достижения компетенции, с которым соотнесены планируемые результаты обучения	Средства оценки
ПК-2.1	ИД-1ПК-2.1	<p>Знает – основные принципы построения алгоритмов решения прикладных задач и способов их про-граммной реализации;</p> <p>– основные подходы к разработке программно-го обеспечения: структурный, объектно-ориентированный;</p> <p>принципы нисходящей и восходящей разработки программ;</p> <p>– основные понятия языков программирования; синтаксис, семантику языков программирования; формальные способы описания алгоритмов; типы данных, способы и механизмы управления данными; методы и основные этапы трансляции; конструкции распределенного и параллельного программирования; методы и алгоритмы управления динамическими структурами данных;</p> <p>– хотя бы один язык программирования (C/C++).</p>	<p>Знает методы и приемы формализации и алгоритмизации поставленных задач, алгоритмы решения типовых задач, синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования, технологии программирования, методы и приемы отладки программного кода</p>	Контрольная работа
ПК-2.1	ИД-2ПК-2.1	<p>Умеет – применять на практике методы и алгоритмы управления динамическими структурами дан-ных;</p> <p>– программировать хотя бы на одном языке вы-сокого уровня (C/C++);</p> <p>– понимать и применять на практике компьютерные технологии для решения различных задач</p> <p>– решать научно-технические задачи с</p>	<p>Умеет использовать методы и приемы формализации и алгоритмизации задач, применять стандартные алгоритмы, применять выбранные языки программирования для написания программного кода, применять современные компиляторы</p>	Индивидуальное задание

Компетенция	Индекс индикатора	Планируемые результаты обучения по дисциплине (знать, уметь, владеть)	Индикатор достижения компетенции, с которым соотнесены планируемые результаты обучения	Средства оценки
		применением методов прикладной математики и информатики.		
ПК-2.1	ИД-3ПК-2.1	Владеет: – навыками разработки алгоритмов решения прикладных задач с использованием ПК; – навыками разработки прикладных программ на языках программирования высокого уровня, в средах визуального программирования и средах разработки (MS Visual Studio); – навыками решения практических задач с использованием современных языков и сред программирования.	Владеет навыками составления формализованных описаний решений и разработки алгоритмов, создания программного кода решения поставленных задач в соответствии с требованиями технического задания	Отчёт по практическому занятию

3. Объем и виды учебной работы

Вид учебной работы	Всего часов	Распределение по семестрам в часах	
		Номер семестра	
		2	
1. Проведение учебных занятий (включая проведение текущего контроля успеваемости) в форме:	42	42	
1.1. Контактная аудиторная работа, из них:			
- лекции (Л)	16	16	
- лабораторные работы (ЛР)			
- практические занятия, семинары и (или) другие виды занятий семинарского типа (ПЗ)	24	24	
- контроль самостоятельной работы (КСР)	2	2	
- контрольная работа			
1.2. Самостоятельная работа студентов (СРС)	30	30	
2. Промежуточная аттестация			
Экзамен	36	36	
Дифференцированный зачет			
Зачет			
Курсовой проект (КП)			
Курсовая работа (КР)			
Общая трудоемкость дисциплины	108	108	

4. Содержание дисциплины

Наименование разделов дисциплины с кратким содержанием	Объем аудиторных занятий по видам в часах			Объем внеаудиторных занятий по видам в часах
	Л	ЛР	ПЗ	СРС
2-й семестр				
Язык программирования C/C++	7	0	10	13
<p>Тема 1. Обзор алгоритмического языка C/C++ Синтаксис алгоритмического языка C/C++. Основные типы данных. Операторы. Бинарные и унарные операции. Приоритет операций. Логические операции, битовые операции.</p> <p>Тема 2. Функции. Функции в C/C++. Прототипы функций. Контроль параметров функции. Рекурсивные функции.</p> <p>Тема 3. Структуры данных.</p> <p>Тема 4. Указатели и динамическая память Указатели. Адресная арифметика. Операции с указателями.</p> <p>Тема 5. Операции с динамической памятью Выделение и освобождение динамической памяти. Фрагментация памяти.</p> <p>Тема 6. Классы. Классы и объекты. Синтаксис описания класса. Примеры реализации.</p> <p>Тема 7. Реализация основных принципов ООП в языках C/C++, C#. Основные принципы объектно-ориентированного подхода к программированию: инкапсуляция, наследование, полиморфизм. Примеры реализации</p>				

Наименование разделов дисциплины с кратким содержанием	Объем аудиторных занятий по видам в часах			Объем внеаудиторных занятий по видам в часах
	Л	ЛР	ПЗ	СРС
Динамические структуры данных	5	0	9	11
<p>Тема 8. Методы синтаксического анализа Форма Бэкуса-Наура. Методы синтаксического анализа: конечные автоматы, автоматы с магазинной памятью.</p> <p>Тема 9. Управление данными. Типы данных, способы и механизмы управления данными. Принципы и механизмы обра-ботки данных.</p> <p>Тема 10. Динамические структуры данных. Линейные списки. Типы динамических структур данных. Понятие линейного списка. Односвязные и дву-связные списки. Алгоритмы поиска, вставки и удаления элементов списков. Структуры дан-ных: список, стек, дек.</p> <p>Тема 11. Графы и деревья. Основные понятия и определения. Бинарное дерево. Обход дерева. Алгоритмы на графах и деревьях. Алгоритмы поиска, вставки и удаления элемента.</p> <p>Тема 12. Алгоритмы над динамическими структурами данных. Обратная польская нотация (постфиксная нотация). Вычисления на стеке.</p>				
Современные технологии программирования	4	0	5	6
<p>Тема 13. Разработка пре-/постпроцессора Разработка пре-/постпроцессора с использованием современных сред визуального про-граммирования и сред разработки (MS Visual Studio), основанных на использовании объект-но-ориентированного подхода к программированию. Реализация человеко-машинного ин-терфейса.</p> <p>Тема 14. Отладка программного кода Отладка программ: понятия, цели, возможности отладки. Профилирование программ</p> <p>Тема 15. Производительность работы программ</p> <p>Введение в высокопроизводительные вычисления. Производительность работы программ. Введение в оптимизацию. Методы оптимизации программ.</p> <p>Тема 16. Обзор языков и компиляторов для высокопроизводительных вычислений Языки и компиляторы для высокопроизводительных вычислений. Введение в OpenMP.</p>				
ИТОГО по 2-му семестру	16	0	24	30
ИТОГО по дисциплине	16	0	24	30

Тематика примерных практических занятий

№ п.п.	Наименование темы практического (семинарского) занятия
1	Практическое изучение синтаксиса языка C/C++. Проведение ма-тематических расчетов с численной реализацией на алгоритмиче-ском языке C/C++. Решение задач по программированию.
2	Работа со структурированными типами данных в языке C/C++. Об-работка массивов данных. Реализация алгоритмов обработки мас-сивов данных (алгоритмов сортировки, поиска элементов и т.д.). Работа с файлами. Решение задач по программированию.
3	Работа с динамической памятью в C/C++. Выделение и освобожде-ние динамической памяти. Решение задач по программированию.
4	Разработка программы для ЭВМ на алгоритмическом языке C/C++ с применением принципов структурного подхода к программиро-ванию.
5	Разработка программы для ЭВМ на алгоритмическом языке C/C++ с применением принципов объектно-ориентированного подхода к программированию.
6	Разработка программы для ЭВМ, реализующей линейный одно-связный/двусвязный список. Реализация алгоритмов поиска, встав-ки и удаления элементов списка.
7	Реализация алгоритмов сортировки для линейных списков.
8	Разработка программы для ЭВМ, реализующей динамическую структуру данных типа бинарное дерево. Реализация алгоритмов поиска, вставки и удаления элементов.
9	Решение прикладных задач по обработке данных с использованием динамических структур данных.
10	Разработка прикладной программы для ЭВМ с реализацией пре-процессорного и постпроцессорного интерфейсов. Отладка про-граммного кода.
11	Оптимизация программного кода. Разработка прикладной про-граммы для ЭВМ с использованием технологии OpenMP

5. Организационно-педагогические условия

5.1. Образовательные технологии, используемые для формирования компетенций

Проведение лекционных занятий по дисциплине основывается на активном методе обучения, при котором учащиеся не пассивные слушатели, а активные участники занятия, отвечающие на вопросы преподавателя. Вопросы преподавателя нацелены на активизацию процессов усвоения материала, а также на развитие логического мышления. Преподаватель заранее намечает список вопросов, стимулирующих ассоциативное мышление и установление связей с ранее освоенным материалом.

Практические занятия проводятся на основе реализации метода обучения действием: определяются проблемные области, формируются группы. При проведении практических занятий преследуются следующие цели: применение знаний отдельных дисциплин и креативных методов для решения проблем и принятия решений; отработка у обучающихся навыков командной работы, межличностных коммуникаций и развитие лидерских качеств; закрепление основ теоретических знаний.

При проведении учебных занятий используются интерактивные лекции, групповые дискуссии, ролевые игры, тренинги и анализ ситуаций и имитационных моделей.

5.2. Методические указания для обучающихся по изучению дисциплины

При изучении дисциплины обучающимся целесообразно выполнять следующие рекомендации:

1. Изучение учебной дисциплины должно вестись систематически.
2. После изучения какого-либо раздела по учебнику или конспектным материалам рекомендуется по памяти воспроизвести основные термины, определения, понятия раздела.
3. Особое внимание следует уделить выполнению отчетов по практическим занятиям, индивидуальным комплексным заданиям на самостоятельную работу.
4. Вся тематика вопросов, изучаемых самостоятельно, задается на лекциях преподавателем. Им же даются источники (в первую очередь вновь изданные в периодической научной литературе) для более детального понимания вопросов, озвученных на лекции.

6. Перечень учебно-методического и информационного обеспечения для самостоятельной работы обучающихся по дисциплине

6.1. Печатная учебно-методическая литература

№ п/п	Библиографическое описание (автор, заглавие, вид издания, место, издательство, год издания, количество страниц)	Количество экземпляров в библиотеке
1. Основная литература		
1	Барте́ньев О. В. Фортран для студентов. Москва : Диалог-МИФИ, 1999. 397 с.	2
2	Барте́ньев О.В. Фортран для профессионалов. Математическая библиотека IMSL. Ч.2. М. : Диалог-МИФИ, 2001. 319 с.	1
3	Барте́ньев О.В. Фортран для профессионалов. Математическая библиотека IMSL. Ч.3. М. : Диалог-МИФИ, 2001. 368 с.	1
4	Вирт Н. Алгоритмы и структуры данных : пер. с англ. Москва : Мир, 1989. 360 с.	5
5	Вирт Н. Построение компиляторов : пер. с англ. Москва : ДМК Пресс, 2010. 191 с.	2
6	Новиков Ф.А. Дискретная математика для программистов : учебное пособие. Санкт-Петербург : Питер, 2003. 301 с.	39
7	Самохин А. Б., Самохина А. С. Численные методы и программирование на Фортране для персонального компьютера. Москва : Радио и связь, 1996. 224 с.	2
2. Дополнительная литература		
2.1. Учебные и научные издания		
1	Искусство программирования. Основные алгоритмы. 3-е изд. М. : Вильямс, 2000. 712 с.	2
2	Искусство программирования. Получисленные алгоритмы. 3-е изд. М. : Вильямс, 2004. 828 с.	1
3	Искусство программирования. Сортировка и поиск. 2-е изд. М. : Вильямс, 2004. 822 с.	3
4	Программирование. Т. 1. Москва : Академия, 2013. 268 с. 17,0 усл. печ. л.	4
5	Программирование. Т. 2. Москва : Академия, 2013. 239 с. 15,0 усл. печ. л.	4
6	Рихтер Д. CLR via C#. Программирование на платформе Microsoft .NET Framework 2.0 на языке C# : пер. с англ. М. : Рус. ред., 2007. 636 с.	1

7	Рихтер Д. Программирование на платформе Microsoft .NET Framework : пер. с англ. 3-е изд. М. : Рус. ред., 2006. 480 с.	1
2.2. Периодические издания		
	Не используется	
2.3. Нормативно-технические издания		
	Не используется	
3. Методические указания для студентов по освоению дисциплины		
	Не используется	
4. Учебно-методическое обеспечение самостоятельной работы студента		
	Не используется	

6.2. Электронная учебно-методическая литература

Вид литературы	Наименование разработки	Ссылка на информационный ресурс	Доступность (сеть Интернет / локальная сеть; авторизованный / свободный доступ)
Дополнительная литература	Введение в математическое моделирование: Учеб. пособие / Под ред. П.В. Трусова. — М.: Университетская книга, Логос, 2007. - 440 с. ISBN 978-5-98704-037-X.	http://www.mmsp.pstu.ru/userfiles/Posobie/MM_intro_PVTrusov.pdf	сеть Интернет; свободный доступ

6.3. Лицензионное и свободно распространяемое программное обеспечение, используемое при осуществлении образовательного процесса по дисциплине

Вид ПО	Наименование ПО
Операционные системы	MS Windows 7 (подп. Azure Dev Tools for Teaching до 27.03.2022)
Офисные приложения.	Microsoft Office Professional 2007. лиц. 42661567
Прикладное программное обеспечение общего назначения	Dr.Web Enterprise Security Suite, 3000 лиц, ПНИПУ ОЦНИТ 2017
Среды разработки, тестирования и отладки	C++ Builder 2007 Enterprise , лиц. PO-398ESD, ПНИПУ
Среды разработки, тестирования и отладки	MS Visual studio 2019 community (Free)

6.4. Современные профессиональные базы данных и информационные справочные системы, используемые при осуществлении образовательного процесса по дисциплине

Наименование	Ссылка на информационный ресурс
Научная библиотека Пермского национального исследовательского политехнического университета	http://lib.pstu.ru/
Электронно-библиотечная система Лань	https://e.lanbook.com/

Наименование	Ссылка на информационный ресурс
Электронно-библиотечная система IPRbooks	http://www.iprbookshop.ru/
Информационные ресурсы Сети КонсультантПлюс	http://www.consultant.ru/

7. Материально-техническое обеспечение образовательного процесса по дисциплине

Вид занятий	Наименование необходимого основного оборудования и технических средств обучения	Количество единиц
Лекция	Аудитории, оборудованные ноутбуком, видеопроектором	1
Практическое занятие	Персональные компьютеры (локальная компьютерная сеть)	10

8. Фонд оценочных средств дисциплины

Описан в отдельном документе

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Пермский национальный исследовательский политехнический
университет»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

для проведения промежуточной аттестации обучающихся по дисциплине

«Языки и методы программирования - 2»

Приложение к рабочей программе дисциплины

Направление подготовки:	01.03.02 Прикладная математика и информатика
Направленность (профиль) образовательной программы:	Математическое моделирование
Квалификация выпускника:	«Бакалавр»
Выпускающая кафедра:	Математическое моделирование систем и процессов
Форма обучения:	Очная

Курс: 1

Семестр: 2

Трудоёмкость:

Кредитов по рабочему учебному плану:	4	ЗЕ
Часов по рабочему учебному плану:	144	ч.

Виды промежуточного контроля:

Экзамен: 2 семестр

Пермь 2023

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине является частью (приложением) к рабочей программе дисциплины. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине разработан в соответствии с общей частью фонда оценочных средств для проведения промежуточной аттестации основной образовательной программы, которая устанавливает систему оценивания результатов промежуточной аттестации и критерии выставления оценок. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине устанавливает формы и процедуры текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине.

1. Перечень контролируемых результатов обучения по дисциплине, объекты оценивания и виды контроля

Согласно РПД освоение учебного материала дисциплины запланировано в течение одного семестра (2-го семестра базового учебного плана) и разбито на 3 учебных модуля. В каждом модуле предусмотрены аудиторские лекционные и практические занятия, а также самостоятельная работа студентов. В рамках освоения учебного материала дисциплины формируются компоненты дисциплинарных компетенций *знать, уметь, владеть*, указанные в РПД, и которые выступают в качестве контролируемых результатов обучения (табл. 1.1).

Контроль уровня усвоенных знаний, усвоенных умений и приобретенных владений осуществляется в рамках текущего, рубежного и промежуточного контроля при изучении теоретического материала, сдаче отчетов по лабораторным работам и экзамена. Виды контроля сведены в таблицу 1.1.

Таблица 1.1. Перечень контролируемых результатов обучения по дисциплине

Контролируемые результаты обучения по дисциплине (ЗУВы)	Вид контроля					
	Текущий		Рубежный		Промежуточный	
	ПЗ	ТО	КР			Экзамен
Усвоенные знания						
3.1 способы и методы разработки программных средств и методы работы с ними		ТО1				ТВ
3.2 современные концепции и принципы языков программирования и особенности их применения при решении задач по специальности		ТО2				ТВ
3.3. современные тенденции развития языков программирования		ТО3				ТВ
3.4. основные принципы построения алгоритмов решения прикладных задач и способов их программной реализации		ТО4				ТВ
3.5. основные подходы к разработке программного обеспечения: структурный, объектно-ориентированный; принципы нисходящей и восходящей разработки программ		ТО5				ТВ
3.6. основные понятия языков программирования; синтаксис, семантику языков программирования; формальные способы описания алгоритмов; типы данных, способы и механизмы управления данными; методы и основные этапы трансляции; конструкции распределенного и параллельного программирования; методы и алгоритмы управления динамическими		ТО6				ТВ

структурами данных						
3.7. хотя бы один язык программирования (C/C++)		ТО7				ТВ
Освоенные умения						
У.1 разрабатывать и использовать сложные программные продукты для решения задач по специальности	ПЗ		КР1			ПЗ
У.2 применять на практике методы и алгоритмы управления динамическими структурами данных	ПЗ					ПЗ
У.3. программировать хотя бы на одном языке высокого уровня (C/C++)	ПЗ		КР1			ПЗ
У.4. понимать и применять на практике компьютерные технологии для решения различных задач	ПЗ					ПЗ
Приобретенные владения						
В.1 навыками работы прикладного программиста и продвинутого пользователя	ПЗ		КР1			ПЗ
В.2 навыками применения современных средств разработки для реализации алгоритмов решения прикладных задач	ПЗ		КР2			ПЗ
В.3 навыками разработки алгоритмов решения прикладных задач с использованием ПК	ПЗ					ПЗ
В.4 навыками разработки прикладных программ на языках программирования высокого уровня, в средах визуального программирования и средах разработки (MS Visual Studio)	ПЗ		КР2			ПЗ
В.5 навыками решения практических задач с использованием современных языков и сред программирования	ПЗ		КР2			ПЗ

ТО – коллоквиум (теоретический опрос); КР – рубежная контрольная работа; ТВ – теоретический вопрос; ПЗ – практическое задание.

Итоговой оценкой освоения дисциплинарных компетенций (результатов обучения по дисциплине) является промежуточная аттестация в виде экзамена, проводимая с учетом результатов текущего и рубежного контроля.

2. Виды контроля, типовые контрольные задания и шкалы оценивания результатов обучения

Текущий контроль успеваемости имеет целью обеспечение максимальной эффективности учебного процесса, управление процессом формирования заданных компетенций обучаемых, повышение мотивации к учебе и предусматривает оценивание хода освоения дисциплины. В соответствии с Положением о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся по образовательным программам высшего образования – программам бакалавриата, специалитета и магистратуры в ПНИПУ предусмотрены следующие виды и периодичность текущего контроля успеваемости обучающихся:

- входной контроль, проверка исходного уровня подготовленности обучаемого и его соответствия предъявляемым требованиям для изучения данной дисциплины;
- текущий контроль усвоения материала (уровня освоения компонента «знать» заданных компетенций) на каждом групповом занятии и контроль посещаемости лекционных занятий;
- промежуточный и рубежный контроль освоения обучаемыми отдельных компонентов «знать», «уметь» заданных компетенций путем компьютерного или бланочного тестирования, контрольных опросов, контрольных работ

(индивидуальных домашних заданий), защиты отчетов по лабораторным работам, рефератов, эссе и т.д.

Рубежный контроль по дисциплине проводится на следующей неделе после прохождения модуля дисциплины, а промежуточный – во время каждого контрольного мероприятия внутри модулей дисциплины;

- межсессионная аттестация, единовременное подведение итогов текущей успеваемости не менее одного раза в семестр по всем дисциплинам для каждого направления подготовки (специальности), курса, группы;

- контроль остаточных знаний.

2.1. Текущий контроль

Текущий контроль для оценивания знаниевого компонента дисциплинарных частей компетенций (табл. 1.1) в форме теоретического опроса или практических заданий проводится по каждой теме. Результаты по 4-балльной шкале оценивания заносятся в книжку преподавателя и учитываются в виде интегральной оценки при проведении промежуточной аттестации.

2.2. Рубежный контроль

Рубежный контроль для комплексного оценивания усвоенных знаний, усвоенных умений и приобретенных владений дисциплинарных частей компетенций (табл. 1.1) проводится согласно графика учебного процесса, приведенного в РПД, в форме рубежных контрольных работ (после изучения модулей 1 и 3 учебной дисциплины).

2.2.1. Рубежная контрольная работа

Согласно РПД запланировано 2 рубежные контрольные работы (КР) после освоения студентами учебных модулей дисциплины. Первая КР по модулю 1 «Язык программирования С/С++», вторая КР – по модулю 3 «Современные технологии программирования».

Типовые задания первой КР:

1. Концепция объектного программирования.
2. Реализация ООП в языке С++.
3. Ранее и позднее связывание.

Типовые задания второй КР:

1. Применение OpenMP для повышения эффективности вычислений.
2. Директивы OpenMP.

Типовые шкала и критерии оценки результатов рубежной контрольной работы приведены в общей части ФОС магистерской программы.

2.3. Промежуточная аттестация

Допуск к промежуточной аттестации осуществляется по результатам текущего и рубежного контроля. Условиями допуска являются успешная сдача не менее 80% практических заданий и всех теоретических опросов и положительная интегральная оценка по результатам текущего и рубежного контроля.

Промежуточная аттестация, согласно РПД, проводится в виде экзамена по дисциплине устно по билетам. Билет содержит теоретические вопросы (ТВ) для проверки усвоенных знаний и практические задания (ПЗ) для проверки усвоенных

умений и контроля уровня приобретенных владений всех заявленных дисциплинарных компетенций.

Билет формируется таким образом, чтобы в него попали вопросы и практические задания, контролирующие уровень сформированности *всех* заявленных дисциплинарных компетенций. Форма билета представлена в общей части ФОС программы бакалавров.

2.3.1. Типовые вопросы и задания для экзамена по дисциплине

Типовые вопросы для контроля усвоенных знаний:

1. Языки программирования С и С++. История создания и развития языка. Стандарты языка. Структура языка: алфавит, виды лексем. Особенности компиляции в С++. Директивы препроцессора `#include`, `#define`, `#undef`, `#pragma`, `#error`.
2. Языки программирования С и С++. Условная компиляция. Директивы `#if`, `#ifdef`, `#ifndef`, `#else`, `#elif`, `#endif`. Именованные и неименованные константы.
3. Языки программирования С и С++. Переменные, объявление переменных. Классы переменных. Простые типы данных: целочисленные, вещественные, логические, символьные. Операции над переменными этих типов. Области видимости переменных. Объявления `namespace` и `using`. Преобразование типов. Примеры.
4. Языки программирования С и С++. Основные операторы: присваивания, условная операция, выбора, множественного выбора, циклов, прерывание и продолжение циклов, перехода. Структура программы. Примеры программ.
5. Языки программирования С и С++. Форматный ввод/вывод в С/С++, возможности библиотеки `stdio`. Примеры форматного ввода/вывода. Библиотека потокового ввода/вывода `iostream`, ее возможности и примеры использования.

Типовые вопросы и практические задания для контроля освоенных умений:

1. Реализовать программу, позволяющую переводить число из десятичной в двоичную систему счисления.
2. На основе исходного файла создать массив удвоенных нечетных чисел. Упорядочить его по возрастанию элементов. Входящий файл `input.txt`, вывод осуществляется в файл `output.txt`.
3. Дана строка до точки, группа символов между пробелами считается словом. Определить, сколько слов заканчивается буквой «о».

Полный перечень теоретических вопросов и практических заданий в форме утвержденного комплекта экзаменационных билетов хранится на выпускающей кафедре.

2.3.2. Шкалы оценивания результатов обучения на экзамене

Оценка результатов обучения по дисциплине в форме уровня сформированности компонентов *знать, уметь, владеть* заявленных

дисциплинарных компетенций проводится по 4-х балльной шкале оценивания путем выборочного контроля во время экзамена.

Типовые шкала и критерии оценки результатов обучения при сдаче экзамена для компонентов *знать, уметь и владеть* приведены в общей части ФОС программы бакалавров.

3. Критерии оценивания уровня сформированности компонентов и дисциплинарных компетенций

3.1. Оценка уровня сформированности компонентов дисциплинарных компетенций

При оценке уровня сформированности дисциплинарных компетенций в рамках выборочного контроля при экзамене считается, что *полученная оценка за компонент проверяемой в билете дисциплинарной компетенции обобщается на соответствующий компонент всех дисциплинарных компетенций, формируемых в рамках данной учебной дисциплины.*

Типовые критерии и шкалы оценивания уровня сформированности компонентов дисциплинарных компетенций приведены в общей части ФОС программы бакалавров.

3.2. Оценка уровня сформированности дисциплинарных компетенций

Общая оценка уровня сформированности всех дисциплинарных компетенций проводится путем агрегирования оценок, полученных студентом за каждый компонент формируемых компетенций, с учетом результатов текущего и рубежного контроля в виде интегральной оценки по 4-х балльной шкале. Все результаты контроля заносятся в оценочный лист и заполняются преподавателем по итогам промежуточной аттестации.

Форма оценочного листа и требования к его заполнению приведены в общей части ФОС программы бакалавров.

При формировании итоговой оценки промежуточной аттестации в виде экзамена используются типовые критерии, приведенные в общей части ФОС программы бакалавров.