

Федеральное государственное образовательное
бюджетное учреждение высшего образования
**«ФИНАНСОВЫЙ УНИВЕРСИТЕТ
ПРИ ПРАВИТЕЛЬСТВЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»**
(Финансовый университет)

Краснодарский филиал Финуниверситета

Кафедра «Математика и информатика»

СОГЛАСОВАНО

ООО «Портал-Юг»
Генеральный директор



Е.В. Мостовой

«21» февраля 2025 г.

УТВЕРЖДАЮ

Краснодарский филиал
Финуниверситета

Директор



Э.В. Соболев

«21» февраля 2025 г.

Кирий В.А.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
ПРАКТИКУМ ПО ПРОГРАММИРОВАНИЮ**

студентов, обучающихся по направлению подготовки

01.03.02 Прикладная математика и информатика

в соответствии с образовательными стандартами Краснодарского филиала

Финансового университета

(программа подготовки бакалавров)

*Рекомендовано Ученым советом Краснодарского филиала Финуниверситета
(протокол № 24 от 18.02.2025)*

*Одобрено кафедрой «Математика и информатика»
(протокол № 12 от 11.02.2025)*

Краснодар 2025

УДК 004.42
ББК 32.973
A21

Рецензенты: Н.Г. Пьянкова - доцент кафедры «Математика и информатика» Краснодарского филиала Финуниверситета. В.А. Кирий кандидат физико-математических наук, доцент кафедры «Математика и информатика» Краснодарского филиала Финуниверситета.

Кирий В.А. Рабочая программа дисциплины практикум по программированию для обучающихся по направлению 01.03.02 Прикладная математика и информатика, профиль «Анализ данных и принятие решений в экономике и финансах». – Краснодар: Краснодарский филиал Финуниверситета, кафедра «Математика и информатика», 2025 г.

Дисциплина Практикум по программированию относится к модулю общепрофессиональных дисциплин по направлению подготовки 01.03.02-Прикладная математика и информатика.

В рабочей программе дисциплины определены ее цель, требования к результатам освоения дисциплины, содержание программы, тематика аудиторных занятий, формы самостоятельной работы, оценочные средства для текущего контроля и промежуточной аттестации, учебно-методическое и информационное обеспечение.

Рабочая программа дисциплины практикум по программированию

*Формат 60*90/16. Гарнитура Times New Roman*

Усл. п.л. 2,0. Изд. № _от.

Тираж 100 экз.

Заказ № .

Отпечатано в Краснодарском филиале Финуниверситета

Содержание

1.Наименование дисциплины	4
2.Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине	4
3.Место дисциплины в структуре образовательной программы	6
4.Объем дисциплины в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся	6
5.Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий	7
5.1.Содержание дисциплины	7
5.2.Учебно-тематический план	9
5.3.Содержание семинаров, практических занятий.....	11
6.Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине	12
6.1.Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы	12
6.2.Перечень вопросов, заданий, тем для подготовки к текущему контролю	14
7.Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине	19
7.3.Практико-ориентированные задания.....	29
7.4.Тесты	31
8.Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины	34
9.Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины	35
10.Методические материалы, определяющие процедуры оценивания знаний и умений, характеризующих степень сформированности компетенций	35
12.Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине	38

1.Наименование дисциплины

Б1.О.03.02 «Практикум по программированию».

2.Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине

Дисциплина «Практикум по программированию» обеспечивает формирование следующих компетенций: УК-10; ПКН-4.

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции
УК - 10	Способность осуществлять поиск, критически анализировать, обобщать и систематизировать информацию, использовать системный подход для решения поставленных задач	1.Четко описывает состав и структуру требуемых данных и информации, грамотно реализует процессы их сбора, обработки и интерпретации	Знать: состав и структуру требуемых данных и информации, процессы их сбора, обработки и интерпретации Уметь: четко описывать состав и структуру требуемых данных и информации, грамотно реализовывать процессы их сбора, обработки и интерпретации
		2.Обосновывает сущность происходящего, выявляет закономерности, понимает природу вариабельности	Знать: сущность происходящего, закономерности, природу вариабельности Уметь: обосновать сущность происходящего, выявлять закономерности, понимать природу вариабельности
		3.Формулирует признак классификации, выделяет соответствующие ему группы однородных «объектов», идентифицирует общие свойства элементов этих групп, оценивает полноту результатов классификации, показывает прикладное назначение классификационных групп	Знать: признаки классификации, соответствующие ему группы однородных «объектов», общие свойства элементов этих групп, полноту результатов классификации, прикладное назначение классификационных групп Уметь: формулировать признак классификации, выделять соответствующие ему группы однородных «объектов», идентифицировать общие свойства элементов этих групп, оценивать полноту результатов классификации, показывать прикладное назначение классификационных групп

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции
		4.Грамотно, логично, аргументировано формирует собственные суждения и оценки. Отличает факты от мнений, интерпретаций, оценок и т.д. в рассуждениях других участников деятельности	Знать: принципы формирования собственных суждений и оценок, методы отличий фактов от мнений, интерпретаций, оценок и т.д. в рассуждениях других участников деятельности Уметь: грамотно, логично, аргументировано формировать собственные суждения и оценки. Отличать факты от мнений, интерпретаций, оценок и т.д. в рассуждениях других участников деятельности
		5.Аргументированно и логично представляет свою точку зрения посредством и на основе системного описания	Знать: принципы представления своей точки зрения посредством и на основе системного описания Уметь: аргументированно и логично представлять свою точку зрения посредством и на основе системного описания
ПКН - 4	Способен проектировать и реализовывать прикладные программные системы в соответствии с анализом задачи и требований к ним	1.Демонстрирует базовые знания о существующих математических методах и системах программирования	Знать: базовые знания о существующих математических методах и системах программирования Уметь: демонстрировать базовые знания о существующих математических методах и системах программирования
		2.Использует и адаптирует существующие математические методы и системы программирования для решения прикладных задач	Знать: существующие математические методы и системы программирования для решения прикладных задач Уметь: использовать и адаптировать существующие математические методы и системы программирования для решения прикладных задач
		3.Владеет навыками проектирования и разработки компонентов программного обеспечения на основе современных парадигм, технологий и языков программирования	Знать: навыки проектирования и разработки компонентов программного обеспечения на основе современных парадигм, технологий и языков программирования Уметь: владеть навыками проектирования и разработки компонентов программного обеспечения на основе

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции
			современных парадигм, технологий и языков программирования
		4.Применяет методы машинного обучения для решения прикладных задач анализа данных	Знать: методы машинного обучения для решения прикладных задач анализа данных Уметь: применять методы машинного обучения для решения прикладных задач анализа данных

3.Место дисциплины в структуре образовательной программы

Дисциплина «Практикум по программированию» относится к модулю общепрофессиональных дисциплин направления подготовки 01.03.02 Прикладная математика и информатика, профиль «Анализ данных и принятие решений в экономике и финансах».

Дисциплина «Практикум по программированию» служит для закрепления и углубления знаний и навыков в области программирования, получаемых при освоении других дисциплин, изучаемых в рамках направления подготовки бакалавров «Прикладная математика и информатика».

4.Объем дисциплины в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся

Очная форма обучения.

Вид учебной работы по дисциплине	Всего (в з/ед. и часах)	Семестр 3 (в часах)	Семестр 4 (в часах)
Общая трудоемкость дисциплины	8 зач.ед. / 288 час.	144	144
Контактная работа - Аудиторные занятия	64	32	32
Лекции	-	-	-
Семинары, практические занятия	64	32	32
Самостоятельная работа	224	112	112
Вид промежуточной аттестации	Зачеты	Зачет	Зачет
Вид текущего контроля	Контрольные работы	Контрольная работа	Контрольная работа

Очно – заочная форма обучения.

Вид учебной работы по дисциплине	Всего (в з/ед. и часах)	Семестр 3 (в часах)	Семестр 4 (в часах)
Общая трудоемкость дисциплины	8 зач.ед. / 288 час.	144	144
Контактная работа - Аудиторные занятия	32	16	16
Лекции	-	-	-
Семинары, практические занятия	32	16	16
Самостоятельная работа	256	128	128
Вид промежуточной аттестации	Зачеты	Зачет	Зачет
Вид текущего контроля	Контрольные работы	Контрольная работа	Контрольная работа

5.Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий

5.1.Содержание дисциплины

Тема 1. Основы языка Python

Обработка числовой информации. Встроенные функции и модули для работы с числами. Реализация числовых алгоритмов с использованием инструкции ветвления и циклов.

Обработка текстовой информации. Функции и методы для работы со строками. Регулярные выражения.

Списки. Использование списков для хранения информации. Методы списков. Многомерные списки. Кортежи.

Словари. Типовые случаи использования словарей в программах. Методы для работы со словарями. Множества.

Тема 2. Функции и модули

Функции в Python: общая семантика. Создание функции и ее вызов. Расположение определений функций. Анонимные функции в Python. Необязательные параметры функций и сопоставление по ключам. Возвращение нескольких значений из функции. Распаковка и запаковка параметров функции. Аннотации и документирование функций. Глобальные и локальные переменные.

Глобальные и локальные переменные. Вложенные функции. Функции высшего порядка.

Создание и использование модулей и пакетов. Модули стандартной

библиотеки.

Тема 3. Обработка исключений. Работа с файлами средствами языка Python

Понятие исключения. Инструкция `try ... except ... else ... finally`. Классы встроенных исключений. Инструкции `raise` и `assert`. Инструкция `with ... as`.

Работа с файлами в Python. Концепция файла в современных ОС и языках программирования. Операции с файлами: открытие/закрытие файла, чтение и записи и другие методы для работы с файлами. Инструкция `with ... as` и ее использование для файлов.

Использование текстовых файлов в программе. Двоичные файлы. Сохранение объектов в файл. Работа с файлами различных форматов: `csv`, `xlsx`.

Тема 4. Объектно-ориентированное программирование на Python

Python как объектно-ориентированный язык программирования. Базовые возможности ООП в Python: создание классов и объектов; наследование и полиморфизм; функция `super()`; проверка принадлежности к классу. Базовые типы в Python.

Методы классов и статические переменные и методы в Python. Управление доступом к атрибутам класса в Python. Динамические операции с атрибутами и интроспекция в Python. Использование специальных методов для расширенного функционала пользовательских классов. Кейс построения иерархии классов.

Тема 5. Функциональное программирование на Python

Элементы функционального программирования в Python: функции «граждане первого класса», функции высшего порядка, замыкания, функции без побочных эффектов, рекурсия. Неизменяемые структуры данных. Идиомы, распространенные в функциональных языках программирования: итераторы, последовательности, ленивые вычисления.

Декораторы в Python: использование и создание собственных декораторов.

Подход: `map`, `filter`, `reduce`. Реализация функций `map`, `filter`, `reduce` в Python.

Итераторы в Python, итерируемый тип данных. Модуль `itertools`. Функции-генераторы и выражения-генераторы в Python.

Тема 6. Алгоритмы и структуры данных

Динамические массивы. Стеки, очереди, деки. Связные списки. Реализация связных списков на языке Python. Бинарные деревья. Использование бинарных деревьев в прикладных задачах. Двоичное дерево поиска.

Асимптотическая оценка сложности алгоритма. Алгоритмы сортировки и поиска. Бинарный поиск. Простые методы сортировки: обменные сортировки, сортировка выбором, сортировка вставками. Сортировка Шелла. Быстрая сортировка. Сортировка слиянием.

Тема 7. Паттерны проектирования

Основные принципы объектно-ориентированного проектирования

приложений. Принцип абстракции. Уменьшение зависимости. Основные паттерны проектирования: интерфейс, делегирование. Порождающие шаблоны: фабричный метод, абстрактная фабрика, строитель. Структурные шаблоны: адаптер, мост, декоратор, фасад. Поведенческие шаблоны: цепочка обязанностей, команда, посредник, наблюдатель, состояние, стратегия, посетитель, шаблонный метод.

Тема 8. Программирование графических интерфейсов

Событийно-ориентированное программирование. События и обработчики событий. События мыши и клавиатуры. Основные библиотеки графических интерфейсов в Python: tkinter, PyQt, PyGTK. Главный цикл программы. Основные элементы управления: кнопки, ползунки, поля ввода. Работа с графикой.

Тема 9. Системное программирование на Python

Чтение и запись файлов. Работа с путями в Linux. Обход папок. Работа с архивами. Работа с офисными форматами. Работа со структурированными данными. Основные форматы хранения данных: CSV, XML, JSON.

Тема 10. Сетевое программирование на Python

Получение и разбор HTML-страниц. Библиотеки HTML-парсинга. Сокеты. Клиент-серверные приложения. Обращение к внешним API. Многопоточность. Библиотеки многопоточности и многопроцессности. Отправка и получение электронных писем.

Тема 11. Тестирование программ на Python

Основные виды тестирования программного обеспечения. Модульное и интеграционное тестирование. Понятие регрессии. Фиксирование и формализация требований. Библиотеки автоматизированного тестирования. Написание автоматических модульных тестов по техническому заданию.

Разработка через тестирование. Проектирование через тестирование. Методология TDD.

Тема 12. Документирование и развертывание программ на Python

Основные приемы документирования программного кода. Соглашения о стиле документирования кода. Написание программной документации в формате docstring. Языки форматирования документации: ReST, Markdown. Автоматическая сборка документации с использованием Sphinx. Экспорт документации в популярные форматы.

5.2. Учебно-тематический план

Очная форма обучения.

№ п/п	Наименование тем (разделов)	Трудоемкость в часах		Формы текущего
		Аудиторная работа	Самостоят	

	дисциплины	Всего	Общая, в т.ч.:	Лекции	Семинары, практические занятия	ельная работа	контроля успеваемости
1.	Основы языка Python	24	4	-	4	20	Опрос, проверка выполненных заданий
2.	Функции и модули	24	6	-	6	18	
3.	Обработка исключений. Работа с файлами средствами языка Python	24	4	-	4	20	
4.	Объектно-ориентированное программирование на Python	24	6	-	6	18	
5.	Функциональное программирование на Python	24	4	-	4	20	
6.	Алгоритмы и структуры данных	24	6	-	6	18	
7.	Паттерны проектирования	24	6	-	6	18	
8.	Программирование графических интерфейсов	24	4	-	4	20	
9.	Системное программирование на Python	24	6	-	6	18	
9.	Системное программирование на Python	24	6	-	6	18	
10.	Сетевое программирование на Python	24	6	-	6	18	
11.	Тестирование программ на Python	24	6	-	6	17	
12.	Документирование и развертывание программ на Python	24	6	-	6	1	
В целом по дисциплине		288	64	-	64	224	

5.3.Содержание семинаров, практических занятий

Наименование тем (разделов) дисциплины	Перечень вопросов для обсуждения на семинарских, практических занятиях, рекомендуемые источники из разделов 8,9 (указывается раздел и порядковый номер источника)	Формы проведения занятий
1. Основы языка Python	Решение задач с использованием инструкции ветвления. Решение задач с использованием циклов. Создание и обработка списков. Многомерные списки. Обработка текстовой информации. Использование словарей и множеств. Совместное использование различных типов данных. Источники п.8,9	Интерактивная форма, работа на компьютере
2. Функции и модули	Создание и использование функций. Анонимные функции. Функции высшего порядка. Источники п.8,9	Интерактивная форма, работа на компьютере
3. Обработка исключений. Работа с файлами средствами языка Python	Работа с текстовыми файлами. Работа с двоичными файлами. Сохранение объектов в файл. Источники п.8,9	Интерактивная форма, работа на компьютере
4. Объектно- ориентированное программирование на Python	Создание классов и объектов. Наследование и полиморфизм. Специальные методы классов. Источники п.8,9	Интерактивная форма, работа на компьютере
5. Функциональное программирование на Python	Функции map, filter, reduce, any, all. Декораторы. Итераторы. Функции генераторы. Источники п.8,9	Интерактивная форма, работа на компьютере
6. Алгоритмы и структуры данных	Создание связанных списков. Использование стеков и очередей при решении задач. Алгоритмы сортировки и поиска. Источники п.8,9	Интерактивная форма, работа на компьютере
7. Паттерны проектирования	Создание примеров программных реализаций для распространенных шаблонов проектирования. Источники п.8,9	Интерактивная форма, работа на компьютере
8. Программирование графических интерфейсов	Создание графического приложения с использованием встроенных возможностей языка и библиотек Источники п.8,9	Интерактивная форма, работа на компьютере
9. Системное программирование на Python	Автоматизация рутинных административных задач. Создание Web-парсера.	Интерактивная форма, работа на компьютере

Наименование тем (разделов) дисциплины	Перечень вопросов для обсуждения на семинарских, практических занятиях, рекомендуемые источники из разделов 8,9 (указывается раздел и порядковый номер источника)	Формы проведения занятий
	Источники п.8,9	
Наименование тем (разделов) дисциплины	Перечень вопросов для обсуждения на семинарских, практических занятиях, рекомендуемые источники из разделов 8,9 (указывается раздел и порядковый номер источника)	Формы проведения занятий
10. Сетевое программирование на Python	Создание многопоточного многопользовательского многофункционального сервера на сокетах. Источники п.8,9	Интерактивная форма, работа на компьютере
11. Тестирование программ на Python	Написание модульных тестов по описанию. Разработка программы по тестам. Источники п.8,9	Интерактивная форма, работа на компьютере
12. Документирование и развертывание программ на Python	Документирование программы по описанию и исходному коду. Экспорт документации в формате сайта. Источники п.8,9	Интерактивная форма, работа на компьютере

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

6.1.Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы

Наименование тем (разделов) дисциплины	Перечень вопросов, отводимых на самостоятельное освоение	Формы внеаудиторной самостоятельной работы
1. Основы языка Python	Функции модуля math, random, copy. Регулярные выражения. Кортежи. Множества.	Работа с учебной литературой и документацией. Решение задач с использованием Jupyter Notebook. Выполнение домашних заданий.
2. Функции и модули	Создание и использование модулей и пакетов.	Работа с учебной литературой и документацией. Решение задач с использованием Jupyter Notebook. Выполнение домашних заданий.

Наименование тем (разделов) дисциплины	Перечень вопросов, отводимых на самостоятельное освоение	Формы внеаудиторной самостоятельной работы
3. Обработка исключений. Работа с файлами средствами языка Python	Работа с файлами различных форматов: csv, docx, xlsx.	Работа с учебной литературой и документацией. Решение задач с использованием Jupyter Notebook. Выполнение домашних заданий.
4. Объектно-ориентированное программирование на Python	Специальные методы классов.	Работа с учебной литературой и документацией. Решение задач с использованием Jupyter Notebook. Выполнение домашних заданий.
5. Функциональное программирование на Python	Функции any, all, zip.	Работа с учебной литературой и документацией. Решение задач с использованием Jupyter Notebook. Выполнение домашних заданий.
6. Алгоритмы и структуры данных	Бинарные деревья. Использование бинарных деревьев в прикладных задачах. Двоичное дерево поиска.	Работа с учебной литературой и документацией. Решение задач с использованием Jupyter Notebook. Выполнение домашних заданий.
7. Паттерны проектирования	Внедрение зависимости, инверсия управления.	Работа с учебной литературой и документацией. Решение задач с использованием Jupyter Notebook. Выполнение домашних заданий.
8. Программирование графических интерфейсов	Создание игр. Библиотека PyGame.	Работа с учебной литературой и документацией. Решение задач с использованием Jupyter Notebook. Выполнение домашних заданий.
9. Системное программирование на Python	Создание административных скриптов.	Работа с учебной литературой и документацией. Решение задач с использованием Jupyter Notebook. Выполнение домашних заданий.
10. Сетевое программирование на Python	Создание веб-сервиса.	Работа с учебной литературой и документацией. Решение задач с использованием Jupyter Notebook. Выполнение домашних заданий.
11. Тестирование программ на Python	Экстремальное программирование.	Работа с учебной литературой и документацией. Решение задач с использованием Jupyter Notebook. Выполнение домашних заданий.

Наименование тем (разделов) дисциплины	Перечень вопросов, отводимых на самостоятельное освоение	Формы внеаудиторной самостоятельной работы
12. Документирование и развертывание программ на Python	Непрерывная интеграция.	Работа с учебной литературой и документацией. Решение задач с использованием Jupyter Notebook. Выполнение домашних заданий.

6.2.Перечень вопросов, заданий, тем для подготовки к текущему контролю

Пример задач контрольной работы №1

1. Реализовать программу, с которой можно играть в логическую игру «Быки и коровы» (описание правил игры: <http://побомозг.рф/Articles/BullsAndCowsRules>). Программа загадывает число, пользователь вводит очередной вариант отгадываемого числа, программа возвращает количество быков и коров и в случае выигрыша игрока сообщает о победе и завершается. Сама программа НЕ ходит, т.е. не пытается отгадать число загаданное игроком.

Взаимодействие с программой производится через консоль, при запросе данных от пользователя программа сообщает, что ожидает от пользователя и проверяет корректность ввода.

2. Реализовать программу, при помощи которой 2 игрока могут играть в «Крестики-нолики» на поле 3 на 3. Взаимодействие с программой производится через консоль. Игровое поле изображается в виде трех текстовых строк и перерисовывается при каждом изменении состояния поля. При запросе данных от пользователя программа сообщает, что ожидает от пользователя (в частности, координаты новой отметки на поле) и проверяет корректность ввода. Программа должна уметь автоматически определять, что партия окончена, и сообщать о победе одного из игроков или о ничьей. Сама программа НЕ ходит, т.е. не пытается ставить крестики и нолики с целью заполнить линию.

3. Реализовать программу, при помощи которой 2 игрока могут играть в игру «Супер ним». Правила игры следующие. На шахматной доске в некоторых клетках случайно разбросаны фишки или пуговицы. Игроки ходят по очереди. За один ход можно снять все фишки с какой-либо горизонтали или вертикали, на которой они есть. Выигрывает тот, кто заберет последние фишки.

Взаимодействие с программой производится через консоль. Игровое поле изображается в виде текстовых строк и перерисовывается при каждом изменении состояния поля. При запросе данных от пользователя программа сообщает, что ожидает от пользователя (в частности, координаты новой отметки на поле) и проверяет корректность ввода. Программа должна уметь автоматически определять, что партия окончена, и сообщать о победе одного из игроков. Сама программа НЕ ходит, т.е. не пытается выбирать строки или столбцы с целью

победить в игре.

4. Реализовать программу, с которой можно играть в игру «19». Правила игры следующие. Нужно выписать подряд числа от 1 до 19: в строчку до 9, а потом начать следующую строку, в каждой клетке по 1 цифре (не числу (см пример по ссылке)). Затем игроку необходимо вычеркнуть парные цифры или дающие в сумме 10. Условие - пары должны находиться рядом или через зачеркнутые цифры по горизонтали или по вертикали. После того как все возможные пары вычеркнуты, оставшиеся цифры переписываются в конец таблицы. Цель - полностью вычеркнуть все цифры. (описание правил игры <http://podelki-fox.ru/igry-dlya-detey-na-bumage-s-chislami/>)

Взаимодействие с программой производится через консоль. Игровое поле изображается в виде трех текстовых строк и перерисовывается при каждом изменении состояния поля. При запросе данных от пользователя программа сообщает, что ожидает от пользователя (в частности, координаты очередного хода) и проверяет корректность ввода. Программа должна уметь автоматически определять, что нужно выписать новые строки с цифрами и то, что партия окончена. Сама программа НЕ ходит, т.е. не пытается выбирать пары цифр с целью окончить игру.

5. Реализовать программу, при помощи которой 3 игрока могут играть в игру «Лоскутное одеяло». Правила игры следующие. На поле, имеющем размер 4 на 5 клеток за один ход каждый игрок должен заполнить одну клетку своим символом. Игрок старается, чтобы его символы были как можно дальше друг от друга. В ходе игры ведется подсчет очков: за каждое соседство клеток с одинаковыми символами игроку, владельцу символа добавляется одно штрафное очко. Соседними считаются клетки, имеющие общую сторону или расположенные наискосок друг от друга. Выигрывает тот, у кого в конце игры меньше всего штрафных очков.

Взаимодействие с программой производится через консоль. Игровое поле изображается в виде 4 текстовых строк и перерисовывается при каждом изменении состояния поля. При запросе данных от пользователя программа сообщает, что ожидает от пользователя (например, координаты очередного хода) и проверяет корректность ввода. Программа должна уметь автоматически определять количество штрафных очков и окончание партии и ее победителя.

Сама программа НЕ ходит, т.е. не пытается заполнять клетки символами с целью выиграть игру.

6. Реализовать программу, при помощи которой 2 игрока могут играть в игру «Клондайк». Правила игры следующие. Игра ведётся на игровом поле размером 10 на 10 клеток. Игроки по очереди выставляют в любую свободную клетку по отметке, и тот игрок, после чьего хода получилась цепочка длиной хотя бы в 3 отметке, проигрывает. При этом в цепочке считаются как свои отметки, так и отметки соперника, у игровых фишек как бы нет хозяина. Цепочка - это ряд фишек, следующая фишка в котором примыкает к предыдущей с любого из 8-ми направлений. (описание правил игры: <https://www.iqfun.ru/printable-puzzles/klondike-igra.shtml>)

Взаимодействие с программой производится через консоль. Игровое поле изображается в виде 10 текстовых строк и перерисовывается при каждом изменении состояния поля. При запросе данных от пользователя программа сообщает, что ожидает от пользователя (например, координаты очередного хода) и проверяет корректность ввода. Программа должна уметь автоматически определять окончание партии и ее победителя.

Сама программа НЕ ходит, т.е. не пытается ставить в клетки отметки с целью выиграть игру.

7. Реализовать программу, при помощи которой 2 игрока могут играть в игру «Максит». Правила игры следующие. В клетках квадрата 3 на 3 пишутся случайные числа из диапазона от 1 до 9. Начинаящий выбирает любое понравившееся ему число и вычеркивает его, прибавляя к своей сумме. Вторым игроком может выбрать любое из оставшихся чисел того столбца, в котором первый игрок делал свой предыдущий ход. Он тоже вычеркивает выбранное число, прибавляя его к своей сумме. Первый игрок далее поступает аналогично, выбирая число-кандидата из той строки, в которой второй игрок ходил перед этим. Может так случиться, что у какого-то игрока не будет хода. Тогда его соперник продолжает игру, делая ход в той же строке (для первого игрока) или в том же столбце (для второго игрока), что и до этого. Игра заканчивается, когда оба играющих не имеют ходов. Результат определяется по набранным суммам, у кого она больше, тот и выиграл. При равенстве сумм фиксируется ничья. (описание правил игры: <https://www.iqfun.ru/articles/maxit.shtml>).

Взаимодействие с программой производится через консоль. Игровое поле изображается в виде 3 текстовых строк и перерисовывается при каждом изменении состояния поля. При запросе данных от пользователя программа сообщает, что ожидает от пользователя (например, координаты очередного хода) и проверяет корректность ввода. Программа должна уметь автоматически определять сумму очков каждого из игроков и окончание партии и ее победителя.

Сама программа НЕ ходит, т.е. не пытается вычеркивать числа с целью выиграть игру.

8. (*) Реализовать программу, при помощи которой 2 игрока могут играть в игру «Мостики». Правила игры следующие. В ходе игры каждый из игроков старается построить мост с одного своего берега на другой по камням, образующим массив 4 на 5 (4 камня вдоль берега игрока и 5 камней между берегами). У первого игрока - крестики в качестве камней и берега крестиков (левый и правый край поля), у второго игрока - нолики и берега ноликов (верхний и нижний край поля). Игру можно начинать в любой точке поля. За один ход игрок может соединить два своих соседних камня вертикальным или горизонтальным мостиком (обозначаются в текстовом режиме символами «-» и «|»). Мосты первого и второго игрока пересекаться не должны. Выигрывает тот, кто построит непрерывный мост с одного своего берега на другой.

Взаимодействие с программой производится через консоль. Игровое поле изображается в виде 9 текстовых строк и перерисовывается при каждом изменении состояния поля. При запросе данных от пользователя программа сообщает, что

ожидает от пользователя (например, координаты очередного хода) и проверяет корректность ввода. Программа должна уметь автоматически определять недопустимые ходы (приводящие к пересечению мостов соперников) и окончание партии и ее победителя.

Сама программа НЕ ходит, т.е. не пытается строить мосты с целью выиграть игру.

9. (*) Реализовать программу, с которой можно играть в игру «Морской бой». Программа автоматически случайно расставляет на поле размером 10 на 10 клеток: 4 1-палубных корабля, 3 2-палубных корабля, 2 3-палубных корабля и 1 4-х палубный. Между любыми двумя кораблями по горизонтали и вертикали должна быть как минимум 1 незанятая клетка. Программа позволяет игроку ходить, производя выстрелы. Сама программа НЕ ходит т.е. не пытается топить корабли расставленные игроком.

Взаимодействие с программой производится через консоль. Игровое поле изображается в виде 10 текстовых строк и перерисовывается при каждом изменении состояния поля. При запросе данных от пользователя программа сообщает, что ожидает от пользователя (в частности, координаты очередного «выстрела») и проверяет корректность ввода. Программа должна уметь автоматически определять потопление корабля и окончание партии и сообщать об этих событиях.

10. (*) Реализовать программу, с которой можно играть в игру «Пятнашки». Правила игры следующие.

Головоломка представляет собой 15 квадратных костяшек с числами от 1 до 15. Все костяшки заключены в квадратную коробку (поле) размером 4 на 4. При размещении костяшек в коробке остается одно пустое место, которое можно использовать для перемещения костяшек внутри коробки. Цель игры - упорядочить размещение чисел в коробке, разместив их по возрастанию слева направо и сверху вниз, начиная с костяшки с номером 1 в левом верхнем углу и заканчивая пустым местом в правом нижнем углу коробки.

Взаимодействие с программой производится через консоль. Игровое поле изображается в виде 4 текстовых строк и перерисовывается при каждом изменении состояния поля. При запросе данных от пользователя программа сообщает, что ожидает от пользователя (например, координаты очередного хода) и проверяет корректность ввода. Программа должна считать количество сделанных ходов, уметь автоматически определять недопустимые ходы, окончание партии и ее победителя.

Сама программа НЕ ходит, т.е. не пытается упорядочить костяшки с целью выиграть игру.

Пример задач контрольной работы №2

Базовая часть:

Написать калькулятор для строковых выражений вида '<число> <операция> <число>', где <число> - не отрицательное целое число меньше 100, записанное

словами, например "тридцать четыре", <арифметическая операция> - одна из операций "плюс", "минус", "умножить". Результат выполнения операции вернуть в виде текстового представления числа. Пример `calc("двадцать пять плюс тринадцать")` -> "тридцать восемь"

Оформить калькулятор в виде функции, которая принимает на вход строку и возвращает строку.

1. Реализовать поддержку операции деления и остатка от деления и работу с дробными числами (десятичными дробями). Пример: `calc("сорок один и тридцать одна сотая разделить на семнадцать")` -> "два и сорок три сотых". Обращать дробную часть до тысячных включительно, если при делении получаются числа с меньшей дробной частью выполнять округление до тысячных. Сложность: 2

2. Расширение задания 1. Реализовать поддержку десятичной дробной части до миллионных долей включительно. Реализовать корректный вывод информации о периодической десятичной дроби (период дроби вплоть до 4х десятичных знаков). Пример: `calc("девятнадцать и восемьдесят две сотых разделить на девяносто девять")` -> "ноль и двадцать сотых и ноль два в периоде ". Сложность 3

3. Реализовать текстовый калькулятор для выражения из произвольного количества операций с учетом приоритета операций. Пример: `calc("десять плюс два умножить на три минус один")` -> "девятнадцать". (Для реализации рекомендуется использовать алгоритмы основанные на польской инверсной

4. Расширение задания 3. Добавить поддержку приоритета операций с помощью скобок. Пример: `calc("скобка открывается пять плюс два скобка закрывается умножить на три минус один")` -> "двадцать". Сложность 3

5. Добавить возможность использования отрицательных чисел. Пример: `calc("пять минус один")` -> "четыре". Сложность 1

6. Добавить возможность оперировать с дробями (правильными и смешанными). Реализовать поддержку сложения, вычитания и умножения, дробей. Результат операций не должен представлять неправильную дробь, такие результаты нужно превращать в смешанные дроби. Пример: `calc("один и четыре пятых плюс шесть седьмых ")` -> "два и двадцать три тридцать пятых". Сложность 3

7. Расширение задания 6. Добавить автоматическое сокращение дроби в ответе. Пример: `calc("шесть умножить на две третьих")` -> "четыре". Сложность 1

8. Расширение задания 1. Добавить операции возведения в степень и тригонометрические операции синус, косинус, тангенс и константу пи. Допускается как минимум одна из этих функций в выражении с обычными операциями. Пример: `calc("два в степени четыре")` -> "шестнадцать". Пример: `calc("синус от пи разделить на четыре")` -> "ноль и семьсот семь тысячных". Сложность 1 или 2

9. Добавить комбинаторные операции перестановки, размещения и сочетания. Пример: `calc("размещений из трех по два")` -> "шесть". Сложность 1 или 2

10. Диагностировать ошибки: неправильную запись числа; неправильную последовательность чисел и операций; (задание 1) деление на ноль; (задание 3) неправильную последовательность чисел и операций; (задание 4) некорректный баланс и вложенность скобок; (задание 6) некорректную запись числа. Сложность 1 или 2

Критерии балльной оценки различных форм текущего контроля успеваемости

Критерии балльной оценки различных форм текущего контроля успеваемости содержатся в соответствующих методических рекомендациях кафедры «Математика и информатика» Краснодарского филиала Финуниверситета.

7.Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

Оценочные средства предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины «Практикум по программированию».

Фонд оценочных средств включает контрольные материалы для проведения текущего контроля и промежуточной аттестации.

7.1.Описание показателей и критериев оценивания компетенций

Планируемые результаты освоения компетенции (индикатора достижения компетенции)	Уровень освоения				Оценочное средство
	«неудовлетворительно»	«удовлетворительно»	«хорошо»	«отлично»	
УК-10 Способность осуществлять поиск, критически анализировать, обобщать и систематизировать информацию, использовать системный подход для решения поставленных задач					
Четко описывает состав и структуру требуемых данных и информации, грамотно реализует процессы их сбора, обработки и интерпретации					
Знать: состав и структуру требуемых данных и информации, процессы их	Фрагментарное представление о составе и структуре требуемых	Неполные представления о составе и структуре требуемых данных и информации,	Сформированные, но содержащие отдельные пробелы представления о составе и	Сформированные систематические представления о составе и структуре	Вопросы для оценки знаний и умений, задания в виде практико-

Планируемые результаты освоения компетенции (индикатора достижения компетенции)	Уровень освоения				Оценочное средство
	«неудовлетво рительно»	«удовлетвори тельно»	«хорошо»	«отлично»	
сбора, обработки и интерпретаци и	данных и информаци и, процессах их сбора, обработки и интерпрета ции	процессах их сбора, обработки и интерпретаци и	структуре требуемых данных и информации, процессах их сбора, обработки и интерпретации	требуемых данных и информации, процессах их сбора, обработки и интерпретации	ориентирова нных задач, тестовые задания
Уметь: четко описывать состав и структуру требуемых данных и информации, грамотно реализовыват ь процессы их сбора, обработки и интерпретаци и	Фрагментар ное умение четко описывать состав и структуру требуемых данных и информаци и, грамотно реализовыв ать процессы их сбора, обработки и интерпрета ции	Несистемати ческое применение умений четко описывать состав и структуру требуемых данных и информации, грамотно реализовыват ь процессы их сбора, обработки и интерпретаци и	В целом успешное, но содержащее отдельные пробелы умение четко описывать состав и структуру требуемых данных и информации, грамотно реализовывать процессы их сбора, обработки и интерпретации	Сформированн ое умение четко описывать состав и структуру требуемых данных и информации, грамотно реализовывать процессы их сбора, обработки и интерпретации	Вопросы для оценки знаний и умений, задания в виде практико- ориентирова нных задач, тестовые задания
Обосновывает сущность происходящего, выявляет закономерности, понимает природу вариабельности					
Знать: сущность происходяще го, закономернос ти, природу вариабельнос ти	Фрагментар ное представле ние о сущности происходя щего, закономерн ости, природе вариабельн ости	Неполные представлени я о сущности происходяще го, закономернос ти, природе вариабельнос ти	Сформированн ые, но содержащие отдельные пробелы представления о сущности происходящего, закономерности, природе вариабельности	Сформированн ые систематическ ие представления о сущности происходящег о, закономерност и, природе вариабельност и	Вопросы для оценки знаний и умений, задания в виде практико- ориентирова нных задач, тестовые задания

Планируемые результаты освоения компетенции (индикатора достижения компетенции)	Уровень освоения				Оценочное средство
	«неудовлетво рительно»	«удовлетвори тельно»	«хорошо»	«отлично»	
Уметь: обосновать сущность происходяще го, выявлять закономернос ти, понимать природу вариабельнос ти	Фрагментар ное умение обосновать сущность происходя щего, выявлять закономерн ости, понимать природу вариабельн ости	Несистемати ческое применение умений обосновыват ь сущность происходяще го, выявлять закономернос ти, понимать природу вариабельнос ти	В целом успешное, но содержащее отдельные пробелы умение обосновать сущность происходящего, выявлять закономерности, понимать природу вариабельности	Сформированн ое умение обосновать сущность происходящег о, выявлять закономерност и, понимать природу вариабельност и	Вопросы для оценки знаний и умений, задания в виде практико- ориентирова нных задач, тестовые задания
Формулирует признак классификации, выделяет соответствующие ему группы однородных «объектов», идентифицирует общие свойства элементов этих групп, оценивает полноту результатов классификации, показывает прикладное назначение классификационных групп					
Знать: признаки классификаци и, соответствую щие ему группы однородных «объектов», общие свойства элементов этих групп, полноту результатов классификаци и, прикладное назначение классификаци онных групп	Фрагментар ное представле ние о признаках классифика ции, соответству ющих определенн ой группе однородны х «объектов», общих свойствах элементов этих групп, полноте результатов классифика ции, прикладно м назначении классифика ционных групп	Неполные представлени я о признаках классификац ии, соответствую щих определенно й группе однородных «объектов», общих свойствах элементов этих групп, полноте результатов классификац ии, прикладном назначении классификац ионных групп	Сформированны е, но содержащие отдельные пробелы представления о признаках классификации, соответствующи х определенной группе однородных «объектов», общих свойствах элементов этих групп, полноте результатов классификации, прикладном назначении классификацион ных групп	Сформированн ые систематическ ие представления о признаках классификации , соответствую щих определенной группе однородных «объектов», общих свойствах элементов этих групп, полноте результатов классификации , прикладном назначении классификацио нных групп	Вопросы для оценки знаний и умений, задания в виде практико- ориентирова нных задач, тестовые задания

Планируемые результаты освоения компетенции (индикатора достижения компетенции)	Уровень освоения				Оценочное средство
	«неудовлетво рительно»	«удовлетвори тельно»	«хорошо»	«отлично»	
Уметь: формулирова ть признак классификаци и, выделять соответствую щие ему группы однородных «объектов», идентифицир овать общие свойства элементов этих групп, оценивать полноту результатов классификаци и, показывать прикладное назначение классификаци онных групп	Фрагментар ное умение формулиро вать признак классифика ции, выделять соответству ющие ему группы однородны х «объектов», идентифици ровать общие свойства элементов этих групп, оценивать полноту результатов классифика ции, показывать прикладное назначение классифика ционных групп	Несистемати ческое применение умений формулирова ть признак классификац ии, выделять соответствую щие ему группы однородных «объектов», идентифицир овать общие свойства элементов этих групп, оценивать полноту результатов классификац ии, показывать прикладное назначение классификац ионных групп	В целом успешное, но содержащее отдельные пробелы умение формулировать признак классификации, выделять соответствующи е ему группы однородных «объектов», идентифицирова ть общие свойства элементов этих групп, оценивать полноту результатов классификации, показывать прикладное назначение классификацион ных групп	Сформированн ое умение формулироват ь признак классификации , выделять соответствую щие ему группы однородных «объектов», идентифициро вать общие свойства элементов этих групп, оценивать полноту результатов классификации , показывать прикладное назначение классификацио нных групп	Вопросы для оценки знаний и умений, задания в виде практико- ориентирова нных задач, тестовые задания
Грамотно, логично, аргументировано формирует собственные суждения и оценки. Отличает факты от мнений, интерпретаций, оценок и т.д. в рассуждениях других участников деятельности					
Знать: принципы формировани я собственных суждений и оценок, методы Отличий фактов от мнений,	Фрагментар ное представле ние о принципах формирова ния собственны х суждений и оценок, методах	Неполные представлени я о принципах формировани я собственных суждений и оценок, методах отличий	Сформированны е, но содержащие отдельные пробелы представления о принципах формирования собственных суждений и оценок, методах	Сформированн ые систематическ ие представления о принципах формирования собственных суждений и оценок, методах	Вопросы для оценки знаний и умений, задания в виде практико- ориентирова нных задач, тестовые задания

Планируемые результаты освоения компетенции (индикатора достижения компетенции)	Уровень освоения				Оценочное средство
	«неудовлетво рительно»	«удовлетвори тельно»	«хорошо»	«отлично»	
интерпретаци й, оценок и т.д. в рассуждениях других участников деятельности	отличий фактов от мнений, интерпрета ций, оценок и т.д. в рассуждени ях других участников деятельнос ти	фактов от мнений, интерпретаци й, оценок и т.д. в рассуждения х других участников деятельности	отличий фактов от мнений, интерпретаций, оценок и т.д. в рассуждениях других участников деятельности	отличий фактов от мнений, интерпретаций , оценок и т.д. в рассуждениях других участников деятельности	
Уметь: грамотно, логично, аргументиров ано формировать собственные суждения и оценки. Отличать факты от мнений, интерпретаци й, оценок и т.д. в рассуждениях других участников деятельности	Фрагментар ное умение грамотно, логично, аргументир овано формирова ть собственны е суждения и оценки. Отличать факты от мнений, интерпрета ций, оценок и т.д. в рассуждени ях других участников деятельнос ти	Несистемати ческое применение умений грамотно, логично, аргументиров ано формировать собственные суждения и оценки. Отличать факты от мнений, интерпретаци й, оценок и т.д. в рассуждения х других участников деятельности	В целом успешное, но содержащее отдельные пробелы умение грамотно, логично, аргументирован о формировать собственные суждения и оценки. Отличать факты от мнений, интерпретаций, оценок и т.д. в рассуждениях других участников деятельности	Сформированн ое умение грамотно, логично, аргументирова но формировать собственные суждения и оценки. Отличать факты от мнений, интерпретаций , оценок и т.д. в рассуждениях других участников деятельности	Вопросы для оценки знаний и умений, задания в виде практико- ориентирова нных задач, тестовые задания
Аргументированно и логично представляет свою точку зрения посредством и на основе системного описания					
Знать: принципы представлени я своей точки зрения посредством и на основе	Фрагментар ное представле ние о принципах представле ния своей	Неполные представлени я о принципах представлени я своей точки зрения	Сформированны е, но содержащие отдельные пробелы представления о принципах	Сформированн ые систематическ ие представления о принципах представления	Вопросы для оценки знаний и умений, задания в виде практико-

Планируемые результаты освоения компетенции (индикатора достижения компетенции)	Уровень освоения				Оценочное средство
	«неудовлетво рительно»	«удовлетвори тельно»	«хорошо»	«отлично»	
системного описания	точки зрения посредство м и на основе системного описания	посредством и на основе системного описания	представления своей точки зрения посредством и на основе системного описания	своей точки зрения посредством и на основе системного описания	ориентирова нных задач, тестовые задания
Уметь: аргументиров анно и логично представлять свою точку зрения посредством и на основе системного описания	Фрагментар ное умение аргументир ованно и логично представля ть свою точку зрения посредство м и на основе системного описания	Несистемати ческое применение умений аргументиров анно и логично представлять свою точку зрения посредством и на основе системного описания	В целом успешное, но содержащее отдельные пробелы умение аргументирован но и логично представлять свою точку зрения посредством и на основе системного описания	Сформированн ое умение аргументирова нно и логично представлять свою точку зрения посредством и на основе системного описания	Вопросы для оценки знаний и умений, задания в виде практико- ориентирова нных задач, тестовые задания
ПКН-4 Способен проектировать и реализовывать прикладные программные системы в соответствии с анализом задачи и требований к ним					
Демонстрирует базовые знания о существующих математических методах и системах программирования					
Знать: базовые знания о существующ их математическ их методах и системах программиро вания	Фрагментар ное представле ние о базовых знаниях существую щих математиче ских методах и системах программи рования	Неполные представлени я о базовых знаниях существующ их математическ их методах и системах программиро вания	Сформированны е, но содержащие отдельные пробелы представления о базовых знаниях существующих математических методах и системах программирован ия	Сформированн ые систематическ ие представления о базовых знаниях существующих математически х методах и системах программиров ания	Вопросы для оценки знаний и умений, задания в виде практико- ориентирова нных задач, тестовые задания

Планируемые результаты освоения компетенции (индикатора достижения компетенции)	Уровень освоения				Оценочное средство
	«неудовлетво рительно»	«удовлетвори тельно»	«хорошо»	«отлично»	
Уметь: демонстриров ать базовые знания о существующ их математическ их методах и системах программиро вания	Фрагментар ное умение демонстрир овать базовые знания о существую щих математиче ских методах и системах программи рования	Несистемати ческое применение умений демонстриро вать базовые знания о существующ их математическ их методах и системах программиро вания	В целом успешное, но содержащее отдельные пробелы умение демонстрироват ь базовые знания о существующих математических методах и системах программирован ия	Сформированн ое умение демонстрирова ть базовые знания о существующих математически х методах и системах программиров ания	Вопросы для оценки знаний и умений, задания в виде практико- ориентирова нных задач, тестовые задания
Использует и адаптирует существующие математические методы и системы программирования для решения прикладных задач					
Знать: существующ ие математическ ие методы и системы программиро вания для решения прикладных задач	Фрагментар ное представле ние о существую щих математиче ских методах и систем программи рования для решения прикладны х задач	Неполные представлени я о существующ их математическ их методах и систем программиро вания для решения прикладных задач	Сформированны е, но содержащие отдельные пробелы представления о существующих математических методах и систем программирован ия для решения прикладных задач	Сформированн ые систематическ ие представления о существующих математически х методах и систем программиров ания для решения прикладных задач	Вопросы для оценки знаний и умений, задания в виде практико- ориентирова нных задач, тестовые задания
Уметь: использовать и адаптировать существующ ие математическ ие методы и системы программиро вания для	Фрагментар ное умение использова ть и адаптирова ть существую щие математиче ские методы и	Несистемати ческое применение умений использовать и адаптировать существующ ие математическ ие методы и	В целом успешное, но содержащее отдельные пробелы умение использовать и адаптировать существующие математические методы и системы	Сформированн ое умение использовать и адаптировать существующие математически е методы и системы программиров ания для решения	Вопросы для оценки знаний и умений, задания в виде практико- ориентирова нных задач, тестовые задания

Планируемые результаты освоения компетенции (индикатора достижения компетенции)	Уровень освоения				Оценочное средство
	«неудовлетво рительно»	«удовлетвори тельно»	«хорошо»	«отлично»	
решения прикладных задач	системы программи рования для решения прикладны х задач	системы программиро вания для решения прикладных задач	программирован ия для решения прикладных задач	прикладных задач	
Владеет навыками проектирования и разработки компонентов программного обеспечения на основе современных парадигм, технологий и языков программирования					
Знать: навыки проектирован ия и разработки компонентов программног о обеспечения на основе современных парадигм, технологий и языков программиро вания	Фрагментар ное представле ние о навыках проектиров ания и разработка х компоненто в программн ого обеспечени я на основе современн ых парадигм, технологий и языков программи рования	Неполные представлени я о навыках проектирован ия и разработках компонентов программног о обеспечения на основе современных парадигм, технологий и языков программиро вания	Сформированны е, но содержащие отдельные пробелы представления о навыках проектирования и разработках компонентов программного обеспечения на основе современных парадигм, технологий и языков программирован ия	Сформированны е систематическ ие представления о навыках проектировани я и разработках компонентов программного обеспечения на основе современных парадигм, технологий и языков программиров ания	Вопросы для оценки знаний и умений, задания в виде практико- ориентирова нных задач, тестовые задания
Уметь: владеть навыками проектирован ия и разработки компонентов программног о обеспечения на основе современных парадигм, технологий и языков	Фрагментар ное умение владеть навыками проектиров ания и разработки компоненто в программн ого обеспечени я на основе современн	Несистемати ческое применение умений владеть навыками проектирован ия и разработки компонентов программног о обеспечения на основе	В целом успешное, но содержащее отдельные пробелы умение владеть навыками проектирования и разработки компонентов программного обеспечения на основе современных	Сформированн ое умение владеть навыками проектировани я и разработки компонентов программного обеспечения на основе современных парадигм, технологий и языков	Вопросы для оценки знаний и умений, задания в виде практико- ориентирова нных задач, тестовые задания

Планируемые результаты освоения компетенции (индикатора достижения компетенции)	Уровень освоения				Оценочное средство
	«неудовлетворительно»	«удовлетворительно»	«хорошо»	«отлично»	
программирования	ых парадигм, технологий и языков программирования	современных парадигм, технологий и языков программирования	парадигм, технологий и языков программирования	программирования	
Применяет методы машинного обучения для решения прикладных задач анализа данных					
Знать: методы машинного обучения для решения прикладных задач анализа данных	Фрагментарное представление о методах машинного обучения для решения прикладных задач анализа данных	Неполные представления о методах машинного обучения для решения прикладных задач анализа данных	Сформированные, но содержащие отдельные пробелы представления о методах машинного обучения для решения прикладных задач анализа данных	Сформированные систематические представления о методах машинного обучения для решения прикладных задач анализа данных	Вопросы для оценки знаний и умений, задания в виде практико-ориентированных задач, тестовые задания
Уметь: применять методы машинного обучения для решения прикладных задач анализа данных	Фрагментарное умение применять методы машинного обучения для решения прикладных задач анализа данных	Несистематическое применение умений применять методы машинного обучения для решения прикладных задач анализа данных	В целом успешное, но содержащее отдельные пробелы умение применять методы машинного обучения для решения прикладных задач анализа данных	Сформированное умение применять методы машинного обучения для решения прикладных задач анализа данных	Вопросы для оценки знаний и умений, задания в виде практико-ориентированных задач, тестовые задания

7.2. Вопросы для оценки знаний и умений, характеризующих формирование компетенций

Шифр компетенции	Вопросы	Правильный ответ
УК-10	1. Дайте определение системного подхода к решению задач в программировании?	Состоит из взаимосвязанных компонентов, каждый из которых выполняет свою функцию в общем процессе

Шифр компетенции	Вопросы	Правильный ответ
	2.Как осуществляется поиск информации в процессе программирования?	Базы данных
	3.Какие методы можно использовать для критического анализа данных при разработке программного обеспечения?	Проверка корректности данных; валидация, верификация, аудит, контроль.
	4.Какие инструменты можно использовать для обобщения информации при решении программистских задач?	Электронные таблицы; базы данных; инструменты для работы с Big Data
	5.Каким образом осуществляется систематизация информации в контексте практикума по программированию?	Система управления обучением (LMS)
	6.Опишите основные этапы процесса решения поставленной задачи с использованием системного подхода.	Определение проблемы; анализ; разработка; выбор методов и инструментов; тестирование; оценка результатов; внедрение; мониторинг и поддержка
	7.Какими навыками и знаниями должен обладать студент, чтобы успешно осуществлять поиск, анализ, обобщение и систематизацию информации в рамках практикума по программированию?	Знание языков программирования и основ алгоритмики.
	8.Каковы преимущества использования системного подхода к решению программистских задач на практике?	улучшение качества решений, повышение эффективности работы, улучшение коммуникации и координации, укрепление знаний и навыков
	9.Каковы недостатки использования системного подхода к решению программистских задач на практике?	Сложность и занимает больше времени
	10.Какие этические аспекты необходимо учитывать при поиске, анализе и обобщении информации в рамках программистского практикума?	Конфиденциальность, точность и достоверность информации, прозрачность и открытость, а также справедливость и равенство.
	11.Дайте определение понятию «прикладная программная система».	Решение конкретной задачи или класса задач в определенной предметной области
ПKN - 4	12.Какие основные этапы включает в себя процесс	Анализ требований; проектирование архитектуры; реализация модулей; тестирование и отладка; документирование

Шифр компетенции	Вопросы	Правильный ответ
	проектирования прикладной программной системы?	
	13. Дайте определение анализу задачи в прикладной программной системе?	Процесс исследования требований и ограничений
	14. Какие требования могут предъявляться к прикладным программным системам?	Функциональность, надежность, производительность, удобство использования, масштабируемость и безопасность
	15. Опишите основные методы анализа требований к прикладным программным системам.	Интервьюирование пользователей, анализ документации, использование специальных инструментов для сбора и анализа требований, а также проведение тестирования и оценки качества системы
	16. Какие технологии могут использоваться для реализации прикладной программной системы?	Python, Java, C++
	17. В чем заключается процесс тестирования прикладной программной системы?	Проверка работоспособности системы, выявление ошибок и несоответствий требованиям
	18. Какие техники отладки могут применяться для устранения ошибок в прикладных программных системах?	Остановка и трассировка стека
	19. Как следует документировать процесс проектирования и реализации прикладной программной системы для обеспечения прозрачности?	Описание требований к системе, архитектура системы, описание модулей и их взаимодействия, тестовые сценарии и результаты тестирования
	20. В каких случаях может потребоваться изменение требований к системе после ее проектирования и реализации?	Появление новых требований или обнаружение ошибок

7.3. Практико-ориентированные задания

Шифр компетенции	Практико-ориентированные задания	Правильный ответ
УК-10	1. Напечатайте строку 'Mindmajix' 5 раз.	n = 5 x = "Mindmajix"; print(a * n)
	2. Найдите наиболее часто встречающееся значение в списке	test = [1, 2, 3, 9, 2, 7, 3, 5, 9, 9, 9] print(max(set(test), key = test.count))

Шифр компетенции	Практико-ориентированные задания	Правильный ответ
	[1, 2, 3, 9, 2, 7, 3, 5, 9, 9, 9] и выведите ответ на экран.	
	3.Какой результат будет выведен после завершения работы следующего кода? i = 0 while i < 5: print(i) i += 1 if i == 3: break else: print(0)	0 1 2
	4.Как импортировать модули в Python?	import
	5.Напишите функцию на Python для создания треугольника из символов «*»	def pyfunc(r): for x in range(r): print(' '*(r-x-1)+'*'* (2*x+1)) pyfunc(9)
	6.Какой результат будет выведен после завершения работы следующего кода? import numpy as np arr = np.array([1, 3, 2, 4, 5]) print(arr.argsort()[-3:][::-1])	[4 3 1]
	7.Какой результат будет выведен после завершения работы следующего кода? f = None for i in range (5): with open("data.txt", "w") as f: if (i > 2): break print f.closed	True
	8.Как удалить все начальные пробелы в строке: “ Python”	Python”.lstrip
	9.Как бы вы заменили все вхождения подстроки old новой подстрокой new в строке str количество раз count?	str.replace(old, new, count)
ПКН - 4	10.Напишите код бесконечного цикла while на Python, который будет выводить на экран слово Python?	while True: print('Python')
	11.Напишите программу, в которой рассчитывается сумма и произведение цифр положительного трёхзначного числа	a = int(input()) a1 = a // 100 a2 = a % 100 // 10 a3 = a % 10 print('Сумма цифр =', a1 + a2 + a3)

Шифр компетенции	Практико-ориентированные задания	Правильный ответ
		<code>print('Произведение цифр =', a1 * a2 * a3)</code>
	12.Как напечатать сумму всех чисел от 1 до 101?	<code>Print(sum(range(1,102)))</code>
	13.Напишите программу на Python, которая меняет порядок цифр числа на обратный. Заданное число вводится с клавиатуры.	<code>n = int(input("Enter number: ")) reverse = 0 while(n>0): digit = n%10 reverse = reverse*10+digit n=n//10 print("The reverse of the number:",reverse)</code>
	14.Как в Python создать пустой класс?	<code>class my_class: pass object_ = my_class() object_.name = "ClassName" print("Name = ", object_.name)</code>
	15.Как извлечь индексы N максимальных значений в массиве NumPy?	<code>import numpy array = numpy.array([4, 8, 4, 9, 2]) print(array.argsort()[-2:][::-1])</code>
	16.Напишите код Python, чтобы перевернуть заданный список и вывести его элементы в обратном порядке <code>array = [23, 12, 5, 24, 23, 76, 86, 24, 86, 24, 75]</code>	<code>print("Reverse order of array is") for i in range(len(array)-1, -1, -1): print(array[i], end=' ')</code>
	17.Напишите программу, которая читает файл 'file1.txt' слово за словом и выводит на экран.	<code>with open('file1.txt','r') as file: for line in file: for word in line.split(): print(word)</code>
	18.Напишите программу, которая читает файл 'file1.txt' посимвольно и выводит на экран.	<code>file = open('file1.txt', 'r') while 1: char = file.read(1) if not char: break print(char) file.close()</code>

7.4.Тесты

Шифр компетенции	Тестовые задания	Правильный ответ
УК-10	1.Какой результат будет выведен после завершения работы следующего кода? <code>def addItem(listParam): listParam += [1] mylist = [1, 2, 3, 4] addItem(mylist) print(len(mylist))</code>	а

Шифр компетенции	Тестовые задания	Правильный ответ
	a) 5; b) 8; c) 2; d) 1.	
	2.Какой результат будет выведен после завершения работы следующего кода? <code>z=set('abc\$de')</code> <code>'a' in z</code> a) Error; b) True; c) False; d) No output.	b
	3.Какой из следующих операторов Python приведет к выводу ответа: 6? <code>A = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]</code> a) <code>A[2][1];</code> b) <code>A[1][2];</code> c) <code>A[3][2];</code> d) <code>A[2][3].</code>	b
	4.Какой результат будет выведен после завершения работы следующего кода? <code>x = 'abcd'</code> <code>for i in range(len(x)):</code> <code> print(i)</code> a) error; b) 1 2 3 4; c) a b c d; d) 0 1 2 3.	d
	5.Какой результат будет выведен после завершения работы следующего кода? <code>string = "This is a string."</code> <code>string_list = string.split(' ')</code> #delimiter is 'space' character or ' <code>print(string_list)</code> a) error; b) ['This', 'is', 'a', 'string.']; c) ['This', 'is', 'string.']; d) ['This', 'is', 'a.'].]	b
	6.Какой результат будет выведен после завершения работы следующего кода? <code>x = ['ab', 'cd']</code> <code>for i in x:</code> <code> i.upper()</code> <code>print(x)</code> a) ['ab', 'cd']; b) ['AB', 'CD']; c) [None, None]; d) none of the mentioned.	a

Шифр компетенции	Тестовые задания	Правильный ответ
ПКН – 4	7. Каково будет значение следующего выражения Python? $4 + 3 \% 5$ a) 7; b) 2; c) 4; d) 1.	a
	8. Каким будет вывод следующего кода Python? <pre>i = 1 while True: if i%3 == 0: break print(i) i += 1</pre> a) 1 2 3; b) error; c) 1 2; d) ни один из перечисленных.	b
	9. Что означает pip для Python? a) Pip устанавливает Python; b) Pip устанавливает пакеты; c) Предпочтительная программа установки; d) все перечисленное.	c
	10. 9. С каким числом параметров может работать следующая функция Python? <pre>def f(x): def f1(*args, **kwargs): print("Sanfoundry") return x(*args, **kwargs) return f1</pre> a) любое количество; b) 0; c) 1; d) 2.	a
	11. Каким будет вывод следующей функции Python? <pre>len(["hello", 2, 4, 6])</pre> a) Error; b) 6; c) 4; d) 3.	c
	12. Каким будет вывод следующей функции Python? <pre>def foo(x): x[0] = ['def'] x[1] = ['abc'] return id(x) q = ['abc', 'def'] print(id(q) == foo(q))</pre> a) Error;	d

Шифр компетенции	Тестовые задания	Правильный ответ
	b) None; c) False; d) True.	

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

Основная литература

1. Андрианова, А. А. Алгоритмизация и программирование. Практикум : учебное пособие для спо / А. А. Андрианова, Л. Н. Исмагилов, Т. М. Мухтарова. — 2-е изд., стер. — Санкт-Петербург : Лань, 2022. — 240 с. — ISBN 978-5-8114-8948-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/186390> . — Режим доступа: для авториз. пользователей.

2. Быкадорова, Е. А. Программирование. Практикум : учебное пособие для спо / Е. А. Быкадорова. — 2-е изд., стер. — Санкт-Петербург : Лань, 2022. — 60 с. — ISBN 978-5-8114-8952-7. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/185905> . — Режим доступа: для авториз. пользователей.

3. Немцова, Т. И. Программирование на языке высокого уровня. Программирование на языке С++ : учебное пособие / Т.И. Немцова, С.Ю. Голова, А.И. Терентьев ; под ред. Л.Г. Гагариной. — Москва : ФОРУМ : ИНФРА-М, 2024. — 512 с. + Доп. материалы [Электронный ресурс]. — (Среднее профессиональное образование). - ISBN 978-5-8199-0699-6. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2083383>

Дополнительная литература

4. Карякин, М. И. Технологии программирования и компьютерный практикум на языке Python : учебное пособие / М. И. Карякин, К. А. Ватульян, Р. М. Мнухин. — Ростов-на-Дону : ЮФУ, 2022. — 242 с. — ISBN 978-5-9275-4108-9. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/293606> . — Режим доступа: для авториз. пользователей.

5. Каширская, Е. Н. Процедурное программирование: Практикум : учебное пособие / Е. Н. Каширская. — Москва : РТУ МИРЭА, 2020. — 75 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/163905> . — Режим доступа: для авториз. пользователей.

6. Логачев, М. С. Информационные системы и программирование. Специалист по информационным системам. Выпускная квалификационная работа : учебник / М.С. Логачёв. — Москва : ИНФРА-М, 2024. — 576 с. — (Среднее профессиональное образование). - ISBN 978-5-16-015919-5. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2136719>

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

- 1.Личный кабинет обучающегося <https://org.fa.ru>
- 2.Электронная библиотека Финансового университета (ЭБ) <http://elib.fa.ru/>
- 3.Электронно-библиотечная система BOOK.RU <http://www.book.ru>
- 4.Электронно-библиотечная система «Университетская библиотека ОНЛАЙН» <http://biblioclub.ru/>
- 5.Электронно-библиотечная система Znanium <http://www.znaniy.com>
- 6.Электронно-библиотечная система издательства «ЮРАЙТ» <https://urait.ru/>
- 7.Электронно-библиотечная система издательства Проспект <http://ebs.prospekt.org/books>
- 8.Электронно-библиотечная система издательства «Лань» <https://e.lanbook.com/>
- 9.Деловая онлайн-библиотека Alpina Digital <http://lib.alpinadigital.ru/>

10.Методические материалы, определяющие процедуры оценивания знаний и умений, характеризующих степень сформированности компетенций

Рекомендации по подготовке к лекционным занятиям

Изучение дисциплины требует систематического и последовательного накопления знаний и практических навыков, следовательно, пропуски отдельных лекций необходимо сразу наверстывать посредством самостоятельного изучения пропущенной темы и консультаций с преподавателем, ведущим занятия.

Рекомендации по подготовке к практическим (семинарским) занятиям

Студентам следует на каждое практическое занятие приходить с результатами выполненной домашней работы предыдущего семинара. Такое требование связано с тем, что сложные программы обсуждаются и выполняются несколько семинаров подряд, и для работы по теме текущего семинара используется результаты работы на предыдущем семинаре и соответствующей домашней работы.

Самостоятельная работа студентов включает в себя выполнение различного рода заданий, которые ориентированы на более глубокое усвоение материала изучаемой дисциплины и приобретение практических навыков по дисциплине Практикум по программированию.

К выполнению заданий для самостоятельной работы предъявляются следующие требования: задания должны выполняться самостоятельно. Результатом выполнения задания является контрольная работа. Задание может быть выполнено как на компьютере студента (домашнем или в компьютерном классе), так и на компьютере преподавателя (домашнем или установленным в компьютерном классе).

Студентам следует:

- руководствоваться графиком самостоятельной работы, определенным РПД
- выполнять все плановые задания, выдаваемые преподавателем для

самостоятельного выполнения

–разбирать на семинарах и консультациях ошибки в программах и прочие непонятные вопросы.

Форма промежуточной аттестации по дисциплине – *зачет*.

Критерии оценивания знаний и умений, характеризующих степень сформированности компетенций:

- **«зачтено»** выставляется в том случае, если компетенция по дисциплине освоена. Оценка выставляется при получении обучающимся более 50 баллов. При этом он:

знает: состав и структуру требуемых данных и информации, процессы их сбора, обработки и интерпретации; сущность происходящего, закономерности, природу вариабельности; признаки классификации, соответствующие ему группы однородных «объектов», общие свойства элементов этих групп, полноту результатов классификации, прикладное назначение классификационных групп; принципы формирования собственных суждений и оценок, методы отличий фактов от мнений, интерпретаций, оценок и т.д. в рассуждениях других участников деятельности; принципы представления своей точки зрения посредством и на основе системного описания; базовые знания о существующих математических методах и системах программирования; существующие математические методы и системы программирования для решения прикладных задач; навыки проектирования и разработки компонентов программного обеспечения на основе современных парадигм, технологий и языков программирования; методы машинного обучения для решения прикладных задач анализа данных.

умеет: четко описывать состав и структуру требуемых данных и информации, грамотно реализовывать процессы их сбора, обработки и интерпретации; обосновать сущность происходящего, выявлять закономерности, понимать природу вариабельности; формулировать признак классификации, выделять соответствующие ему группы однородных «объектов», идентифицировать общие свойства элементов этих групп, оценивать полноту результатов классификации, показывать прикладное назначение классификационных групп; грамотно, логично, аргументировано формировать собственные суждения и оценки. Отличать факты от мнений, интерпретаций, оценок и т.д. в рассуждениях других участников деятельности; аргументированно и логично представлять свою точку зрения посредством и на основе системного описания; демонстрировать базовые знания о существующих математических методах и системах программирования; использовать и адаптировать существующие математические методы и системы программирования для решения прикладных задач; владеть навыками проектирования и разработки компонентов программного обеспечения на основе современных парадигм, технологий и языков программирования; применять методы машинного обучения для решения прикладных задач анализа данных.

- **«не зачтено»** выставляется в том случае, если компетенция не освоена, ответы содержат существенные ошибки и обучающимся получено менее 50 баллов. При этом он:

не знает: состав и структуру требуемых данных и информации, процессы их сбора, обработки и интерпретации; сущность происходящего, закономерности, природу вариабельности; признаки классификации, соответствующие ему группы однородных «объектов», общие свойства элементов этих групп, полноту результатов классификации, прикладное назначение классификационных групп; принципы формирования собственных суждений и оценок, методы отличий фактов от мнений, интерпретаций, оценок и т.д. в рассуждениях других участников деятельности; принципы представления своей точки зрения посредством и на основе системного описания; базовые знания о существующих математических методах и системах программирования; существующие математические методы и системы программирования для решения прикладных задач; навыки проектирования и разработки компонентов программного обеспечения на основе современных парадигм, технологий и языков программирования; методы машинного обучения для решения прикладных задач анализа данных.

не умеет: четко описывать состав и структуру требуемых данных и информации, грамотно реализовывать процессы их сбора, обработки и интерпретации; обосновать сущность происходящего, выявлять закономерности, понимать природу вариабельности; формулировать признак классификации, выделять соответствующие ему группы однородных «объектов», идентифицировать общие свойства элементов этих групп, оценивать полноту результатов классификации, показывать прикладное назначение классификационных групп; грамотно, логично, аргументировано формировать собственные суждения и оценки. Отличать факты от мнений, интерпретаций, оценок и т.д. в рассуждениях других участников деятельности; аргументированно и логично представлять свою точку зрения посредством и на основе системного описания; демонстрировать базовые знания о существующих математических методах и системах программирования; использовать и адаптировать существующие математические методы и системы программирования для решения прикладных задач; владеть навыками проектирования и разработки компонентов программного обеспечения на основе современных парадигм, технологий и языков программирования; применять методы машинного обучения для решения прикладных задач анализа данных.

11.Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем

11.1.Комплект лицензионного программного обеспечения

- Пакет офисных программ
- Антивирус Kaspersky

11.2.Современные профессиональные базы данных и информационные справочные системы

- Информационно-правовая система «Консультант Плюс»;
- Информационно-правовая система «Гарант»;
- Система комплексного раскрытия информации «СКРИН»: <https://skrin.ru>

11.3.Сертифицированные программные и аппаратные средства защиты информации

- не предусмотрены.

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

Материально-техническая база Финансового университета Краснодарского филиала, необходимая для осуществления образовательного процесса по данной дисциплине, в соответствии с требованиями ФОС включает в себя специальные помещения для проведения лекций, семинарских занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации; помещения укомплектованы специализированной мебелью и техническими средствами обучения, необходимыми для представления информации большой аудитории.

Помещения для самостоятельной работы студентов включают в себя библиотеку с читальным залом, укомплектованную в соответствии с существующими нормами необходимой учебной и учебно-методической литературой и иными материалами; медиатеку с выходом в Интернет, компьютерные классы с возможностью работы в Интернет; аудитории для консультационной деятельности.