

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ»

УТВЕРЖДАЮ
Проректор по учебной работе

_____ Василенко В.Н.
(подпись) (Ф.И.О.)

«25» мая 2023 г.

РАБОЧАЯ ПРОГРАММА
ДИСЦИПЛИНЫ

Программирование на языках высокого уровня

Направление подготовки

09.03.03 Прикладная информатика

Направленность (профиль) подготовки

Цифровизация бизнес-процессов

Квалификация выпускника

бакалавр

Воронеж

1. Цели и задачи дисциплины (модуля)

Целью освоения дисциплины «Программирование на языках высокого уровня» является формирование компетенций обучающегося в области профессиональной деятельности по Реестру Минтруда – 06 Связь, информационные и коммуникационные технологии (в сфере исследования, разработки, внедрения и сопровождения информационных технологий и систем).

В рамках освоения ООП ВО выпускники готовятся к решению задач профессиональной деятельности следующих типов:

- производственно-технологический;
- организационно-управленческий;
- проектный.

Программа составлена в соответствии с требованиями Федерального государственного образовательного стандарта высшего образования, на основе примерной основной образовательной программы высшего образования по направлению подготовки 09.03.03 «Прикладная информатика», (уровень образования - бакалавриат).

2. Перечень планируемых результатов обучения, соотнесенных с планируемыми результатами освоения образовательной программы

№ п/п	Код компетенции	Наименование компетенции	Код и наименование индикатора достижения компетенции
1	УК-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	ИД2 _{УК-1} – Решает поставленные задачи, используя системный подход, на основе критического анализа и синтеза информации и оценивает последствия возможных решений
2	ПКв-2	Способность разрабатывать и адаптировать прикладное программное обеспечение	ИД3 _{ПК-2} – Демонстрирует навыки технического и рабочего проектирования компонентов информационных систем в соответствии со спецификой профиля подготовки
3	ПКв-8	Способность проводить тестирование компонентов программного обеспечения ИС	ИД1 _{ПК-8} – Выявление и исправление дефектов и несоответствий в архитектуре и дизайне ИС

Код и наименование индикатора достижения компетенции	Результаты обучения (показатели оценивания)
ИД2 _{УК-1} – Решает поставленные задачи, используя системный подход, на основе критического анализа и синтеза информации и оценивает последствия возможных решений	Знает:
	- способы ввода и вывода данных в С++ и Python, их достоинства и недостатки;
	- способы представления данных в С++ и Python, их достоинства и недостатки;
	- операторы языков в С++ и Python, их достоинства и недостатки;
	- основные структуры языков в С++ и Python, их достоинства и недостатки
	Умеет:
	- выбирать наиболее рациональные способы ввода и вывода данных;
	- выбирать наиболее рациональные способы представления данных;
	- выбирать наиболее рациональную структуру программы;
	- выбирать наиболее рациональный метод решения задачи
	Владеет:

Код и наименование индикатора достижения компетенции	Результаты обучения (показатели оценивания)
	- навыками выбора наиболее рациональных способов решения задачи на языках C++ и Python на основе анализа их достоинств и недостатков
ИДЗ _{ПК-2} – Демонстрирует навыки технического и рабочего проектирования компонентов информационных систем в соответствии со спецификой профиля подготовки	Знает основные виды алгоритмов проектирования компонентов информационных систем
	Умеет разрабатывать алгоритмы проектирования компонентов информационных систем
	Владеет основными способами разработки алгоритмов проектирования компонентов информационных систем
ИД1 _{ПКв-8} – Выявление и исправление дефектов и несоответствий в архитектуре и дизайне ИС	Знает: - основные методы разработки ИС на языках программирования C++ и Python; - методы отладки и тестирования ИС с целью выявления и исправления дефектов и несоответствий в архитектуре и дизайне
	Умеет: - использовать языки программирования C++ и Python для разработки ИС; - применять методы отладки и тестирования ИС с целью выявления и исправления дефектов и несоответствий в архитектуре и дизайне
	Владеет навыками разработки ИС, их отладки и тестирования

3. Место дисциплины в структуре ООП ВО

Дисциплина «Программирование на языках высокого уровня» относится к части, формируемой участниками образовательных отношений, Блока 1 ООП. Дисциплина является обязательной к изучению.

Дисциплина является предшествующей для изучения дисциплин «Объектно-ориентированные системы программирования», «Моделирование информационных и технологических процессов», учебной практики, ознакомительной практики.

4. Объем дисциплины и виды учебной работы

Общая трудоемкость дисциплины составляет 5 зачетных единиц.

Виды учебной работы	Всего ак. ч	Распределение трудоемкости по семестрам, ак. ч	
		1	2
Общая трудоемкость дисциплины	180	108	72
Контактная работа, в т.ч. аудиторные занятия:	102,95	47,95	55
Лекции	33	15	18
<i>в том числе в форме практической подготовки</i>	-	-	-
Практические занятия	66	30	36
<i>в том числе в форме практической подготовки</i>	66	30	36
Консультации текущие	1,65	0,75	0,9
Консультации перед экзаменом	2	2	-
Вид аттестации (экзамен/ за-	0,3	0,2	0,1

Виды учебной работы	Всего ак. ч	Распределение трудоемкости по семестрам, ак. ч	
		1	2
чет)			
Самостоятельная работа:	43,25	26,25	17
Проработка материалов по конспекту лекций	16,75	10,75	6
Проработка материалов по учебнику	10	5	5
Подготовка к практическим занятиям	16,5	10,5	6
Контроль	33,8	33,8	

5. Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

5.1 Содержание разделов дисциплины

№ п/п	Наименование раздела дисциплины	Содержание раздела	Трудоемкость раздела, акад. ч
1 семестр			
1.	Основные понятия языка C++	Структура программы на C++. Ввод и вывод данных. Представление данных	11
2	Разветвляющиеся алгоритмы	Условные операторы. Логические выражения. Оператор множественного выбора. Оператор перехода goto	9
3	Циклические алгоритмы. Функции	Операторы цикла. Функциональное программирование на C++	9
4	Обработка массивов в C++	Одномерные массивы и двумерные массивы. Инициализация. Доступ к элементам массива. Объявление и использование указателей в C++. Динамические массивы. Функции и массивы	20,35
5	Работа с символьными и строковыми данными в C++	Ввод-вывод символьных данных. Обработка символьных данных. Строки. Ввод-вывод строковых данных. Обработка строковых данных. Стандартные функции обработки строк. Передача строк в качестве параметров функций	7
6	Обработка структур в C++	Объявление шаблонов структур. Определение структур-переменных. Доступ к компонентам структуры. Инициализация структур. Копирование структур-переменных	8,3
7	Файловый ввод-вывод в C++	Файловый ввод-вывод. Функции состояния. Открытие файла для ввода и вывода. Изменение поведения объекта ofstream. Произвольный доступ к файлам	6,6
		<i>Консультации текущие</i>	0,75
		<i>Консультации перед экзаменом</i>	2
		<i>Экзамен</i>	0,2
2 семестр			
8	Основные понятия языка программирования Python	Особенности Python. Структура Python-программ. Структура файла и кодировка программы. Выполнение Python-программ. Ввод-вывод в Python.	8,5
9	Типы данных в Python	Тип данных и переменная. Классификация типов данных. Скалярные типы данных. Коллекции. Последовательности. Строки. Операции над строка-	20,5

№ п/п	Наименование раздела дисциплины	Содержание раздела	Трудоемкость раздела, акад. ч
		ми. Список. Кортеж. Числовой диапазон. Множества. Словари. Преобразование типов	
10	Операторы условия и цикла	Условный оператор. Циклы. Комбинация циклов и условий.	8,5
11	Подпрограммы	Функции. Глобальные и локальные функции. Анонимные функции	8,5
12	Ошибки и исключения	Разновидности ошибок. Поиск ошибок и отладка программы. Обработка исключений	8
13	Файлы и сериализация данных	Работа с файлами в Python. Сериализация и десериализация	8,5
14	Модули и пакеты	Основные понятия. Модули и пакеты в Python. Особенности модулей в Python	8,5
		<i>Консультации текущие</i>	0,9
		<i>Зачет</i>	0,1

5.2 Разделы дисциплины и виды занятий

№ п/п	Наименование раздела дисциплины	Лекции, ак. час	Практические занятия, ак. час	СРС, ак. час
1 семестр				
1	Основные понятия языка C++	2	4	5
2	Разветвляющиеся алгоритмы	2	2	5
3	Циклические алгоритмы. Функции	2	2	5
4	Обработка массивов в C++	4	14	2,35
5	Работа с символьными и строковыми данными в C++	2	2	3
6	Обработка структур в C++	2	4	2,3
7	Файловый ввод-вывод в C++	1	2	3,6
	<i>Консультации текущие</i>		0,75	
	<i>Консультации перед экзаменом</i>		2	
	<i>Экзамен</i>		0,2	
2 семестр				
8	Основные понятия языка программирования Python	2	4	2,5
9	Типы данных в Python	6	12	2,5
10	Операторы условия и цикла	2	4	2,5
11	Подпрограммы	2	4	2,5
12	Ошибки и исключения	2	4	2
13	Файлы и сериализация данных	2	4	2,5
14	Модули и пакеты	2	4	2,5
	<i>Консультации текущие</i>		0,9	
	<i>Зачет</i>		0,1	

5.2.1 Лекции

№ п/п	Наименование раздела дисциплины	Тематика лекционных занятий	Трудоемкость, ак. час
1 семестр			
1	Основные понятия языка C++	Структура программы на C++. Ввод и вывод данных. Представление данных	2
2	Разветвляющиеся алгоритмы	Условные операторы. Логические выражения. Оператор множественного выбора. Оператор перехода goto	2

№ п/п	Наименование раздела дисциплины	Тематика лекционных занятий	Трудоемкость, ак. час
3	Циклические алгоритмы. Функции	Операторы цикла. Функциональное программирование на C++	2
4	Обработка массивов в C++	Одномерные массивы и двумерные массивы. Инициализация. Доступ к элементам массива. Объявление и использование указателей в C++.	2
5	Обработка массивов в C++	Динамические массивы. Функции и массивы	2
6	Работа с символьными и строковыми данными в C++	Ввод-вывод символьных данных. Обработка символьных данных. Строки. Ввод-вывод строковых данных. Обработка строковых данных. Стандартные функции обработки строк. Передача строк в качестве параметров функций	2
7	Обработка структур в C++	Объявление шаблонов структур. Определение структур-переменных. Доступ к компонентам структуры. Инициализация структур. Копирование структур-переменных	2
8	Файловый ввод-вывод в C++	Файловый ввод-вывод. Функции состояния. Открытие файла для ввода и вывода. Изменение поведения объекта ofstream. Произвольный доступ к файлам	1
2 семестр			
1	Основные понятия языка программирования Python	Особенности Python. Структура Python-программ. Структура файла и кодировка программы. Выполнение Python-программ. Ввод-вывод в Python.	2
2	Типы данных в Python	Тип данных и переменная. Классификация типов данных. Скалярные типы данных	2
3	Типы данных в Python	Коллекции. Последовательности. Операции, общие для последовательностей. Строки. Операции над строками	2
4	Типы данных в Python	Список. Кортеж. Числовой диапазон. Множества. Словари. Преобразование типов	2
5	Операторы условия и цикла	Условный оператор. Циклы. Комбинация циклов и условий.	2
6	Подпрограммы	Функции. Глобальные и локальные функции. Анонимные функции	2
7	Ошибки и исключения	Разновидности ошибок. Поиск ошибок и отладка программы. Обработка исключений	2
8	Файлы и сериализация данных	Работа с файлами в Python. Сериализация и десериализация	2
9	Модули и пакеты	Основные понятия. Модули и пакеты в Python. Особенности модулей в Python	2

5.2.2 Практические занятия

№ п/п	Наименование раздела дисциплины	Тематика практических занятий (семинаров)	Трудоемкость, ак. час
1 семестр			
1	Основные понятия языка C++	Базовые алгоритмические структуры	2
2	Основные понятия языка C++	Исследование возможностей Microsoft Development Environment Visual C++ для отладки программ	2
3	Разветвляющиеся алгорит-	Реализация разветвляющихся алгоритмов в	2

№ п/п	Наименование раздела дисциплины	Тематика практических занятий (семинаров)	Трудоемкость, ак. час
	мы	C++	
4	Циклические алгоритмы. Функции	Программирование циклических алгоритмов в C++	2
5	Обработка массивов в C++	Описание и работа с одномерными массивами в языке C++	4
6	Обработка массивов в C++	Описание и работа с двумерными массивами в языке C++	4
7	Обработка массивов в C++	Объявление и использование указателей в C++	2
8	Обработка массивов в C++	Динамические массивы. Функции и массивы	4
9	Работа с символьными и строковыми данными в C++	Работа с символьными и строковыми данными в C++	2
10	Обработка структур в C++	Структуры в C++	4
11	Файловый ввод-вывод в C++	Файловый ввод-вывод в C++	2
2 семестр			
1	Основные понятия языка программирования Python	Выполнение Python-программ. Ввод-вывод в Python	4
2	Типы данных в Python	Скалярные типы данных	4
3	Типы данных в Python	Строки. Операции над строками	4
4	Типы данных в Python	Список. Кортеж. Числовой диапазон. Множества. Словари. Преобразование типов	4
5	Операторы условия и цикла	Условный оператор. Циклы. Комбинация циклов и условий.	4
6	Подпрограммы	Функции. Глобальные и локальные функции. Анонимные функции	4
7	Ошибки и исключения	Обработка исключений	4
8	Файлы и сериализация данных	Работа с файлами в Python. Сериализация и десериализация	4
9	Модули и пакеты	Работа с модулями и пакетами в Python	4

5.2.3 Лабораторный практикум не предусмотрен

5.2.4 Самостоятельная работа обучающихся (СРО)

№ п/п	Наименование раздела дисциплины	Вид СРО	Трудоемкость, ак. час
1	Основные понятия языка C++	Проработка конспекта лекций	2
		Проработка материалов по учебнику	1
		Подготовка к практическим занятиям	2
2	Разветвляющиеся алгоритмы	Проработка конспекта лекций	2
		Проработка материалов по учебнику	1
		Подготовка к практическим занятиям	2
3	Циклические алгоритмы. Функции	Проработка конспекта лекций	2
		Проработка материалов по учебнику	1
		Подготовка к практическим занятиям	2
4	Обработка массивов в C++	Проработка конспекта лекций	1
		Проработка материалов по учебнику	0,35
		Подготовка к практическим занятиям	1
5	Работа с символьными и строковыми данными в C++	Проработка конспекта лекций	1
		Проработка материалов по учебнику	1
		Подготовка к практическим занятиям	1
6	Обработка структур в C++	Проработка конспекта лекций	1
		Проработка материалов по учебнику	0,3
		Подготовка к практическим занятиям	1
7	Файловый ввод-вывод в C++	Проработка конспекта лекций	1,75
		Проработка материалов по учебнику	0,35
		Подготовка к практическим занятиям	1,5

№ п/п	Наименование раздела дисциплины	Вид СРО	Трудоемкость, ак. час
8	Основные понятия языка программирования Python	Проработка конспекта лекций	1
		Проработка материалов по учебнику	0,5
		Подготовка к практическим занятиям	1
9	Типы данных в Python	Проработка конспекта лекций	1
		Проработка материалов по учебнику	0,5
		Подготовка к практическим занятиям	1
10	Операторы условия и цикла	Проработка конспекта лекций	1
		Проработка материалов по учебнику	0,5
		Подготовка к практическим занятиям	1
11	Подпрограммы	Проработка конспекта лекций	0,5
		Проработка материалов по учебнику	1
		Подготовка к практическим занятиям	1
12	Ошибки и исключения	Проработка конспекта лекций	0,5
		Проработка материалов по учебнику	1
		Подготовка к практическим занятиям	0,5
13	Файлы и сериализация данных	Проработка конспекта лекций	1
		Проработка материалов по учебнику	1
		Подготовка к практическим занятиям	0,5
14	Модули и пакеты	Проработка конспекта лекций	1
		Проработка материалов по учебнику	0,5
		Подготовка к практическим занятиям	1

6 Учебно-методическое и информационное обеспечение дисциплины (модуля)

Для освоения дисциплины обучающийся может использовать:

6.1. Основная литература

1. Груздев, Д. В. Программирование C++ (1 курс) : учебное пособие / Д. В. Груздев. — Воронеж : ВГУ, 2017. — 80 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/154781> (дата обращения: 14.05.2021). — Режим доступа: для авториз. пользователей.

2. Тарланов, А. Т. Основы языка программирования Python : учебно-методическое пособие / А. Т. Тарланов, Ш. Г. Магомедов. — Москва : РТУ МИРЭА, 2019. — 107 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/171465> (дата обращения: 14.05.2021). — Режим доступа: для авториз. пользователей.

3. Косицин, Д. Ю. Язык программирования Python : учебно-методическое пособие / Д. Ю. Косицин. — Минск : БГУ, 2019. — 136 с. — ISBN 978-985-566-746-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/180546> (дата обращения: 14.05.2021). — Режим доступа: для авториз. пользователей.

6.2 Дополнительная литература

1. Бохан, К. А. Информационные технологии. Базовые типы и инструкции, функции языка C++ : учебное пособие / К. А. Бохан. — Рязань : РГРТУ, 2005. — 64 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/167940> (дата обращения: 14.05.2021). — Режим доступа: для авториз. пользователей.

2. Викентьева, О. Л. Проектирование программ и программирование на C++ : учебное пособие : в 2 частях / О. Л. Викентьева, А. Н. Гусин, О. А. Полякова. — Пермь :

ПНИПУ, [б. г.]. — Часть 1 : Структурное программирование — 2012. — 139 с. — ISBN 978-5-398-00761-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/160813> (дата обращения: 14.05.2021). — Режим доступа: для авториз. пользователей.

3. Широбокова, С. Н. Программирование на языке Python для лабораторных занятий : учебное пособие / С. Н. Широбокова, А. А. Кацупеев, А. В. Сулыз. — Новочеркасск : ЮРГПУ, 2020. — 104 с. — ISBN 978-5-9997-0725-3. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/180938> (дата обращения: 14.05.2021). — Режим доступа: для авториз. пользователей.

4. Полупанов, Д. В. Программирование в Python 3 : учебное пособие / Д. В. Полупанов, С. Р. Абдюшева, А. М. Ефимов. — Уфа : БашГУ, 2020. — 164 с. — ISBN 978-5-7477-5230-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/179915> (дата обращения: 14.05.2021). — Режим доступа: для авториз. пользователей.

6.3 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся

1. Медведкова И. Е. Программирование на языках высокого уровня [Текст] : методические указания по выполнению контрольной работы № 3 для студентов, обучающихся по направлениям 09.03.02 и 09.03.03, очной формы обучения / И. Е. Медведкова, С. В. Чикунов; ВГУИТ, Кафедра информационных технологий, моделирования и управления. - Воронеж, 2014. - 28 с.

6.4 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля)

Наименование ресурса сети «Интернет»	Электронный адрес ресурса
<i>Сайт научной библиотеки ВГУИТ</i>	http://biblos.vsu.ru/megapro/web
<i>Научная электронная библиотека</i>	https://elibrary.ru/defaultx.asp?
<i>Базовые федеральные образовательные порталы</i>	http://www.edu.ru/db/portal/sites/portal_page.htm
<i>Государственная публичная научно-техническая библиотека</i>	www.gpntb.ru/
<i>Информационно-коммуникационные технологии в образовании. Система федеральных образовательных порталов</i>	http://www.ict.edu.ru/
<i>Поисковая система «Яндекс»</i>	www.yandex.ru/
<i>Сайт ЭБС «Лань»</i>	http://e.lanbook.com
<i>Электронная информационно-образовательная среда ФГБОУ ВО «ВГУИТ»</i>	https://education.vsu.ru/

6.5 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения, современных профессиональных баз данных и информационных справочных систем

Используемые виды программного обеспечения:

Информационная среда для дистанционного обучения «Moodle», ОС Windows; MSOffice; Microsoft Visual Studio Professional 2010, Anaconda 3.

Локальная сеть университета и глобальная сеть Internet.

7. Материально-техническое обеспечение дисциплины

Для практических работ:

ауд. 336а - компьютерный класс каф. ВМиИТ: количество ПЭВМ – 9 (Corei3 540) Microsoft Windows Professional 7 Russian Upgrade Academic OPEN 1 License No Level #47881748 от 24.12.2010 г. <http://eopen.microsoft.com>;

Microsoft Office 2007, <http://eopen.microsoft.com>);

Для лекционных занятий используется лекционный аудиторный фонд университета и мультимедийное оборудование.

8 Оценочные материалы для промежуточной аттестации обучающихся по дисциплине

Оценочные материалы (ОМ) для дисциплины (модуля) включают в себя:

- перечень компетенций с указанием индикаторов достижения компетенций, этапов их формирования в процессе освоения образовательной программы;
- описание шкал оценивания;
- типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков;
- методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности.

ОМ представляются в виде отдельного документа и входят в состав рабочей программы дисциплины (модуля) в виде приложения.

Оценочные материалы формируются в соответствии с П ВГУИТ «Положение об оценочных материалах».

ПРИЛОЖЕНИЕ
к рабочей программе

1. Организационно-методические данные дисциплины для очно-заочной или заочной форм обучения

1.1 Объемы различных форм учебной работы и виды контроля в соответствии с учебным планом

Общая трудоемкость дисциплины составляет 5 зачетных единиц.

Виды учебной работы	Всего ак. ч	Распределение трудоемкости по семестрам, ак. ч	
		2	3
Общая трудоемкость дисциплины	180	108	72
Контактная работа, в т.ч. аудиторные занятия:	35,7	19,9	15,8
Лекции	12	6	6
<i>в том числе в форме практической подготовки</i>	-	-	-
Практические занятия	18	10	8
<i>в том числе в форме практической подготовки</i>	18	10	8
Консультации текущие	3,4	1,7	1,7
Консультации перед экзаменом	2	2	-
Вид аттестации (экзамен, зачет)	0,3	0,2	0,1
Самостоятельная работа:	133,6	81,3	52,3
Проработка материалов по конспекту лекций	30	20	10
Проработка материалов по учебнику	31	19	12
Подготовка к практическим занятиям	37,3	22,3	15
Выполнение контрольной работы	35,3	20	15,3
Контроль	10,7	6,8	3,9

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ**

по дисциплине
ПРОГРАММИРОВАНИЕ НА ЯЗЫКАХ ВЫСОКОГО УРОВНЯ

1. Перечень компетенций с указанием этапов их формирования

№ п/п	Код компетенции	Формулировка компетенции	Код и наименование индикатора достижения компетенции
1	УК-1	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	ИД2 _{УК-1} – Решает поставленные задачи, используя системный подход, на основе критического анализа и синтеза информации и оценивает последствия возможных решений
2	ПКв-2	Способность разрабатывать и адаптировать прикладное программное обеспечение	ИД3 _{ПК-2} – Демонстрирует навыки технического и рабочего проектирования компонентов информационных систем в соответствии со спецификой профиля подготовки
3	ПКв-8	Способность проводить тестирование компонентов программного обеспечения ИС	ИД1 _{ПК-8} – Выявление и исправление дефектов и несоответствий в архитектуре и дизайне ИС

Код и наименование индикатора достижения компетенции	Результаты обучения (показатели оценивания)
ИД2 _{УК-1} – Решает поставленные задачи, используя системный подход, на основе критического анализа и синтеза информации и оценивает последствия возможных решений	<p>Знает:</p> <ul style="list-style-type: none"> - способы ввода и вывода данных в C++ и Python, их достоинства и недостатки; - способы представления данных в C++ и Python, их достоинства и недостатки; - операторы языков в C++ и Python, их достоинства и недостатки; - основные структуры языков в C++ и Python, их достоинства и недостатки
	<p>Умеет:</p> <ul style="list-style-type: none"> - выбирать наиболее рациональные способы ввода и вывода данных; - выбирать наиболее рациональные способы представления данных; - выбирать наиболее рациональную структуру программы; - выбирать наиболее рациональный метод решения задачи
	<p>Владеет:</p> <ul style="list-style-type: none"> - навыками выбора наиболее рациональных способов решения задачи на языках C++ и Python на основе анализа их достоинств и недостатков
ИД3 _{ПК-2} – Демонстрирует навыки технического и рабочего проектирования компонентов информационных систем в соответствии со спецификой профиля подготовки	Знает основные виды алгоритмов проектирования компонентов информационных систем
	Умеет разрабатывать алгоритмы проектирования компонентов информационных систем
	Владеет основными способами разработки алгоритмов проектирования компонентов информационных систем
ИД1 _{ПКв-8} – Выявление и исправление дефектов и несоответствий в архитектуре и дизайне ИС	<p>Знает:</p> <ul style="list-style-type: none"> - основные методы разработки ИС на языках программирования C++ и Python; - методы отладки и тестирования ИС с целью выявления и исправления дефектов и несоответствий в архитектуре и дизайне
	<p>Умеет:</p> <ul style="list-style-type: none"> - использовать языки программирования C++ и Python для разработки ИС; - применять методы отладки и тестирования ИС с целью выявления и исправления дефектов и несоответствий в архитектуре и дизайне

Код и наименование индикатора достижения компетенции	Результаты обучения (показатели оценивания)
	Владеет навыками разработки ИС, их отладки и тестирования

2 Паспорт оценочных материалов по дисциплине

№ п/п	Разделы дисциплины	Индекс контролируемой компетенции (или ее части)	Оценочные средства		Технология оценивания (способ контроля)
			наименование	№№ заданий	
1	Основные понятия языка С++	УК-1	Вопросы к экзамену	1-3	Уровневая шкала
		УК-1	Тесты	1,2	Процентная шкала
2	Разветвляющиеся алгоритмы	УК-1	Вопросы к экзамену	4-7	Уровневая шкала
		УК-1	Тесты	3	Процентная шкала
3	Циклические алгоритмы. Функции	УК-1	Вопросы к экзамену	8-9	Уровневая шкала
		УК-1	Тесты	4	Процентная шкала
4	Обработка массивов в С++	УК-1	Вопросы к экзамену	10-15	Уровневая шкала
		УК-1	Тесты	5	Процентная шкала
5	Работа с символьными и строковыми данными в С++	ПКв-2	Вопросы к экзамену	16-21	Уровневая шкала
		ПКв-2	Тесты	6	Процентная шкала
6	Обработка структур в С++	ПКв-2	Вопросы к экзамену	22-26	Уровневая шкала
		ПКв-2	Тесты	7	Процентная шкала
7	Файловый ввод-вывод в С++	ПКв-8	Вопросы к экзамену	27-32	Уровневая шкала
		ПКв-2	Тесты	8	Процентная шкала
8	Основные понятия языка программирования Python	УК-1	Вопросы к зачету	32-36	Отметка в системе «зачтено-незачтено»
9	Типы данных в Python	УК-1	Вопросы к зачету	37-50	Отметка в системе «зачтено-незачтено»
		ПКв-2	Тесты	11,15	Процентная шкала
		ПКв-8	Кейс-задача	23	Уровневая шкала
10	Операторы условия и цикла	УК-1	Вопросы к зачету	51-54	Отметка в системе «зачтено-незачтено»
		ПКв-2	Тесты	12	Процентная шкала
11	Подпрограммы	ПКв-2	Вопросы к зачету	55-58	Отметка в системе «зачтено-незачтено»

№ п/п	Разделы дисциплины	Индекс контролируемой компетенции (или ее части)	Оценочные средства		Технология оценивания (способ контроля)
			наименование	№№ заданий	
		ПКв-8	Кейс-задача	24,25	Уровневая шкала
		ПКв-2	Тесты	13	Процентная шкала
12	Ошибки и исключения	ПКв-8	Вопросы к зачету	59-61	Отметка в системе «заче-но-незачтено»
		ПКв-2	Тесты	10	Процентная шкала
13	Файлы и сериализация данных	ПКв-8	Вопросы к зачету	62-64	Отметка в системе «заче-но-незачтено»
		ПКв-2	Тесты	14	Процентная шкала
14	Модули и пакеты	ПКв-8	Вопросы к зачету	65-69	Отметка в системе «заче-но-незачтено»
		ПКв-2	Тесты	9	Процентная шкала

3 Оценочные материалы для промежуточной аттестации. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Аттестация обучающегося по дисциплине проводится в форме тестирования (и предусматривает возможность последующей сдачи экзамена и зачета.

Каждый вариант теста включает 10 контрольных заданий, из них:

- 4 контрольных задания на проверку знаний;
- 3 контрольных задания на проверку умений;
- 3 контрольных задания на проверку навыков.

Каждый билет включает в себя:

- 2 вопроса на проверку знаний;
- 1 задание на проверку умений и навыков.

3.1 Тесты (тестовые задания)

3.1.1 УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач

№ задания	Тестовое задание с вариантами ответов и правильными ответами
1	Для чего предназначен оператор namespace: 1) для использования классов, переменных и функций из других модулей программы без использования заголовочных файлов 2) для заключения в группу объявлений классов, переменных и функций в отдельный контекст со своим именем 3) для заключения в группу объявлений классов, переменных и функций для использования только в текущем модуле 4) все ответы правильные
2	В чем различие использования следующих выражений #include <...> и #include «...»: 1) различие заключается в методе поиска препроцессором включаемого файла 2) в различии использования заголовочных и исходных файлов 3) нет различий 4) правильного ответа нет
3	Определите значение переменной a после выполнения фрагмента программы: int a=2; if (a<5) a=a+12; else a=a-7; 1) -5 2) 14 3) 5 4) 12
4	Чем отличается цикл do ... while от цикла while? 1) Цикл do ... while выполняется как минимум один раз 2) Цикл while выполняется до проверки условия 3) Цикл do ... while проверяет условие до выполнения 4) Ничем
5	Отметьте все правильные утверждения о массивах в языке C++. 1) элементы массива могут быть разных типов 2) все элементы массива должны быть одного типа 3) элементы в памяти расположены рядом 4) элементы могут быть расположены в памяти по одному 5) элементы всегда нумеруются с нуля

3.1.2 ПКв-2 Способность разрабатывать и адаптировать прикладное программное обеспечение

№ задания	Тестовое задание с вариантами ответов и правильными ответами
6	Что будет выведено на экран после выполнения программы? string s; s = "123"; s = s + "0" + s; s = s + s; cout << s; Ответ:12301231230123
7	Какие из следующих объявлений структур имеют правильный синтаксис? 1) struct sA {char a[12]; sA* this_struct}; 2) struct sA {int i; float f}; 3) struct sA {int j, sA this_struct}; 4) все
8	Какое из следующих утверждений верно для кода, представленного ниже? FILE *fp; char ch;

№ задания	Тестовое задание с вариантами ответов и правильными ответами								
	<pre>int i=1; fp = fopen("myfile.c", "r"); while((ch=getc(fp))!=EOF) { if(ch == '\n') i++;} fclose(fp);</pre> <p>1) Вычисляет количество строк в файле 2) Вычисляет количество слов в файле 3) Вычисляет количество символов в файле 4) Вычисляет количество пустых строк в файле</p>								
9	<p>Для чего в пакетах модулей python в файле __init__.py служит список __all__?</p> <p>1) Для конструкторов классов, как и всё, что связано с __init__ 2) Список определяет, что экспортировать, когда происходит импорт с помощью from * 3) Для перечисления переменных, которые будут скрыты для импортирования. 4) Такого списка нет</p>								
10	<p>Что выведет следующий код?</p> <pre>try: raise IndexError except IndexError: print('Получено исключение.') else: print('Но в этом нет ничего страшного.')</pre> <p>Ответ: Получено исключение.</p>								
11	<p>Установите соответствие между именем функции и ее назначением</p> <table border="0"> <tr> <td>1) len(s)</td> <td>3) преобразует строку s к нижнему регистру</td> </tr> <tr> <td>2) s.isdigit()</td> <td>1) возвращает длину строки s</td> </tr> <tr> <td>3) s.lower()</td> <td>2) определяет, состоит ли строка из цифр</td> </tr> <tr> <td>4) s.capitalize()</td> <td>4) переводит первый символ строки s в верхний регистр, а все остальные в нижний</td> </tr> </table>	1) len(s)	3) преобразует строку s к нижнему регистру	2) s.isdigit()	1) возвращает длину строки s	3) s.lower()	2) определяет, состоит ли строка из цифр	4) s.capitalize()	4) переводит первый символ строки s в верхний регистр, а все остальные в нижний
1) len(s)	3) преобразует строку s к нижнему регистру								
2) s.isdigit()	1) возвращает длину строки s								
3) s.lower()	2) определяет, состоит ли строка из цифр								
4) s.capitalize()	4) переводит первый символ строки s в верхний регистр, а все остальные в нижний								
12	<p>Определите, чему будет равна сумма ряда $\sum_{n=1}^{10} (n+4)x^{n-1}$ при $x=3$</p> <p>Ответ: 398579</p>								
13	<p>Установите соответствие между функциями и выполняемыми ими действиями</p> <table border="0"> <tr> <td>1) help()</td> <td>2) выводит список методов и атрибутов объекта</td> </tr> <tr> <td>2) dir()</td> <td>1) выводит справку по объекту, доступную из строк документации</td> </tr> <tr> <td>3) type()</td> <td>4) определяет, является ли объект экземпляром указанного класса (классов) или его подкласса.</td> </tr> <tr> <td>4) isinstance()</td> <td>3) возвращает тип объекта</td> </tr> </table>	1) help()	2) выводит список методов и атрибутов объекта	2) dir()	1) выводит справку по объекту, доступную из строк документации	3) type()	4) определяет, является ли объект экземпляром указанного класса (классов) или его подкласса.	4) isinstance()	3) возвращает тип объекта
1) help()	2) выводит список методов и атрибутов объекта								
2) dir()	1) выводит справку по объекту, доступную из строк документации								
3) type()	4) определяет, является ли объект экземпляром указанного класса (классов) или его подкласса.								
4) isinstance()	3) возвращает тип объекта								
14	<p>С помощью Python нужно записать данные в файл, но только в том случае, если файла ещё нет. Какой режим указать в инструкции open()?</p> <p>1) 'x' 2) Никакой. Нужна предварительная проверка os.path.exists() 3) 'w' 4) 'r'</p>								
15	<p>Сколько раз можно дублировать один и тот же элемент в множестве?</p> <p>1) сколько угодно 2) от 3 и более раз 3) ровно 2 раза дублирование невозможно</p>								

3.1.3 ПКв-8 Способность проводить тестирование компонентов программного обеспечения ИС

№ задания	Тестовое задание с вариантами ответов и правильными ответами
16	Чему будет равен результат вычисления выражения: int d=5; bool b = true, c; c = (!b (d>3)); Ответ: true
17	Что из себя представляет динамическое выделение памяти: 1) память под объект (переменную) может выделяться не сразу, а в процессе работы программы, освобождение памяти производится вручную 2) память под объект (переменную) может выделяться не сразу, а в процессе работы программы, освобождение памяти производится автоматически после завершения программы 3) память под объект (переменную) выделяется каждый раз при обращении к переменной 4) Правильного ответа нет
18	Чему равно значение выражения (a && ! b c), где a, b и c - величины типа bool, имеющие значения false, true и true соответственно Ответ: true
19	Какие выражения не содержат синтаксических ошибок: 1) sin(abs(0.6(e*3)) 2) ((cos(3*a+1.*abs(x)))) 3) a*exp(t)\(2t) 4) a=b*exp(3.0)*5
20	Деятельность, направленная на обнаружение и исправление ошибок в ПО с использованием процессов выполнения его программ – это ____ Ответ: отладка
21	Способность программного обеспечения быть перенесенным из одного окружения в другое - это ____ Ответ: мобильность
22	Процесс выполнения программ на некотором наборе данных, для которого заранее известен результат применения или известны правила поведения этих программ – это ____ Ответ: тестирование

3.2 Кейс-задания

3.2.1 ПКв-8 Способность проводить тестирование компонентов программного обеспечения ИС

№ задания	Условие задачи (формулировка задания)
23	Создать 2 пустых списка, содержащих числа: 1) [4.5, 100, 3.4, 100, 8.7, 1.3] 2) [200, 14.5, 200, 3.4, 8.7, 11.3] a = # Удалите комментарий и допишите код b = # Удалите комментарий и допишите код # Добавить 2 вещественных числа (4.5 и 3.4) в список 'a' с помощью append a.append(# Удалите комментарий и допишите код) a.append(# Удалите комментарий и допишите код) # Добавить 2 вещественных числа (8.7, 1.3) в список 'a' с помощью extend a.extend(# Удалите комментарий и допишите код) # Добавить 2 вещественных числа (14.5, 3.4) в список 'b' с помощью append # Удалите комментарий и допишите код # Добавить 2 вещественных числа (8.7, 11.3) в список 'b' с помощью extend # Удалите комментарий и допишите код # Вставить целое число 100 на 2-е и 4-е место списка 'a' a.insert(# Удалите комментарий и допишите код)

№ задания	Условие задачи (формулировка задания)
	<pre> a.insert(# Удалите комментарий и допишите код) # Вставить целое число 200 на 1-е и 3-е место списка 'b' # Удалите комментарий и допишите код # Вывести списки на экран print("Исходные списки:") print(# Удалите комментарий и допишите код) print(# Удалите комментарий и допишите код) # Удалить первые элементы из списков 'a' и 'b' del # Удалите комментарий и допишите код del # Удалите комментарий и допишите код # Удалить элемент 100 из списка 'a' a.remove(# Удалите комментарий и допишите код) # Удалить элемент 200 из списка 'b' # Удалите комментарий и допишите код # Вывести списки на экран print("\nПосле удалений:") # Удалите комментарий и допишите код # Создать множества из списков 'a' и 'b', а также их пересечение sa = set(a) sb = # Удалите комментарий и допишите код sa_and_sb = # Удалите комментарий и допишите код # Вывести экран уникальные элементы каждого списка и общие print("\nУникальные элементы:") # Удалите комментарий и допишите код print("общие:", sa_and_sb) # Соединить списки 'a' и 'b' c = # Удалите комментарий и допишите код # Отсортировать список 'c' по возрастанию и убыванию c_asc = # Удалите комментарий и допишите код c_desc = # Удалите комментарий и допишите код # Среднее арифметическое элементов списка 'c', расположенных на четных местах sr_ar = # Удалите комментарий и допишите код # Среднее геометрическое элементов списка 'c', расположенных на нечетных местах sr_geom = # Удалите комментарий и допишите код # Максимальный и минимальный элементы c_max = # Удалите комментарий и допишите код c_min = # Удалите комментарий и допишите код # Вывести результаты на экран print("\nИтоговые:") print("3-й:", c) # Удалите комментарий и допишите код Вывод: Исходные списки: [4.5, 100, 3.4, 100, 8.7, 1.3] [200, 14.5, 200, 3.4, 8.7, 11.3] </pre>

№ задания	Условие задачи (формулировка задания)
	<p>После удалений: [3.4, 100, 8.7, 1.3] [14.5, 3.4, 8.7, 11.3]</p> <p>Уникальные элементы: {8.7, 1.3, 3.4, 100} {8.7, 11.3, 3.4, 14.5} общие: {8.7, 3.4}</p> <p>Итоговые: 3-й: [3.4, 100, 8.7, 1.3, 14.5, 3.4, 8.7, 11.3] [1.3, 3.4, 3.4, 8.7, 8.7, 11.3, 14.5, 100] [100, 14.5, 11.3, 8.7, 8.7, 3.4, 3.4, 1.3] Ср. арифм. = 29.00, ср. геометр. = 7.82 Макс. и мин.: 100 1.3</p>
24	<pre># В данной задаче все значения задаются в коде (без input()) # Оформить все задания в виде отдельных функций (подпрограмм) # 1. Создание словаря # Создать пустой словарь info = # Удалите комментарий и допишите код # Добавить значения для ключей "фио", "дата_рождения", "место_рождения" info["фио"] = # Удалите комментарий и допишите код # Удалите комментарий и допишите код # Удалите комментарий и допишите код # Напечатать словарь print(info) # Создать ключ "хобби" со значением = список из строк - # наименований хобби (например "плавание" и т.д.) # Удалите комментарий и допишите код # Добавить "программирование" в список хобби # Удалите комментарий и допишите код # Создать ключ "животные" со значением = кортеж из строк - # имен домашних животных (например "кошка Мурка" и т.д.) # Удалите комментарий и допишите код # Создать ключ "ЕГЭ" и поместить в него пустой словарь # Удалите комментарий и допишите код # В словарь info["ЕГЭ"] добавить информацию о сданных экзаменах, # где ключ - наименование предмета, а значение - количество баллов # Удалите комментарий и допишите код # Добавить экзамен, который не был сдан, после чего удалить его # Удалите комментарий и допишите код # Создать ключ "вузы" и поместить в него пустой словарь # Удалите комментарий и допишите код # В словарь info["вузы"] добавить информацию о вузах, # где ключ - аббревиатура вуза, а значение - # количество баллов для специальности, на которую хотели поступить # Удалите комментарий и допишите код # 2. Вывод на экран print("Данные:")</pre>

№ задания	Условие задачи (формулировка задания)
	<pre> # Получившийся словарь # Удалите комментарий и допишите код # Список экзаменов ЕГЭ в алфавитном порядке # (используйте метод ``keys()``, преобразовав результат в кортеж) exams = # Удалите комментарий и допишите код print("Предметы:", exams) # Список вузов в алфавитном порядке uni = # Удалите комментарий и допишите код print("Вузы:", uni) # 3. Ответы на вопросы print("\nОтветы на вопросы:") # Выделить имя из info["фио"] name = # Удалите комментарий и допишите код # Начинается на гласную? (True/False) starts_with_vowel = # Удалите комментарий и допишите код print("** мое имя начинается на гласную букву:", starts_with_vowel) # Выделить месяц из info["дата_рождения"] month = # Удалите комментарий и допишите код # Родился зимой или летом? (True/False) born_in_winter_or_summer = # Удалите комментарий и допишите код print("** родился летом или зимой:", # Удалите комментарий и допишите код) # Количество хобби и первое из них hobbies_count = # Удалите комментарий и допишите код print("** у меня {} хобби, первое {}".format(# Удалите комментарий и допишите код)) # Количество сданных экзаменов print("** после окончания школы сдавал {} экз.".format(# Удалите комментарий и допишите код)) # Сумма баллов по экзаменам sum_mark = # Удалите комментарий и допишите код print("** сумма баллов = {}".format(# Удалите комментарий и допишите код)) # Максимальный балл среди сданных max_mark = # Удалите комментарий и допишите код print("** макс. балл = {}".format(# Удалите комментарий и допишите код)) # Количество вузов, в которые Вы проходите по баллам # Подсказка: определить, проходите Вы или нет, можно простым сравнением # суммы баллов с проходным баллом вуза - ``True/False``. # Для того, чтобы определить количество таких вузов, преобразуйте # сравнение в целое число (используя ``int()``) и сложите все сравниваемые вузы. vuz_count = # Удалите комментарий и допишите код print("** кол-во вузов, в которые прохожу: {}".format(# Удалите комментарий и допишите код)) Пример вывода: {'фио': 'Иванов Иван Иванович', 'место_рождения': 'Воронеж', 'дата_рождения': '01/09/2003'} Данные: {'фио': 'Иванов Иван Иванович', 'животные': ('кошка Мурка'), 'вузы': {'МИРЭА': 240, 'МГУ': 295, 'МГТУ им. Баумана': 255}, 'хобби': ['игра на гитаре', 'туризм', 'программирование'], 'ЕГЭ': {'Математика': 90, 'Информатика': 88, 'Русский язык': 67}, 'дата_рождения': '01/09/2003', 'место_рождения': 'Воронеж'} </pre>

№ задания	Условие задачи (формулировка задания)
	<p>Предметы: Информатика, Математика, Русский язык Вузы: МГТУ им. Баумана, МГУ, МИРЭА</p> <p>Ответы на вопросы: * мое имя начинается на гласную букву: True * родился летом или зимой: False * у меня 3 хобби, первое "игра на гитаре" * после окончания школы сдавал 3 экз. * сумма баллов = 245 * макс. балл = 90 * кол-во вузов, в которые прохожу: 1</p>
25	<p>Вставьте в правильное место программы нужные строки кода:</p> <ol style="list-style-type: none"> 1) 2) 3) <pre>def non_negatives(nums): """В новый список чисел 'new_nums' записать только положительные элементы списка 'nums' и вывести его на печать """ new_nums=[] for i in range(len(nums)): if nums[i] > 0: 4) print (new_nums) non_negatives(nums)</pre> <p>Строки кода:</p> <pre>new_nums.append(nums[i]) n = 10 import random nums = [round(random.uniform(-10, 10), 2) for i in range(n)]</pre> <p>Ответ:</p> <ol style="list-style-type: none"> 1) import random 2) n = 10 3) nums = [round(random.uniform(-10, 10), 2) for i in range(n)] 4) new_nums.append(nums[i])

3.3.Экзамен и зачет

3.3.1 УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач

Вопросы для экзамена

Номер вопроса (задачи, задания)	Текст вопроса
1.	Структура программы на C++
2.	Ввод и вывод данных
3.	Представление данных
4.	Условные операторы
5.	Логические выражения
6.	Оператор множественного выбора
7.	Оператор перехода goto
8.	Операторы цикла

Номер вопроса (задачи, задания)	Текст вопроса
9.	Функциональное программирование на C++
10.	Одномерные массивы и двумерные массивы
11.	Инициализация массивов
12.	Доступ к элементам массива
13.	Объявление и использование указателей в C++
14.	Динамические массивы
15.	Функции и массивы

3.3.2 ПКв-2 Способность разрабатывать и адаптировать прикладное программное обеспечение

Вопросы для экзамена

Номер вопроса (задачи, задания)	Текст вопроса
16.	Ввод-вывод символьных данных
17.	Обработка символьных данных
18.	Строки. Ввод-вывод строковых данных
19.	Обработка строковых данных
20.	Стандартные функции обработки строк
21.	Передача строк в качестве параметров функций
22.	Объявление шаблонов структур
23.	Определение структур-переменных
24.	Доступ к компонентам структуры
25.	Инициализация структур
26.	Копирование структур-переменных

3.3.3 ПКв-8 Способность проводить тестирование компонентов программного обеспечения ИС

Вопросы для экзамена

Номер вопроса (задачи, задания)	Текст вопроса
27.	Файловый ввод-вывод
28.	Изменение поведения объекта ofstream
29.	Функции состояния
30.	Открытие файла для ввода и вывода
31.	Основные методы тестирования программного кода

3.3.4 УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач

Вопросы для зачета

Номер вопроса (задачи, задания)	Текст вопроса
32	Особенности Python
33	Структура Python-программ
34	Структура файла и кодировка программы

Номер вопроса (задачи, задания)	Текст вопроса
35	Выполнение Python-программ
36	Ввод-вывод в Python
37	Тип данных и переменная
38	Классификация типов данных
39	Коллекции
40	Скалярные типы данных
41	Последовательности
42	Операции, общие для последовательностей
43	Строки
44	Операции над строками
45	Список
46	Кортеж
47	Числовой диапазон
48	Множества
49	Словари
50	Преобразование типов
51	Условный оператор
52	Циклы
53	Глобальные и локальные переменные
54	Лямбда-функция

3.3.5 ПКв-2 Способность разрабатывать и адаптировать прикладное программное обеспечение

Вопросы для зачета

Номер вопроса (задачи, задания)	Текст вопроса
55	Комбинация циклов и условий
56	Функции
57	Глобальные функции
58	Анонимные функции

3.3.6 ПКв-8 Способность проводить тестирование компонентов программного обеспечения ИС

Вопросы для зачета

Номер вопроса (задачи, задания)	Текст вопроса
59	Разновидности ошибок
60	Поиск ошибок и отладка программы
61	Обработка исключений
62	Работа с файлами в Python
63	Сериализация
64	Десериализация
65	Модули и пакеты в Python
66	Особенности модулей в Python

Номер вопроса (задачи, задания)	Текст вопроса
67	Тернарные операторы
68	Типы объектов
69	Локальные функции

4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

Процедуры оценивания в ходе изучения дисциплины знаний, умений и навыков, характеризующих этапы формирования компетенций, регламентируются положениями:

- П ВГУИТ 2.4.03 Положение о курсовых экзаменах и зачетах;
- П ВГУИТ 4.1.02 Положение о рейтинговой оценке текущей успеваемости.

5. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания для каждого результата обучения по дисциплине/практике

Результаты обучения по этапам формирования компетенций	Предмет оценки (продукт или процесс)	Показатель оценивания	Критерии оценивания сформированности компетенций	Шкала оценивания	
				Академическая оценка или баллы	Уровень освоения компетенции
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач					
<p>Знать:</p> <ul style="list-style-type: none"> - способы ввода и вывода данных в C++ и Python, их достоинства и недостатки; - способы представления данных в C++ и Python, их достоинства и недостатки; - операторы языков в C++ и Python, их достоинства и недостатки; - основные структуры языков в C++ и Python, их достоинства и недостатки <p>Уметь:</p> <ul style="list-style-type: none"> - выбирать наиболее рациональные способы ввода и вывода данных; - выбирать наиболее рациональные способы представления данных; - выбирать наиболее рациональную структуру программы; - выбирать наиболее рациональный метод решения задачи <p>Владеть:</p>	Тесты (тестовые задания)	<p>Знает:</p> <ul style="list-style-type: none"> - особенности Python; - структуру Python-программ; - типы данных; - операторы; - методы функционального и модульного программирования; - методы работы с файлами; - методы обработки ошибок и исключений <p>Умеет:</p> <ul style="list-style-type: none"> - выбирать наиболее рациональные способы ввода и вывода данных; - выбирать наиболее рациональные способы 	Студент ответил на 0- 66 % вопросов теста	2 балла	Не освоена (недостаточный)
			Студент ответил на 66,1- 75 % вопросов теста	3 балла	Освоена (базовый)
			Студент правильно ответил на 75,1 - 85 % вопросов теста	4 балла	Освоена (повышенный)
			Студент правильно ответил на 85,1-100 % вопросов теста	5 баллов	Освоена (повышенный)
	Вопросы к экзамену		Студент не ответил на все вопросы, допустил более 3 ошибок	неудовлетв.	Не освоена (недостаточный)
			Студент ответил не на все вопросы, но в тех, на которые дал ответ не допустил ошибки	удовлетвор.	Освоена (базовый)
			Студент ответил на все вопросы и допустил более 1 ошибки, но менее 3 ошибок	хорошо	Освоена (повышенный)
			Студент ответил на все вопросы и допустил не более 1 ошибки в ответе	отлично	Освоена (повышенный)
			Вопросы к зачету	Студент не ответил на все вопросы, допустил более 3 ошибок	Не зачтено

Результаты обучения по этапам формирования компетенций	Предмет оценки (продукт или процесс)	Показатель оценивания	Критерии оценивания сформированности компетенций	Шкала оценивания	
				Академическая оценка или баллы	Уровень освоения компетенции
- навыками выбора наиболее рациональных способов решения задачи на языках C++ и Python на основе анализа их достоинств и недостатков		представления данных; - выбирать наиболее рациональную структуру программы; - выбирать наиболее рациональный метод решения задачи Владеет навыками выбора наиболее рациональных способов решения задачи на языках C++ и Python на основе анализа их достоинств и недостатков	Студент ответил на все вопросы, допустил не более 3 ошибок в ответах	Зачтено	Освоена (базовый)
ПКв-2 Способность разрабатывать и адаптировать прикладное программное обеспечение					
Знать основные виды алгоритмов проектирования компонентов информационных систем Уметь разрабатывать алгоритмы проектирования компонентов информационных систем Владеть основными способами разработки алгоритмов проектирования компонентов информа-	Тесты (тестовые задания)	Знает основные виды алгоритмов проектирования компонентов информационных систем Умеет разрабатывать алгоритмы проектирования компонентов информационных	Студент ответил на 0- 66 % вопросов теста	2 балла	Не освоена (недостаточный)
			Студент ответил на 66,1- 75 % вопросов теста	3 балла	Освоена (базовый)
			Студент правильно ответил на 75,1 - 85 % вопросов теста	4 балла	Освоена (повышенный)
			Студент правильно ответил на 85,1-100 % вопросов теста	5 баллов	Освоена (повышенный)
	Вопросы к экзамену		Студент не ответил на все вопросы, допустил более 3 ошибок	неудовлетв.	Не освоена (недостаточный)

Результаты обучения по этапам формирования компетенций	Предмет оценки (продукт или процесс)	Показатель оценивания	Критерии оценивания сформированности компетенций	Шкала оценивания		
				Академическая оценка или баллы	Уровень освоения компетенции	
информационных систем		систем Владеет основными способами разработки алгоритмов проектирования компонентов информационных систем	Студент ответил не на все вопросы, но в тех, на которые дал ответ не допустил ошибки	удовлетвор.	Освоена (базовый)	
			Студент ответил на все вопросы и допустил более 1 ошибки, но менее 3 ошибок	хорошо	Освоена (повышенный)	
			Студент ответил на все вопросы и допустил не более 1 ошибки в ответе	отлично	Освоена (повышенный)	
			Студент не ответил на все вопросы, допустил более 3 ошибок	Не зачтено	Не освоена (недостаточный)	
	Студент ответил на все вопросы, допустил не более 3 ошибок в ответах		Зачтено	Освоена (базовый)		
	Вопросы к зачету					
ПКе-8 Способность проводить тестирование компонентов программного обеспечения ИС						
Знать: - основные методы разработки ИС на языках программирования C++ и Python; - методы отладки и тестирования ИС с целью выявления и исправления дефектов и несоответствий в архитектуре и дизайне Уметь: - использовать языки программирования C++ и Python для разработки ИС; - применять методы отладки и тестирования ИС с целью выявления и исправления дефектов и несоответствий в архитектуре и дизайне Владеть навыками разработки ИС, их отладки и тестирования	Тесты (тестовые задания)	Знает - основные методы разработки ИС на языках программирования C++ и Python; - методы отладки и тестирования ИС с целью выявления и исправления дефектов и несоответствий в архитектуре и дизайне Умеет - использовать языки программирования C++ и Python для разработки ИС;	Студент ответил на 0- 66 % вопросов теста	2 балла	Не освоена (недостаточный)	
			Студент ответил на 66,1- 75 % вопросов теста	3 балла	Освоена (базовый)	
			Студент правильно ответил на 75,1 - 85 % вопросов теста	4 балла	Освоена (повышенный)	
			Студент правильно ответил на 85,1-100 % вопросов теста	5 баллов	Освоена (повышенный)	
	Вопросы к экзамену			Студент не ответил на все вопросы, допустил более 3 ошибок	неудовлетв.	Не освоена (недостаточный)
				Студент ответил не на все вопросы, но в тех, на которые дал ответ не допустил ошибки	удовлетвор.	Освоена (базовый)
				Студент ответил на все вопросы и допустил более 1 ошибки, но менее 3 ошибок	хорошо	Освоена (повышенный)
				Студент ответил на все вопросы и допустил не более 1 ошибки в ответе	отлично	Освоена (повышенный)

Результаты обучения по этапам формирования компетенций	Предмет оценки (продукт или процесс)	Показатель оценивания	Критерии оценивания сформированности компетенций	Шкала оценивания	
				Академическая оценка или баллы	Уровень освоения компетенции
	Вопросы к зачету	- применять методы отладки и тестирования ИС с целью выявления и исправления дефектов и несоответствий в архитектуре и дизайне Владеет навыками разработки ИС, их отладки и тестирования	Студент не ответил на все вопросы, допустил более 3 ошибок	Не зачтено	Не освоена (недостаточный)
			Студент ответил на все вопросы, допустил не более 3 ошибок в ответах	Зачтено	Освоена (базовый)
	Кейс-задания		Обучающийся неправильно выполнил задание или ответил не на все вопросы, или допустил более 5 ошибок в ответе	Неудовл.	Не освоена (недостаточный)
			Обучающийся правильно выполнил задание, ответил на все вопросы, допустил не более 5 ошибок в ответе	Удовл.	Освоена (базовый)
			Обучающийся правильно выполнил задание, ответил на все вопросы, допустил не более 2 ошибок в ответе	Хорошо	Освоена (повышенный)
			Обучающийся правильно выполнил задание, ответил на все вопросы, допустил не более 1 ошибки в ответе	Отлично	Освоена (повышенный)