

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
СТАВРОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ**

**УТВЕРЖДАЮ**

Директор/Декан  
факультета цифровых технологий  
Шлаев Дмитрий Валерьевич

\_\_\_\_\_  
«\_\_» \_\_\_\_\_ 20\_\_ г.

**Рабочая программа дисциплины**

**Б1.О.11 Программирование высокопроизводительных систем**

09.04.03 Прикладная информатика

Искусственный интеллект в кибербезопасности

магистр

очная

## 1. Цель дисциплины

Целью освоение дисциплины является овладение основными принципами программирования на высокоуровневом языке Python и прикладными аспектами его применения. развитие у обучающихся алгоритмического мышления, формирование навыков реализации различных алгоритмов на высокоуровневом языке программирования, применение знаний к решению прикладных задач.

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

Процесс изучения дисциплины направлен на формирование следующих компетенций ОП ВО и овладение следующими результатами обучения по дисциплине:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Перечень планируемых результатов обучения по дисциплине
ОПК-2 Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач	ОПК-2.3 Разрабатывает оригинальные алгоритмы и программные средства, в том числе с использованием интеллектуальных технологий, для решения профессиональных задач	<b>знает</b> <b>умеет</b> <b>владеет навыками</b>
ПК-1 Способен разрабатывать, модернизировать и сопровождать программное обеспечение информационных систем	ПК-1.1 Разрабатывает программный код на современных языках программирования, применяя методы и приёмы формализации и алгоритмизации поставленных задач	<b>знает</b> <b>умеет</b> <b>владеет навыками</b>

## 3. Место дисциплины в структуре образовательной программы

Дисциплина «Программирование высокопроизводительных систем» является дисциплиной обязательной части программы.

Изучение дисциплины осуществляется в 2семестре(-ах).

Для освоения дисциплины «Программирование высокопроизводительных систем» студенты используют знания, умения и навыки, сформированные в процессе изучения дисциплин:

Инфокоммуникационные системы и сети

Информационные системы

Программно-аппаратная защита информации

Электроника

Ознакомительная практика

Технологии программирования

Теория информационных процессов и систем  
Современные интеллектуальные системы и технологии

Инфокоммуникационные системы и сети  
 Информационные системы  
 Программно-аппаратная защита информации  
 Электроника  
 Ознакомительная практика  
 Технологии программирования  
 Теория информационных процессов и систем  
 Методы моделирования и анализа сложных систем

Инфокоммуникационные системы и сети  
 Информационные системы  
 Программно-аппаратная защита информации  
 Электроника  
 Ознакомительная практика  
 Технологии программирования  
 Теория информационных процессов и систем  
 Современные системы управления базами данных

Освоение дисциплины «Программирование высокопроизводительных систем» является необходимой основой для последующего изучения следующих дисциплин:

Преддипломная практика  
 Прикладные системы искусственного интеллекта  
 Инженерия серверных веб-систем  
 Подготовка к сдаче и сдача государственного экзамена  
 Подготовка к процедуре защиты и защита выпускной квалификационной работы  
 Технологическая (проектно-технологическая) практика  
 Проектирование автономных ИИ-агентов  
 Технологии разработки защищенного ПО  
 Атаки на системы искусственного интеллекта

**4. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу с обучающимися с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся**

Общая трудоемкость дисциплины «Программирование высокопроизводительных систем» в соответствии с рабочим учебным планом и ее распределение по видам работ представлены ниже.

Семестр	Трудоемкость час/з.е.	Контактная работа с преподавателем, час			Самостоятельная работа, час	Контроль, час	Форма промежуточной аттестации (форма контроля)
		лекции	практические занятия	лабораторные занятия			
2	144/4	12	24		72	36	Эк
в т.ч. часов: в интерактивной форме		4	6				

Семестр	Трудоемкость час/з.е.	Внеаудиторная контактная работа с преподавателем, час/чел					
		Курсовая работа	Курсовой проект	Зачет	Дифференцированный зачет	Консультации перед экзаменом	Экзамен
2	144/4						0.25

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

№	Наименование раздела/темы	Семестр	Количество часов					Формы текущего контроля успеваемости и промежуточной аттестации	Оценочное средство проверки результатов достижения индикаторов компетенций	Код индикаторов достижения компетенций
			всего	Лекции	Семинарские занятия		Самостоятельная работа			
					Практические	Лабораторные				
1.	1 раздел. Программирование высокопроизводительных систем									
1.1.	Базовые конструкции Python	2	4	2	2		14	Задачи		
1.2.	Коллекции и работа с памятью	2	12	4	8		12	Задачи		
1.3.	КТ 1	2	2		2			КТ 1	Тест	
1.4.	Функции и их особенности в Python	2	4	2	2		12	Задачи		
1.5.	Объектно-ориентированное программирование	2	4	2	2		14	Задачи		
1.6.	КТ 2	2	2		2			КТ 2	Тест	
1.7.	Библиотеки для получения и обработки данных	2	8	2	6		20	Задачи		
	Промежуточная аттестация		Эк							
	Итого		144	12	24		72			
	Итого		144	12	24		72			

**5.1. Лекционный курс с указанием видов интерактивной формы проведения занятий**

Тема лекции (и/или наименование раздел) (вид интерактивной формы проведения занятий)/ (практическая подготовка)	Содержание темы (и/или раздела)	Всего, часов / часов интерактивных занятий/ практическая подготовка
Базовые конструкции Python	Условный оператор Циклы Вложенные циклы	2/2
Коллекции и работа с памятью	Строки, кортежи, списки. Множества, словари	2/2
Коллекции и работа с памятью	Встроенные возможности по работе с коллекциями	2/-
Функции и их особенности в Python	Функции и их особенности в Python	2/-
Объектно-ориентированное программирование	Объектно-ориентированное программирование	2/-
Библиотеки для получения и обработки данных	Библиотеки для получения и обработки данных	2/-
Итого		12

### 5.2.1. Семинарские (практические) занятия с указанием видов проведения занятий в интерактивной форме

Наименование раздела дисциплины	Формы проведения и темы занятий (вид интерактивной формы проведения занятий)/(практическая подготовка)	Всего, часов / часов интерактивных занятий/ практическая подготовка	
		вид	часы
Базовые конструкции Python	Условный оператор Циклы Вложенные циклы	Пр	2/-/-
Коллекции и работа с памятью	Строки, кортежи, списки. Множества, словари	Пр	4/-/-
Коллекции и работа с памятью	Встроенные возможности по работе с коллекциями. Списочные выражения. Модель памяти для типов языка Python	Пр	4/2/-
КТ 1	КТ 1	Пр	2/-/-
Функции и их особенности в Python	Функции. Области видимости. Передача параметров в функции Позиционные и именованные аргументы. Функции высших порядков. Лямбда-функции Рекурсия. Декораторы. Генераторы	Пр	2/-/-
Объектно-ориентированное программирование	Объектная модель Python. Классы, поля и методы Волшебные методы, переопределение методов. Наследование Модель исключений Python. Try, except, else, finally. Модули	Пр	2/2/-
КТ 2	КТ 2	Пр	2/-/-
Библиотеки для получения и обработки данных	Модули math и numpy Модуль pandas Модуль requests	Пр	2/-/-
Библиотеки для получения и обработки данных	Модуль pandas	Пр	2/2/-
Библиотеки для получения и обработки данных	Модуль requests	Пр	2/-/-
Итого			

### 5.3. Курсовой проект (работа) учебным планом не предусмотрен

### 5.4. Самостоятельная работа обучающегося

Темы и/или виды самостоятельной работы	Часы

Условный оператор Циклы Вложенные циклы	14
Встроенные возможности по работе с коллекциями. Списочные выражения. Модель памяти для типов языка Python	12
Лямбда-функции Рекурсия. Декораторы. Генераторы	12
Лямбда-функции Рекурсия. Декораторы. Генераторы	14
Библиотеки для получения и обработки данных	20

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

Учебно-методическое обеспечение для самостоятельной работы обучающегося по дисциплине «Программирование высокопроизводительных систем» размещено в электронной информационно-образовательной среде Университета и доступно для обучающегося через его личный кабинет на сайте Университета. Учебно-методическое обеспечение включает:

1. Рабочую программу дисциплины «Программирование высокопроизводительных систем».
2. Методические рекомендации для организации самостоятельной работы обучающегося по дисциплине «Программирование высокопроизводительных систем».
3. Методические рекомендации по выполнению письменных работ ( ) (при наличии).
4. Методические рекомендации по выполнению контрольной работы студентами заочной формы обучения (при наличии)
5. Методические указания по выполнению курсовой работы (проекта) (при наличии).

Для успешного освоения дисциплины, необходимо самостоятельно детально изучить представленные темы по рекомендуемым источникам информации:

№ п/п	Темы для самостоятельного изучения	Рекомендуемые источники информации (№ источника)		
		основная (из п.8 РПД)	дополнительная (из п.8 РПД)	метод. лит. (из п.8 РПД)
1	Базовые конструкции Python. Условный оператор Циклы Вложенные циклы	Л1.1, Л1.2	Л2.1, Л2.2	Л3.1
2	Коллекции и работа с памятью. Встроенные возможности по работе с коллекциями. Списочные выражения. Модель памяти для типов языка Python	Л1.1, Л1.2	Л2.1, Л2.2	Л3.1
3	Функции и их особенности в Python. Лямбда-функции Рекурсия. Декораторы. Генераторы	Л1.1, Л1.2	Л2.1, Л2.2	Л3.1
4	Объектно-ориентированное программирование. Лямбда-функции Рекурсия. Декораторы. Генераторы	Л1.1, Л1.2	Л2.1, Л2.2	Л3.1
5	Библиотеки для получения и обработки данных. Библиотеки для получения и обработки данных	Л1.1, Л1.2	Л2.1, Л2.2	Л3.1

## 7. Фонд оценочных средств (оценочных материалов) для проведения промежуточной аттестации обучающихся по дисциплине «Программирование высокопроизводительных систем»

### 7.1. Перечень индикаторов компетенций с указанием этапов их формирования в процессе освоения образовательной программы

Индикатор компетенции (код и содержание)	Дисциплины/элементы программы (практики, ГИА), участвующие в формировании индикатора компетенции	

### 7.2. Критерии и шкалы оценивания уровня усвоения индикатора компетенций, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Оценка знаний, умений и навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций по дисциплине «Программирование высокопроизводительных систем» проводится в форме текущего контроля и промежуточной аттестации.

Текущий контроль проводится в течение семестра с целью определения уровня усвоения

обучающимися знаний, формирования умений и навыков, своевременного выявления преподавателем недостатков в подготовке обучающихся и принятия необходимых мер по её корректировке, а также для совершенствования методики обучения, организации учебной работы и оказания индивидуальной помощи обучающемуся.

Промежуточная аттестация по дисциплине «Программирование высокопроизводительных систем» проводится в виде Экзамен.

За знания, умения и навыки, приобретенные студентами в период их обучения, выставляются оценки «ЗАЧТЕНО», «НЕ ЗАЧТЕНО». (или «ОТЛИЧНО», «ХОРОШО», «УДОВЛЕТВОРИТЕЛЬНО», «НЕУДОВЛЕТВОРИТЕЛЬНО» для дифференцированного зачета/экзамена)

Для оценивания знаний, умений, навыков и (или) опыта деятельности в университете применяется балльно-рейтинговая система оценки качества освоения образовательной программы. Оценка проводится при проведении текущего контроля успеваемости и промежуточных аттестаций обучающихся. Рейтинговая оценка знаний является интегрированным показателем качества теоретических и практических знаний и навыков студентов по дисциплине.

### Состав балльно-рейтинговой оценки студентов очной формы обучения

Для студентов очной формы обучения знания по осваиваемым компетенциям формируются на лекционных и практических занятиях, а также в процессе самостоятельной подготовки.

В соответствии с балльно-рейтинговой системой оценки, принятой в Университете студентам начисляются баллы по следующим видам работ:

№ контрольной точки	Оценочное средство результатов индикаторов достижения компетенций		Максимальное количество баллов
2 семестр			
КТ 1	Тест		0
КТ 2	Тест		0
<b>Сумма баллов по итогам текущего контроля</b>			<b>0</b>
Посещение лекционных занятий			20
Посещение практических/лабораторных занятий			20
Результативность работы на практических/лабораторных занятиях			30
Итого			70
№ контрольной точки	Оценочное средство результатов индикаторов достижений компетенций	Максимальное количество баллов	Критерии оценки знаний студентов
2 семестр			
КТ 1	Тест	0	
КТ 2	Тест	0	

### Критерии и шкалы оценивания результатов обучения на промежуточной аттестации

При проведении итоговой аттестации «зачет» («дифференцированный зачет», «экзамен») преподавателю с согласия студента разрешается выставлять оценки («отлично», «хорошо», «удовлетворительно», «зачет») по результатам набранных баллов в ходе текущего контроля успеваемости в семестре по выше приведенной шкале.

В случае отказа – студент сдает зачет (дифференцированный зачет, экзамен) по приведенным выше вопросам и заданиям. Итоговая успеваемость (зачет, дифференцированный зачет, экзамен) не может оцениваться ниже суммы баллов, которую студент набрал по итогам текущей и промежуточной успеваемости.

При сдаче (зачета, дифференцированного зачета, экзамена) к заработанным в течение семестра студентом баллам прибавляются баллы, полученные на (зачете, дифференцированном зачете, экзамене) и сумма баллов переводится в оценку.

### Критерии и шкалы оценивания ответа на экзамене

Сдача экзамена может добавить к текущей балльно-рейтинговой оценке студентов не более 20 баллов:

Содержание билета	Количество баллов
Теоретический вопрос №1	до 7
Теоретический вопрос №2	до 7
Задача (оценка умений и	до 6
Итого	20

## Критерии оценки ответа на экзамене

Теоретические вопросы (вопрос 1, вопрос 2)

7 баллов выставляется студенту, полностью освоившему материал дисциплины или курса в соответствии с учебной программой, включая вопросы рассматриваемые в рекомендованной программой дополнительной справочно-нормативной и научно-технической литературы, свободно владеющему основными понятиями дисциплины. Требуется полное понимание и четкость изложения ответов по экзаменационному заданию (билету) и дополнительным вопросам, заданных экзаменатором. Дополнительные вопросы, как правило, должны относиться к материалу дисциплины или курса, не отраженному в основном экзаменационном задании (билете) и выявляют полноту знаний студента по дисциплине.

5 балла заслуживает студент, ответивший полностью и без ошибок на вопросы экзаменационного задания и показавший знания основных понятий дисциплины в соответствии с обязательной программой курса и рекомендованной основной литературой.

3 балла дан недостаточно полный и недостаточно развернутый ответ. Логика и последовательность изложения имеют нарушения. Допущены ошибки в раскрытии понятий, употреблении терминов. Студент не способен самостоятельно выделить существенные и несущественные признаки и причинно-следственные связи. Студент может конкретизировать обобщенные знания, доказав на примерах их основные положения только с помощью преподавателя. Речевое оформление требует поправок, коррекции.

2 балла дан неполный ответ, представляющий собой разрозненные знания по теме вопроса с существенными ошибками в определениях. Присутствуют фрагментарность, нелогичность изложения. Студент не осознает связь данного понятия, теории, явления с другими объектами дисциплины. Отсутствуют выводы, конкретизация и доказательность изложения. Речь неграмотная. Дополнительные и уточняющие вопросы преподавателя не приводят к коррекции ответа студента не только на поставленный вопрос, но и на другие вопросы дисциплины.

1 балл дан неполный ответ, представляющий собой разрозненные знания по теме вопроса с существенными ошибками в определениях. Присутствуют фрагментарность, нелогичность изложения. Студент не осознает связь данного понятия, теории, явления с другими объектами дисциплины. Отсутствуют выводы, конкретизация и доказательность изложения. Речь неграмотная. Дополнительные и уточняющие вопросы преподавателя не приводят к коррекции ответа студента не только на поставленный вопрос, но и на другие вопросы дисциплины.

0 баллов - при полном отсутствии ответа, имеющего отношение к вопросу.

Оценивание задачи

6 баллов Задачи решены в полном объеме с соблюдением необходимой последовательности.

5 баллов Задачи решены с небольшими недочетами.

4 балла Задачи решены с небольшими недочетами.

3 балла Задачи решены не полностью, но объем выполненной части таков, что позволяет получить правильные результаты и выводы.

2 балла Задачи решены не полностью, но объем выполненной части таков, что позволяет получить правильные результаты и выводы.

1 баллов Задачи решены частично, с большим количеством вычислительных ошибок, объем выполненной части работы не позволяет сделать правильных выводов.

0 баллов Задачи не решены или работа выполнена не полностью, и объем выполненной части работы не позволяет сделать правильных выводов.

Перевод рейтинговых баллов в пятибалльную систему оценки знаний обучающихся:  
для экзамена:

- «отлично» – от 89 до 100 баллов – теоретическое содержание курса освоено полностью, без пробелов необходимые практические навыки работы с освоенным материалом сформированы, все

предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному;

- «хорошо» – от 77 до 88 баллов – теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками;

- «удовлетворительно» – от 65 до 76 баллов – теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки;

- «неудовлетворительно» – от 0 до 64 баллов - теоретическое содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки, дополнительная самостоятельная работа над материалом курса не приведет к существенному повышению качества выполнения учебных заданий

### **7.3. Примерные оценочные материалы для текущего контроля успеваемости и промежуточной аттестации по итогам освоения дисциплины «Программирование высокопроизводительных систем»**

1. Язык программирования Python: описание и философия.
2. Исходный код на языке Python: кодировка, физические и логические строки, блоки кода
3. Выражения в языке Python.
4. Идентификаторы, пространства имен и области видимости
5. Управляющие конструкции: операторы выбора и цикла.
6. Обработка исключений
7. Функции в языке Python. Лямбда-выражения.
8. Встроенные типы: целочисленный, вещественный, комплексный, логический
9. Последовательности. Кортежи.
10. Последовательности. Списки. Срезы.
11. Последовательности. Словари.
12. Множества и операции над ними
13. Файлы и операции над ними
14. Стиль программирования: описание и назначение.
15. Модули и пакеты
16. Обзор стандартной библиотеки. Модуль sys
17. Обзор стандартной библиотеки. Модуль copy
18. Обзор стандартной библиотеки. Модуль os
19. Обзор стандартной библиотеки. Модуль math
20. Обзор стандартной библиотеки. Модуль random
21. Функции преобразования типов
22. Функции ввода-вывода
23. Функциональное программирование: определение и основные элементы
24. Виды параметров функций в Python'e. Параметры по умолчанию
25. Функции как параметры и как результат
26. Декораторы
27. Функции для обработки последовательностей: range, xrange, map, filter
28. Функции для обработки последовательностей: sum, reduce zip
29. Списковые включения
30. Генераторы
31. Генераторные выражения
32. Итераторы. Функции iter, enumerate, sorted.
33. Модуль itertools. Функции itertools.chain, itertools.repeat, itertools.count.
34. Объектно-ориентированное программирование в Python. Объявление класса
35. Объектно-ориентированное программирование в Python: атрибуты, свойства, сокрытие данных

Список тем для рефератов по дисциплине "Основы программирования", разделенный по категориям сложности и направленности. Темы варьируются от базовых до более сложных, что позволяет выбрать вариант в зависимости от уровня подготовки.

#### Категория 1: Базовые концепции и синтаксис

Эти темы идеально подходят для демонстрации понимания фундаментальных принципов Python.

Сравнительный анализ типов данных в Python: изменяемые и неизменяемые объекты.

Раскрыть тему: Рассмотреть числа, строки, списки, кортежи, словари и множества. Объяснить разницу между ними, последствия их использования (например, при передаче в функции) и практические примеры применения каждого.

Управляющие конструкции в Python: условные операторы и циклы.

Раскрыть тему: Подробно разобрать `if`, `elif`, `else`, циклы `for` и `while`. Показать их синтаксис, особенности (например, `else` у циклов) и привести примеры решения типовых задач (поиск элемента, подсчет суммы и т.д.).

Функции в Python: от простых до рекурсивных.

Раскрыть тему: Объяснить объявление и вызов функций, параметры (позиционные, именованные, по умолчанию), возвращаемые значения. Отдельно осветить тему рекурсии, ее плюсы и минусы, сравнить с итеративным подходом на примере (например, вычисление факториала, чисел Фибоначчи).

Работа с исключениями в Python: конструкция `try-except-else-finally`.

Раскрыть тему: Объяснить философию обработки ошибок в Python. Разобрать иерархию встроенных исключений, показать, как создавать собственные классы исключений, и как правильно использовать блоки `try-except` для написания устойчивого к ошибкам кода.

#### Категория 2: Углубленное изучение возможностей Python

Темы для тех, кто хочет показать понимание более сложных аспектов языка.

Объектно-ориентированное программирование (ООП) в Python: классы, наследование, инкапсуляция, полиморфизм.

Раскрыть тему: Классическая тема для реферата. Объяснить четыре основных концепции ООП на практических примерах. Показать, как они реализуются в Python (магические методы, доступ к атрибутам, переопределение методов).

Списковые включения (List Comprehensions), генераторы и итераторы в Python.

Раскрыть тему: Показать элегантность и эффективность списковых включений по сравнению с традиционными циклами. Объяснить концепцию генераторов (ключевое слово `yield`) и их преимущество в потреблении памяти при работе с большими данными.

Работа с файлами в Python: текстовые и бинарные режимы, менеджер контекста (`with`).

Раскрыть тему: Описать основные режимы открытия файлов (`r`, `w`, `a`, `b`). Подробно остановиться на использовании оператора `with` для гарантированного закрытия файла и корректного управления ресурсами.

Модули и пакеты в Python: создание, импорт и структура проекта.

Раскрыть тему: Объяснить, как организовать код в модули и пакеты для его повторного использования и поддержки. Разобрать способы импорта, переменную `__name__` и структуру типичного Python-проекта (например, с использованием `setup.py`/`pyproject.toml`).

#### Категория 3: Прикладное применение и популярные библиотеки

Темы, демонстрирующие, как Python используется для решения реальных задач.

Введение в научные вычисления с библиотеками NumPy и SciPy.

Раскрыть тему: Рассказать о назначении этих библиотек, их основных объектах (многомерные массивы `ndarray` в NumPy) и преимуществах перед стандартными типами данных Python для математических операций.

Библиотека Pandas для анализа данных: `Series` и `DataFrame`.

Раскрыть тему: Описать два ключевых структуры данных в Pandas. Показать на примере, как можно загружать данные из CSV-файла, производить их фильтрацию, агрегацию и очистку.

Создание графиков и визуализация данных с помощью библиотек Matplotlib и Seaborn.

Раскрыть тему: Сравнить эти две библиотеки. Показать, как строить основные типы графиков (линейные, столбчатые, гистограммы, `scatter plot`) и настраивать их внешний вид для эффективного представления информации.

Веб-скрапинг на Python: библиотеки Requests и BeautifulSoup.

Раскрыть тему: Объяснить основы парсинга веб-страниц: получение HTML-кода с помощью requests и его последующий разбор с помощью BeautifulSoup для извлечения нужной информации. Важно затронуть этические и юридические аспекты скрапинга.

Категория 4: Обзорные и концептуальные темы

Темы для рефератов, требующие анализа и обобщения информации.

Сравнительный анализ Python с другими языками программирования (например, C++, Java, JavaScript).

Раскрыть тему: Сравнить синтаксис, типизацию (статическая vs динамическая), производительность, области применения и философию языков. Выделить сильные и слабые стороны Python.

Области применения Python: от веб-разработки до искусственного интеллекта.

Раскрыть тему: Дать обзор основных сфер, где используется Python: веб-разработка (Django, Flask), Data Science (Pandas, Scikit-learn), машинное обучение (TensorFlow, PyTorch), автоматизация задач и научные исследования.

Инструменты разработки для Python: среды разработки (PyCharm, VS Code), управление зависимостями (pip, venv).

Раскрыть тему: Рассказать о том, как правильно организовать рабочее окружение: использование виртуальных сред (venv) для изоляции проектов, менеджер пакетов pip, и обзор популярных IDE и их преимуществ.

Знания

Задание 1

Что такое переменная в программировании?

- а) Постоянное значение, которое нельзя изменить
- б) Именованная область памяти для хранения данных
- в) Математическая переменная в уравнениях
- г) Тип данных для чисел с плавающей точкой

Ответ: б

Задание 2

Какие из перечисленных типов данных являются целочисленными? (выберите несколько вариантов)

- а) int
- б) float
- в) char
- г) double

Ответ: а, в

Задание 3

Что такое алгоритм?

- а) Язык программирования
- б) Среда разработки
- в) Конечная последовательность шагов для решения задачи
- г) Математическая формула

Ответ: в

Задание 4

Какие из этих операторов относятся к операторам сравнения? (выберите несколько вариантов)

- а) =
- б) ==
- в) !=
- г) +=

Ответ: б, в

Задание 5

Что выведет этот код?

- а) 2
- б) 2.5

- в) 3
- г) Ошибку

Ответ: б

Задание 6

Какой тип цикла гарантирует одно выполнение тела цикла?

- а) for
- б) while
- в) do-while
- г) foreach

Ответ: в

Задание 7

Что такое массив?

- а) Переменная для хранения одного значения
- б) Структура данных для хранения набора элементов одного типа
- в) Функция для сортировки данных
- г) Математическая матрица

Ответ: б

Задание 8

Какие из этих понятий относятся к ООП? (выберите несколько вариантов)

- а) Инкапсуляция
- б) Компиляция
- в) Наследование
- г) Полиморфизм

Ответ: а, в, г

Задание 9

Что такое функция в программировании?

- а) Математическая функция
- б) Именованный блок кода, который можно вызывать
- в) Тип данных
- г) Оператор цикла

Ответ: б

Задание 10

Какой оператор используется для выполнения действия при истинном условии?

- а) if
- б) for
- в) while
- г) switch

Ответ: а

Задание 11

Что такое компилятор?

- а) Программа для написания кода
- б) Программа, переводящая код в машинные команды
- в) Библиотека функций
- г) Отладчик программ

Ответ: б

Задание 12

Какие из этих структур данных являются линейными? (выберите несколько вариантов)

- а) Массив
- б) Стек
- в) Дерево
- г) Очередь

Ответ: а, б, г

Задание 13

Что такое рекурсия?

- а) Бесконечный цикл
- б) Вызов функцией самой себя

в) Ошибка компиляции

г) Тип данных

Ответ: б

Задание 14

Какой принцип ООП описывает сокрытие внутренней реализации?

а) Наследование

б) Инкапсуляция

в) Полиморфизм

г) Абстракция

Ответ: б

Задание 15

Что такое IDE?

а) Язык программирования

б) Интегрированная среда разработки

в) База данных

г) Операционная система

Ответ: б

Задание 16

Какие из этих типов данных являются вещественными? (выберите несколько вариантов)

а) int

б) float

в) double

г) char

Ответ: б, в

Задание 17

Что такое отладка (debugging)?

а) Написание кода

б) Процесс поиска и исправления ошибок

в) Компиляция программы

г) Тестирование программы

Ответ: б

Задание 18

Какой оператор цикла используется, когда известно количество итераций?

а) while

б) do-while

в) for

г) if

Ответ: в

Задание 19

Что такое логический тип данных?

а) Может хранить только числа

б) Может хранить true/false значения

в) Может хранить текст

г) Может хранить дробные числа

Ответ: б

Задание 20

Какие из этих операторов относятся к арифметическим? (выберите несколько вариантов)

а) +

б) &&

в) \*

г) ||

Ответ: а, в

Умения

Задание 1

Установите соответствие между типами данных и их описаниями:

Тип данных      Описание

1. int А. Символьный тип
2. float Б. Целочисленный тип
3. char В. Логический тип
4. bool Г. Число с плавающей точкой

Правильный ответ: 1-Б, 2-Г, 3-А, 4-В

Задание 2

Установите соответствие между операторами и их назначением:

- | Оператор | Назначение |
|----------|------------|
|----------|------------|

Правильный ответ: 1-Б, 2-А, 3-Г, 4-В

Задание 3

Установите соответствие между видами циклов и их характеристиками:

- | Цикл | Характеристика |
|------|----------------|
|------|----------------|

Правильный ответ: 1-Б, 2-В, 3-А

Задание 4

Установите соответствие между концепциями ООП и их определениями:

- | Концепция | Определение |
|-----------|-------------|
|-----------|-------------|

Правильный ответ: 1-Б, 2-А, 3-В, 4-Г

Задание 5

Установите соответствие между структурами данных и их описаниями:

- | Структура | Описание |
|-----------|----------|
|-----------|----------|

Правильный ответ: 1-Б, 2-В, 3-Г, 4-А

Задание 6

Установите соответствие между типами ошибок и их примерами:

- | Тип ошибки | Пример |
|------------|--------|
|------------|--------|

Правильный ответ: 1-Б, 2-В, 3-А

Задание 7

Установите соответствие между арифметическими операторами и их результатом:

- | Выражение | Результат |
|-----------|-----------|
|-----------|-----------|

Правильный ответ: 1-А, 2-Б, 3-В, 4-Г

Задание 8

Установите соответствие между понятиями и их описаниями:

- | Понятие | Описание |
|---------|----------|
|---------|----------|

3. Функция В. Блок кода с именем  
4. Массив Г. Коллекция элементов одного типа

Правильный ответ: 1-Б, 2-А, 3-В, 4-Г

Задание 9

Установите соответствие между логическими операторами и их результатом:

Выражение Результат (true/false)

1. true && false А. true

2. true || false Б. false

3. !true В. false

4. !false Г. true

Правильный ответ: 1-Б, 2-А, 3-В, 4-Г

Задание 10

Установите соответствие между этапами разработки и их содержанием:

Этап Содержание

1. Анализ А. Написание кода

2. Проектирование Б. Определение требований

3. Реализация В. Создание архитектуры

4. Тестирование Г. Поиск ошибок

Правильный ответ: 1-Б, 2-В, 3-А, 4-Г

Задание 11

Расставьте в правильном порядке приоритет выполнения операторов (от высшего к низшему):

А. +, -

Б. \*, /

В. ()

Г. =

Правильный порядок: В → Б → А → Г

Задание 12

Расставьте в правильном порядке этапы выполнения программы:

А. Компиляция

Б. Написание кода

В. Выполнение

Г. Линковка

Правильный порядок: Б → А → Г → В

Задание 13

Расставьте в правильном порядке жизненный цикл переменной:

А. Использование

Б. Объявление

В. Инициализация

Г. Уничтожение

Правильный порядок: Б → В → А → Г

Задание 14

Расставьте в правильном порядке выполнение операторов в выражении:

А. 3 \* 4

Б. 2 + 12

В. 8 / 2

Г. 4 - 6

Выражение: 8 / 2 + 3 \* 4 - 6

Правильный порядок: В → А → Б → Г

Задание 15

Расставьте в правильном порядке этапы работы цикла for:

А. Выполнение тела цикла

Б. Проверка условия

В. Изменение счетчика

Г. Инициализация счетчика

Правильный порядок: Г → Б → А → В

#### Задание 16

Расставьте в правильном порядке процесс вызова функции:

- А. Возврат значения
- Б. Вызов функции
- В. Выполнение тела функции
- Г. Передача параметров

Правильный порядок: Б → Г → В → А

#### Задание 17

Расставьте в правильном порядке процесс создания массива:

- А. Выделение памяти
- Б. Объявление массива
- В. Инициализация элементов
- Г. Определение размера

Правильный порядок: Б → Г → А → В

#### Задание 18

Расставьте в правильном порядке этапы обработки исключения:

- А. Выполнение блока catch
- Б. Генерация исключения
- В. Нормальное продолжение выполнения
- Г. Поиск обработчика

Правильный порядок: Б → Г → А → В

#### Задание 19

Расставьте в правильном порядке процесс создания объекта в ООП:

- А. Вызов конструктора
- Б. Объявление переменной
- В. Выделение памяти
- Г. Инициализация полей

Правильный порядок: Б → В → А → Г

#### Задание 20

Расставьте в правильном порядке процесс работы со стеком:

- А. Добавление элемента (push)
- Б. Проверка на пустоту
- В. Удаление элемента (pop)
- Г. Чтение верхнего элемента (peek)

Правильный порядок: Б → А → Г → В

#### Навыки

##### Задание 1

Верно ли утверждение: В языке Java все переменные должны быть объявлены с указанием их типа данных.

Ответ: Верно

##### Задание 2

Верно ли утверждение: Оператор '==' и метод equals() в Java всегда работают одинаково для сравнения строк.

Ответ: Неверно

##### Задание 3

Верно ли утверждение: Массив в программировании может содержать элементы разных типов данных.

Ответ: Неверно

##### Задание 4

Верно ли утверждение: Цикл 'while' проверяет условие перед выполнением тела цикла, а цикл 'do-while' - после.

Ответ: Верно

##### Задание 5

Верно ли утверждение: Функция всегда должна возвращать значение.

Ответ: Неверно

##### Задание 6

Верно ли утверждение: В объектно-ориентированном программировании класс является шаблоном для создания объектов.

Ответ: Верно

Задание 7

Верно ли утверждение: Рекурсивная функция обязательно должна иметь базовый случай (условие выхода).

Ответ: Верно

Задание 8

Верно ли утверждение: Переменная, объявленная внутри функции, доступна в любой другой функции программы.

Ответ: Неверно

Задание 9

Верно ли утверждение: В двоичной системе счисления используются только цифры 0 и 1.

Ответ: Верно

Задание 10

Верно ли утверждение: Компилятор и интерпретатор - это одно и то же.

Ответ: Неверно

Задание 11

Как называется ошибка, когда программа пытается обратиться к элементу массива по индексу, который выходит за его границы?

Ответ: Выход за границы массива

Задание 12

Сколько бит занимает один байт?

Ответ: 8

Задание 13

Как называется структура данных, работающая по принципу "последним пришел - первым ушел" (LIFO)?

Ответ: Стек

Задание 14

Какой оператор в Java используется для получения остатка от деления?

Ответ: %

Задание 15

Как называется процесс поиска и исправления ошибок в программе?

Ответ: Отладка

Задание 16

Сколько значений может хранить переменная логического типа (boolean)?

Ответ: 2

Задание 17

Как называется ошибка, когда программа выполняется, но выдает неправильный результат из-за ошибки в алгоритме?

Ответ: Логическая ошибка

Задание 18

Какой тип данных используется для хранения одного символа в Java?

Ответ: char

Задание 19

Как называется специальный метод класса, который вызывается при создании нового объекта?

Ответ: Конструктор

Задание 20

Какой оператор цикла в Java гарантирует одно выполнение тела цикла?

Ответ: do-while

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

а) Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

### основная

Л1.1 Бабушкина И. А., Окулов С. М. Практикум по объектно-ориентированному программированию [Электронный ресурс]:. - Москва: Лаборатория знаний, 2020. - 369 с. – Режим доступа: <https://e.lanbook.com/book/135561>

Л1.2 Дадян Э. Г. Проектирование современных баз данных [Электронный ресурс]:учеб.-метод. пособие; ВО - Специалитет. - Москва: ООО "Научно-издательский центр ИНФРА-М", 2017. - 120 с. – Режим доступа: <http://new.znaniium.com/go.php?id=959294>

### дополнительная

Л2.1 Сакулин С. А. Основы интернет-технологий: HTML, CSS, JavaScript, XML [Электронный ресурс]:учеб. пособие; ВО - Бакалавриат. - Москва: МГТУ им. Н.Э. Баумана, 2017. - 112 с. – Режим доступа: <https://e.lanbook.com/book/103525>

Л2.2 Гуриков С. Р. Интернет-технологии [Электронный ресурс]:учеб. пособие ; ВО - Бакалавриат. - Москва: ООО "Научно-издательский центр ИНФРА-М", 2022. - 174 с. – Режим доступа: <http://znaniium.com/catalog/document?id=391737>

б) Методические материалы, разработанные преподавателями кафедры по дисциплине, в соответствии с профилем ОП.

Л3.1 Никитина Т. П., Королев Л. В. Программирование. Основы Python для инженеров [Электронный ресурс]:учеб. пособие; ВО - Бакалавриат, Магистратура. - Санкт-Петербург: Лань, 2023. - 156 с. – Режим доступа: <https://e.lanbook.com/book/302720>

## 9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

№	Наименование ресурса сети «Интернет»	Электронный адрес ресурса
1	Python : официальный сайт. – URL: <a href="https://www.python.org/">https://www.python.org/</a> (дата обращения: 29.08.2023).	<a href="https://www.python.org/">https://www.python.org/</a>
2	NumPy : официальный сайт. – URL: <a href="https://numpy.org/">https://numpy.org/</a> (дата обращения: 29.08.2023).	<a href="https://numpy.org/">https://numpy.org/</a>

## 10. Методические указания для обучающихся по освоению дисциплины

### Лекционные занятия

Основа освоения дисциплины – лекция, целью которой является целостное и логичное рассмотрение основного материала курса. Вместе с тем значимость лекции определяется тем, что она не только способствует выработке логического мышления, но и способствует развитию интереса к пониманию современной действительности.

Задача студентов в процессе умелой и целеустремленной работы на лекциях – внимательно слушать преподавателя, следить за его мыслью, предлагаемой системой логических посылок, доказательств и выводов, фиксировать (записывать) основные идеи, важнейшие характеристики понятий, теорий, наиболее существенные факты. Лекция задает направление, содержание и эффективность других форм учебного процесса, нацеливает студентов на самостоятельную работу и определяет основные ее направления (подготовку к практическим занятиям, выполнение творческих заданий, рефератов, решение контекстных задач).

Записывание лекции – творческий процесс. Запись лекции крайне важна. Это позволяет надолго сохранить основные положения лекции; способствует поддержанию внимания; способствует лучшему запоминанию материала. Важно уметь оформить конспект так, чтобы важные моменты были выделены графически, а главную информацию следует выделять в самостоятельные абзацы, фиксируя ее более крупными буквами или цветными маркерами. Конспект должен иметь поля для заметок. Это могут быть библиографические ссылки и, наконец, собственные комментарии. Для быстрой записи текста можно придумать условные знаки, при этом таких знаков

не должно быть более 10–15. Условные обозначения придумывают для часто встречающихся слов (существует, который, каждый, точка зрения, на основании и т.п.).

Перед каждой лекцией необходимо внимательно прочитать материал предыдущей лекции. В рабочей тетради графически выделить: тему лекции, основные теоретические положения. Подготовленный студент легко следит за мыслью преподавателя, что позволяет быстрее запоминать новые понятия, сущность которых выявляется в контексте лекции. Повторение материала облегчает в дальнейшем подготовку к экзамену. Затем надо ознакомиться с материалом темы по учебнику, внести нужные уточнения и дополнения в лекционный материал. После усвоения каждой темы рекомендуется проверять свои знания, отвечая на контрольные вопросы по теме.

#### Практические занятия

Целью практических занятий является закрепление, расширение, углубление теоретических знаний, полученных на лекциях и в ходе самостоятельной работы, развитие познавательных способностей.

Являясь частью образовательного процесса, практические занятия преследует ряд основополагающих задач:

- работа с источниками, которая идет на уровнях индивидуальной самостоятельной работы и в ходе коллективного обсуждения;

- формирование умений и навыков индивидуальной и коллективной работы, позволяющих эффективно использовать основные методы исследования, грамотно выстраивать его основные технологические этапы (знакомство с темой и имеющейся по ней информацией, определение основной проблемы, первичный анализ, определение подходов и ключевых узлов механизма ее развития, публичное обсуждение, предварительные выводы);

- анализ поставленных проблем, умение обсуждать тему, высказывать свое мнение, отстаивать свою позицию, слушать и оценивать различные точки зрения, конструктивно полемизировать, учиться думать, говорить, слушать, понимать, находить точки соприкосновения разных позиций, их разумного сочетания;

- формирование установок на творчество;

- диалог, внутренний и внешний; поиск и разрешение проблемы в рамках имеющейся о ней информации;

- поиск рационального зерна в самых противоречивых позициях и подходах к проблеме;

- открытость новому и принципиальную возможность изменить свою позицию и вытекающие из нее решения, в случае получения новой информации и связанных с ней обстоятельств сознательный отход от подготовленного к семинару текста во время своего, построенного на тезисном изложении фактов и мыслей, когда конспект привлекается лишь в том случае, когда надо привести какие-то факты.

#### Рекомендации по подготовке к экзамену

Формой итогового контроля знаний студентов по дисциплине является экзамен.

Экзамен, на который явка обязательна, проводится согласно расписанию учебных занятий. Экзамен является формой отчетности, фиксирующей, что студент выполнил необходимый минимум работы по освоению определенного раздела образовательной программы.

Подготовка к экзамену и успешное освоение материала дисциплины начинается с первого дня изучения дисциплины и требует от студента систематической работы:

- 1) не пропускать аудиторные занятия (лекции, практические занятия);

- 2) активно участвовать в работе семинаров (выступать с сообщениями, проявляя себя в выполнении всех видов заданий – устном опросе, творческих заданиях, в решении и обсуждении контекстных задач, в деловой игре, выполнять все требования преподавателя по изучению курса, приходить подготовленными к занятию).

## **11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства и информационных справочных систем (при необходимости).**

### *11.1 Перечень лицензионного программного обеспечения*

1. Kaspersky Endpoint Security 12.11 - Антивирус

2. Microsoft Windows Server STDCORE AllLngLicense/Software AssurancePack Academic OLV 16Licenses LevelE AdditionalProduct CoreLic 1Year - Серверная операционная система

### 11.3 Перечень программного обеспечения отечественного производства

1. Kaspersky Endpoint Security 12.11 - Антивирус

При осуществлении образовательного процесса студентами и преподавателем используются следующие информационно справочные системы: СПС «Консультант плюс», СПС «Гарант».

## 12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

№ п/п	Наименование специальных помещений и помещений для самостоятельной работы	Номер аудитории	Оснащенность специальных помещений и помещений для самостоятельной работы
1	Учебная аудитория для проведения занятий всех типов (в т.ч. лекционного, семинарского, практической подготовки обучающихся), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации	Э-160	Специализированная мебель на 180 посадочных мест, персональный компьютер – 1 шт., проектор Panasonic EX620 X6A – 1 шт., интерактивная доска SMART Board 690 – 1 шт., трибуна для лектора – 1 шт., микрофон – 1 шт., мониторы - 3 шт., плазменная панель - 1 шт., учебно-наглядные пособия в виде презентаций, информационные плакаты, подключение к сети «Интернет», выход в корпоративную сеть университета.
		422/НК	Оснащение: специализированная мебель на 38 посадочных мест, стол преподавателя – 1 шт., Sharp 70" Информационный ЖК-дисплей – 1 шт., АРМ на основе Intel Core i3 , Монитор Philips 23", Клавиатура + мышь - 25шт., магнитно-маркерная доска – 1 шт., учебно-наглядные пособия в виде тематических презентаций, информационные плакаты, подключение к сети «Интернет», доступ в электронную информационно-образовательную среду университета, выход в корпоративную сеть университета.
2	Помещение для самостоятельной работы обучающихся, подтверждающее наличие материально-технического обеспечения, с перечнем основного оборудования		

		213/НК библио тека	<p>Специализированная мебель на 35 посадочных мест, дисплей - 1 шт., принтер ч/б - 2 шт., МФУ ч/б - 2 шт., сканер - 2 шт., открытый доступ к фонду справочной, краеведческой литературы, Wi-Fi оборудование, подключение к сети «Интернет», доступ к российским и международным ресурсам и базам данных, доступ к электронно-библиотечным системам, доступ в электронную информационно-образовательную среду университета. Открытый доступ к фонду справочной и краеведческой литературы.</p>
		214/НК библио тека	<p>Специализированная мебель на 130 посадочных мест, персональные компьютеры, моноблоки – 80 шт., копир А3 - 3, принтер матричный - 2, МФУ ч/б – 7 шт., МФУ цветной – 2 шт., принтер ч/б – 8 шт., принтер цветн. - 2 шт., сканер – 2 шт., сканеры штрих-кода - 5, наушники - 10 шт., Wi-Fi оборудование, подключение к сети «Интернет», доступ к российским и международным ресурсам и базам данных, доступ к электронно-библиотечным системам, доступ в электронную информационно-образовательную среду университета. Открытый доступ к фонду учебной, научной и художественной литературы.</p>

### 13. Особенности реализации дисциплины лиц с ограниченными возможностями здоровья

Обучающимся с ограниченными возможностями здоровья предоставляются специальные учебники и учебные пособия, иная учебная литература, специальные технические средства обучения коллективного и индивидуального пользования, предоставление услуг ассистента (помощника), оказывающего обучающимся необходимую техническую помощь, а также услуги сурдопереводчиков и тифлосурдопереводчиков.

а) для слабовидящих:

- на промежуточной аттестации 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);

- задания для выполнения, а также инструкция о порядке проведения промежуточной аттестации оформляются увеличенным шрифтом;

- задания для выполнения на промежуточной аттестации зачитываются ассистентом;

- письменные задания выполняются на бумаге, надиктовываются ассистенту;

- обеспечивается индивидуальное равномерное освещение не менее 300 люкс;

- студенту для выполнения задания при необходимости предоставляется увеличивающее устройство;

в) для глухих и слабослышащих:

- на промежуточной аттестации 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);

- промежуточная аттестация проводится в письменной форме;

- обеспечивается наличие звукоусиливающей аппаратуры коллективного пользования, при необходимости поступающим предоставляется звукоусиливающая аппаратура индивидуального пользования;

- по желанию студента промежуточная аттестация может проводиться в письменной форме;

д) для лиц с нарушениями опорно-двигательного аппарата (тяжелыми нарушениями двигательных функций верхних конечностей или отсутствием верхних конечностей):

- письменные задания выполняются на компьютере со специализированным программным обеспечением или надиктовываются ассистенту;

- по желанию студента промежуточная аттестация проводится в устной форме.

Рабочая программа дисциплины «Программирование высокопроизводительных систем» составлена на основе Федерального государственного образовательного стандарта высшего образования - магистратура по направлению подготовки 09.04.03 Прикладная информатика (приказ Минобрнауки России от 19.09.2017 г. № 916).

Автор (ы)

\_\_\_\_\_ доц. КИС, кпн Богданова Светлана Викторовна

Рецензенты

\_\_\_\_\_ доцент , к.т.н. Шлаев Дмитрий Валерьевич

\_\_\_\_\_ доцент , к.т.н. Трошков Александр Михайлович

Рабочая программа дисциплины «Программирование высокопроизводительных систем» рассмотрена на заседании Кафедра инжиниринга IT-решений протокол № 8 от 04.03.2025 г. и признана соответствующей требованиям ФГОС ВО и учебного плана по направлению подготовки 09.04.03 Прикладная информатика

Заведующий кафедрой \_\_\_\_\_ Березницкий А.С.

Рабочая программа дисциплины «Программирование высокопроизводительных систем» рассмотрена на заседании учебно-методической комиссии Факультет цифровых технологий протокол № от г. и признана соответствующей требованиям ФГОС ВО и учебного плана по направлению подготовки 09.04.03 Прикладная информатика

Руководитель ОП \_\_\_\_\_