

Документ подписан простой электронной подписью.
Информация о владельце:

ФИО: Кандрашина Елена Александровна

Должность: И.о. ректора ФГАОУ ВО «Самарский государственный экономический университет»

Дата подписания: 22.07.2024 14:06:28

Уникальный программный ключ:

2db64eb9605ce27edd3b8e8fdd32c70e0674ddd2

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Самарский государственный экономический университет»

Институт Институт экономики предприятий

Кафедра Прикладной информатики

УТВЕРЖДЕНО

Ученым советом Университета

(протокол № 10 от 30 мая 2024 г.)

РАБОЧАЯ ПРОГРАММА

Наименование дисциплины Б1.О.06 Языки программирования и библиотеки

Основная профессиональная образовательная программа 09.04.03 Прикладная информатика программа
Искусственный интеллект и большие данные

Квалификация (степень) выпускника магистр

Самара 2024

Содержание (рабочая программа)

Стр.

- 1 Место дисциплины в структуре ОП
- 2 Планируемые результаты обучения по дисциплине, обеспечивающие достижение планируемых результатов обучения по программе
- 3 Объем и виды учебной работы
- 4 Содержание дисциплины
- 5 Материально-техническое и учебно-методическое обеспечение дисциплины
- 6 Фонд оценочных средств по дисциплине

Целью изучения дисциплины является формирование результатов обучения, обеспечивающих достижение планируемых результатов освоения образовательной программы.

1. Место дисциплины в структуре ОП

Дисциплина Языки программирования и библиотеки входит в обязательную часть блока Б1. Дисциплины (модули)

Предшествующие дисциплины по связям компетенций: Методы оптимизации, Современные проблемы и тренды прикладной информатики, Методы машинного обучения, Иностранный язык в профессиональной деятельности, Системы обработки и анализа больших массивов данных

Последующие дисциплины по связям компетенций: Массово параллельные вычисления для ускорения машинного обучения, Управление проектами создания интеллектуальных информационных систем, Современные методы проектирования систем искусственного интеллекта

2. Планируемые результаты обучения по дисциплине, обеспечивающие достижение планируемых результатов обучения по программе

Изучение дисциплины Языки программирования и библиотеки в образовательной программе направлено на формирование у обучающихся следующих компетенций:

Общепрофессиональные компетенции (ОПК):

ОПК-1 - Способен самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте;

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
ОПК-1	ОПК-1.1: Знать:	ОПК-1.2: Уметь:	ОПК-1.3: Владеть (иметь навыки):
	особенности приобретения, развития и применения математических, естественнонаучных, социально-экономических и профессиональных знаний для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте	самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте	навыками приобретения, развития и применения математических, естественнонаучных, социально-экономических и профессиональных знаний для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте

ОПК-2 - Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач;

Планируемые	Планируемые результаты обучения по дисциплине
-------------	--

е результаты обучения по программе			
ОПК-2	ОПК-2.1: Знать:	ОПК-2.2: Уметь:	ОПК-2.3: Владеть (иметь навыки):
	особенности разработки оригинальных алгоритмов и программных средств, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач	разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач	навыками разработки оригинальных алгоритмов и программных средств, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач

ОПК-4 - Способен применять на практике новые научные принципы и методы исследований;

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
ОПК-4	ОПК-4.1: Знать:	ОПК-4.2: Уметь:	ОПК-4.3: Владеть (иметь навыки):
	особенности применения на практике новых научных принципов и методов исследований	применять на практике новые научные принципы и методы исследований	навыками применения на практике новых научных принципов и методов исследований

ОПК-5 - Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем;

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
ОПК-5	ОПК-5.1: Знать:	ОПК-5.2: Уметь:	ОПК-5.3: Владеть (иметь навыки):
	особенности разработки и модернизации программного и аппаратного обеспечения информационных и автоматизированных систем	разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем	навыками разработки и модернизации программного и аппаратного обеспечения информационных и автоматизированных систем

ОПК-8 - Способен осуществлять эффективное управление разработкой программных средств и проектов.

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
ОПК-8	ОПК-8.1: Знать:	ОПК-8.2: Уметь:	ОПК-8.3: Владеть (иметь навыки):

			навыки):
	особенности эффективного управления разработкой программных средств и проектов	осуществлять эффективное управление разработкой программных средств и проекто	навыками эффективного управления разработкой программных средств и проектов

Профессиональные компетенции (ПК):

ПК-3 - Способен использовать и развивать методы научных исследований и инструментарий в области управления проектами в области ИТ

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
ПК-3	ПК-3.1: Знать:	ПК-3.2: Уметь:	ПК-3.3: Владеть (иметь навыки):
	методы научных исследований и инструментарий в области управления проектами в области ИТ	использовать и развивать методы научных исследований и инструментарий в области управления проектами в области ИТ	навыками использования и применения методов научных исследований и инструментария в области управления проектами в области ИТ

ПК-1 - Способен разрабатывать план конфигурационного управления

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
ПК-1	ПК-1.1: Знать:	ПК-1.2: Уметь:	ПК-1.3: Владеть (иметь навыки):
	системы контроля версий и поддержки конфигурационного управления	разрабатывать план конфигурационного управления	навыками разработки плана конфигурационного управления

3. Объем и виды учебной работы

Учебным планом предусматриваются следующие виды учебной работы по дисциплине:

Очная форма обучения

Виды учебной работы	Всего час/ з.е.
	Сем 2
Контактная работа, в том числе:	14.3/0.4
Занятия лекционного типа	4/0.11
Занятия семинарского типа	8/0.22
Индивидуальная контактная работа (ИКР)	0.3/0.01
Групповая контактная работа (ГКР)	2/0.06
Самостоятельная работа:	131.7/3.66
Промежуточная аттестация	34/0.94
Вид промежуточной аттестации:	
Экзамен	Экз
Общая трудоемкость (объем части образовательной программы): Часы	180

Зачетные единицы	5
------------------	---

заочная форма

Виды учебной работы	Всего час/ з.е.
	Сем 2
Контактная работа, в том числе:	14.3/0.4
Занятия лекционного типа	4/0.11
Занятия семинарского типа	8/0.22
Индивидуальная контактная работа (ИКР)	0.3/0.01
Групповая контактная работа (ГКР)	2/0.06
Самостоятельная работа:	131.7/3.66
Промежуточная аттестация	34/0.94
Вид промежуточной аттестации: Экзамен	Экз
Общая трудоемкость (объем части образовательной программы): Часы	180
Зачетные единицы	5

4. Содержание дисциплины

4.1. Разделы, темы дисциплины и виды занятий:

Тематический план дисциплины Языки программирования и библиотеки представлен в таблице.

Разделы, темы дисциплины и виды занятий Очная форма обучения

№ п/п	Наименование темы (раздела) дисциплины	Контактная работа				Самостоятельная работа	Планируемые результаты обучения в соотношении с результатами обучения по образовательной программе
		Лекции	Занятия семинарского типа	ИКР	ГКР		
	Практич. занятия						
1.	Основы программирования на языке Python	2	4			50	ОПК-1.1, ОПК-1.2, ОПК-1.3, ОПК-2.1, ОПК-2.2, ОПК-2.3, ОПК-4.1, ОПК-4.2, ОПК-4.3, ОПК-5.1, ОПК-5.2, ОПК-5.3, ОПК-8.1, ОПК-8.2, ОПК-8.3, ПК-3.1, ПК-3.2, ПК-3.3, ПК

							-1.1, ПК-1.2, ПК- 1.3
2.	Структуры данных. Объектно-ориентированный подход	2	4			81,7	ОПК-1.1, ОПК-1.2, ОПК-1.3, ОПК-2.1, ОПК-2.2, ОПК-2.3, ОПК-4.1, ОПК-4.2, ОПК-4.3, ОПК-5.1, ОПК-5.2, ОПК-5.3, ОПК-8.1, ОПК-8.2, ОПК-8.3, ПК-3.1, ПК-3.2, ПК-3.3, ПК -1.1, ПК-1.2, ПК- 1.3
	Контроль	34					
	Итого	4	8	0.3	2	131.7	

заочная форма

№ п/п	Наименование темы (раздела) дисциплины	Контактная работа				Самостоятельная работа	Планируемые результаты обучения в соотношении с результатами обучения по образователь ной программе
		Лекции	Занятия семинарс кого типа	ИКР	ГКР		
			Практич. занятия				
1.	Основы программирования на языке Python	2	2			50	ОПК-1.1, ОПК-1.2, ОПК-1.3, ОПК-2.1, ОПК-2.2, ОПК-2.3, ОПК-4.1, ОПК-4.2, ОПК-4.3, ОПК-5.1, ОПК-5.2, ОПК-5.3, ОПК-8.1, ОПК-8.2, ОПК-8.3, ПК-3.1, ПК-3.2, ПК-3.3, ПК -1.1, ПК-1.2, ПК- 1.3

2.	Структуры данных. Объектно-ориентированный подход	2	4			81,7	ОПК-1.1, ОПК-1.2, ОПК-1.3, ОПК-2.1, ОПК-2.2, ОПК-2.3, ОПК-4.1, ОПК-4.2, ОПК-4.3, ОПК-5.1, ОПК-5.2, ОПК-5.3, ОПК-8.1, ОПК-8.2, ОПК-8.3, ПК-3.1, ПК-3.2, ПК-3.3, ПК -1.1, ПК-1.2, ПК- 1.3
	Контроль	34					
	Итого	4	8	0.3	2	131.7	

4.2 Содержание разделов и тем

4.2.1 Контактная работа

Тематика занятий лекционного типа

№п/п	Наименование темы (раздела) дисциплины	Вид занятия лекционного типа*	Тематика занятия лекционного типа
1.	Основы программирования на языке Python	лекция	Язык Python. Основные понятия. Правила именования. Типы данных. Оператор присваивания. Ввод-вывод данных. Вычисления. Арифметические операторы. Условный оператор if, операторы сравнения, логические операторы. Блоки команд. Циклы while и for. Итераторы и генераторы. Строки. Срезы
2.	Структуры данных. Объектно-ориентированный подход	лекция	Структуры данных: списки, множества, словари. Объявление и вызов функций. Рекурсия. Пространства имен и области видимости. Объекты. Классы. Наследование классов

*лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации

педагогическими работниками организации и (или) лицами, привлекаемыми организацией к реализации образовательных программ на иных условиях, обучающимся

Тематика занятий семинарского типа

№п/п	Наименование темы (раздела) дисциплины	Вид занятия семинарского типа**	Тематика занятия семинарского типа
1.	Основы программирования на языке Python	практическое занятие	Сравнение основных сред и языков программирования. Разработка простейшего приложения на языке Python
		практическое занятие	Основные конструкции языка Python. Работа со строками в Python
2.	Структуры данных. Объектно-ориентированный подход	практическое занятие	Структуры данных на языке Python. Функции на языке Python
		практическое занятие	Классы на языке Python. Работа с файлами и файловой системой

** семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия

Иная контактная работа

При проведении учебных занятий СГЭУ обеспечивает развитие у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств (включая при необходимости проведение интерактивных лекций, групповых дискуссий, ролевых игр, тренингов, анализ ситуаций и имитационных моделей, преподавание дисциплин (модулей) в форме курсов, составленных на основе результатов научных исследований, проводимых организацией, в том числе с учетом региональных особенностей профессиональной деятельности выпускников и потребностей работодателей).

Формы и методы проведения иной контактной работы приведены в Методических указаниях по основной профессиональной образовательной программе.

4.2.2 Самостоятельная работа

№п/п	Наименование темы (раздела) дисциплины	Вид самостоятельной работы ***
1.	Основы программирования на языке Python	- тестирование
2.	Структуры данных. Объектно-ориентированный подход	- тестирование

*** самостоятельная работа в семестре, написание курсовых работ, докладов, выполнение контрольных работ

5. Материально-техническое и учебно-методическое обеспечение дисциплины

5.1 Литература:

Основная литература

1. Тузовский, А. Ф. Объектно-ориентированное программирование : учебное пособие для вузов / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2024. — 213 с. — (Высшее образование). — ISBN 978-5-534-16316-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/537332>

Дополнительная литература

1. Гниденко, И. Г. Технологии и методы программирования : учебное пособие для вузов / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 248 с. — (Высшее образование). — ISBN 978-5-534-18130-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/536903>

5.2. Перечень лицензионного программного обеспечения

5.3 Современные профессиональные базы данных, к которым обеспечивается доступ обучающихся

1. Профессиональная база данных «Информационные системы Министерства экономического развития Российской Федерации в сети Интернет» (Портал «Официальная Россия» - <http://www.gov.ru/>)
2. Профессиональная база данных «Финансово-экономические показатели Российской Федерации» (Официальный сайт Министерства финансов РФ - <https://www.minfin.ru/ru/>)
3. Профессиональная база данных «Официальная статистика» (Официальный сайт Федеральной службы государственной статистики - <http://www.gks.ru/>)

5.4. Информационно-справочные системы, к которым обеспечивается доступ обучающихся

1. справочно-правовая система «Консультант Плюс»
2. справочно-правовая система «ГАРАНТ-Максимум»

5.5. Специальные помещения

Учебные аудитории для проведения занятий лекционного типа	Комплекты ученической мебели Мультимедийный проектор Доска Экран
Учебные аудитории для проведения практических занятий (занятий семинарского типа)	Комплекты ученической мебели Мультимедийный проектор Доска Экран Компьютеры с выходом в сеть «Интернет» и ЭИОС СГЭУ
Учебные аудитории для групповых и индивидуальных консультаций	Комплекты ученической мебели Мультимедийный проектор Доска Экран Компьютеры с выходом в сеть «Интернет» и ЭИОС СГЭУ
Учебные аудитории для текущего контроля и промежуточной аттестации	Комплекты ученической мебели Мультимедийный проектор Доска Экран Компьютеры с выходом в сеть «Интернет» и ЭИОС СГЭУ
Помещения для самостоятельной работы	Комплекты ученической мебели Мультимедийный проектор Доска Экран

	Компьютеры с выходом в сеть «Интернет» и ЭИОС СГЭУ
Помещения для хранения и профилактического обслуживания оборудования	Комплекты специализированной мебели для хранения оборудования

5.6 Лаборатории и лабораторное оборудование

Лаборатория информационных технологий в профессиональной деятельности	Комплекты ученической мебели Мультимедийный проектор Доска Экран Компьютеры с выходом в сеть «Интернет» и ЭИОС СГЭУ Лабораторное оборудование
---	--

6. Фонд оценочных средств по дисциплине Языки программирования и библиотеки:

6.1. Контрольные мероприятия по дисциплине

Вид контроля	Форма контроля	Отметить нужное знаком « + »
Текущий контроль	Оценка докладов	-
	Устный/письменный опрос	-
	Тестирование	+
	Практические задачи	-
	Оценка контрольных работ (для заочной формы обучения)	+
Промежуточный контроль	Экзамен	+

Порядок проведения мероприятий текущего и промежуточного контроля определяется Методическими указаниями по основной профессиональной образовательной программе высшего образования; Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры в федеральном государственном автономном образовательном учреждении высшего образования «Самарский государственный экономический университет».

6.2. Планируемые результаты обучения по дисциплине, обеспечивающие достижение планируемых результатов обучения по программе

Общепрофессиональные компетенции (ОПК):

ОПК-1 - Способен самостоятельно приобретать, развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте;

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
	ОПК-1.1: Знать: особенности приобретения, развития и применения математических,	ОПК-1.2: Уметь: самостоятельно приобретать,	ОПК-1.3: Владеть (иметь навыки): навыками приобретения, развития

	естественнонаучных, социально-экономических и профессиональных знаний для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте	развивать и применять математические, естественнонаучные, социально-экономические и профессиональные знания для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте	и применения математических, естественнонаучных, социально-экономических и профессиональных знаний для решения нестандартных задач, в том числе в новой или незнакомой среде и в междисциплинарном контексте
Пороговый	принципы, методы и средства анализа и структурирования профессиональной информации	анализировать профессиональную информацию, выделять в ней главное, структурировать, оформлять и представлять в виде аналитических обзоров	навыками анализа профессиональной информации, выделения в ней главного, структурирования, оформления и представления в виде аналитических обзоров
Стандартный (в дополнение к пороговому)	новые научные принципы и методы исследований	применять на практике новые научные принципы и методы исследований	навыки применения новых научных принципов и методов исследований
Повышенный (в дополнение к пороговому, стандартному)	современное программное и аппаратное обеспечение информационных и автоматизированных систем	модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем для решения профессиональных задач	навыки модернизации программного и аппаратного обеспечения информационных и автоматизированных систем для решения профессиональных задач

ОПК-2 - Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач;

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
	ОПК-2.1: Знать: особенности разработки оригинальных	ОПК-2.2: Уметь: разрабатывать	ОПК-2.3: Владеть (иметь навыки): навыками разработки

	алгоритмов и программных средств, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач	оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач	оригинальных алгоритмов и программных средств, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач
Пороговый	архитектуру информационных систем предприятий и организаций; методологии и технологии реинжиниринга, проектирования и аудита прикладных информационных систем различных классов	выбирать методологию и технологию проектирования информационных систем; обосновывать архитектуру ИС; управлять проектами ИС на всех стадиях жизненного цикла, оценивать эффективность и качество проекта	навыки выбора методологии и технологии проектирования информационных систем; обоснования архитектуры ИС; управления проектами ИС на всех стадиях жизненного цикла, оценки эффективности и качество проекта
Стандартный (в дополнение к пороговому)	инструментальные средства поддержки технологии проектирования и аудита информационных систем и сервисов; методы оценки экономической эффективности и качества, управления надежностью и информационной безопасностью; особенности процессного подхода к управлению прикладными ИС	применять современные методы управления проектами и сервисами ИС; использовать инновационные подходы к проектированию ИС	Навыки применения современных методов управления проектами и сервисами ИС; использования инновационных подходов к проектированию ИС
Повышенный (в дополнение к пороговому, стандартному)	современные ИКТ в процессном управлении; системы управления качеством; концептуальное моделирование процессов управления знаниями; архитектуру систем управления знаниями; онтологии знаний; подсистемы сбора, фильтрации, накопления, доступа, генерации и распространения знаний	принимать решения по информатизации и предприятий в условиях неопределенности; проводить реинжиниринг прикладных и информационных процессов; обосновывать архитектуру системы управления знаниями	Навыки принятия решений по информатизации предприятий в условиях неопределенности; проведения реинжиниринга прикладных и информационных процессов; обоснования архитектуры системы управления знаниями

ОПК-4 - Способен применять на практике новые научные принципы и методы исследований;

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
	ОПК-4.1: Знать:	ОПК-4.2: Уметь:	ОПК-4.3: Владеть (иметь навыки):
	особенности применения на практике новых научных принципов и методов исследований	применять на практике новые научные принципы и методы исследований	навыками применения на практике новых научных принципов и методов исследований
Пороговый	методы представления, сбора и обработки информации в современные формализованные математические, информационно-логические и логико-семантические модели и методы представления, сбора и обработки информации	осуществлять поиск и отбор информации, необходимой для решения конкретной задачи	современными компьютерными технологиями для решения практических задач
Стандартный (в дополнение к пороговому)	современные компьютерные технологии для решения практических задач получения, хранения, обработки и передачи информации	использовать современные компьютерные технологии для решения практических задач получения, хранения, обработки и передачи информации	использования современных компьютерных технологий для решения практических задач получения, хранения, обработки и передачи
Повышенный (в дополнение к пороговому, стандартному)	закономерности, определяющие место информатики в общей картине мира; средства и возможности языков программирования высокого уровня, основы структурного объектно-ориентированного программирования, основы создания Web-ресурсов, возможности различных современных информационных технологий и подходы к их использованию в профессиональной деятельности, профессиональную лексику	использовать различные информационные технологии и прикладные программные средства для оформления результатов своей деятельности	навыками создания программ с применением инструментальных средств поддержки технологий программирования, навыками использования разнообразного прикладного программного обеспечения

ОПК-5 - Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем;

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
	ОПК-5.1: Знать:	ОПК-5.2: Уметь:	ОПК-5.3: Владеть (иметь навыки):
	особенности разработки и модернизации программного и аппаратного обеспечения	разрабатывать и модернизировать программное и	навыками разработки и модернизации программного и

	информационных и автоматизированных систем	аппаратное обеспечение информационных и автоматизированных систем	аппаратного обеспечения информационных и автоматизированных систем
Пороговый	алгоритмические языки программирования, операционные системы и оболочки, современные среды разработки программного обеспечения	составлять алгоритмы, писать и отлаживать коды на языке программирования, тестировать работоспособность программы, интегрировать программные модули	навыками программирования, отладки и тестирования прототипов программно-технических комплексов задач
Стандартный (в дополнение к пороговому)	языки программирования высокого уровня; основные этапы создания прикладного программного обеспечения	разрабатывать, внедрять и адаптировать прикладное программное обеспечение	различными парадигмами программирования; методами внедрения и адаптации прикладного программного обеспечения
Повышенный (в дополнение к пороговому, стандартному)	методы программирования приложения и создания программных прототипов решения прикладных задач	использовать методы программирования для создания программных прототипов решения	навыками применения методов программирования для создания программных прототипов решения

ОПК-8 - Способен осуществлять эффективное управление разработкой программных средств и проектов.

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
	ОПК-8.1: Знать: особенности эффективного управления разработкой программных средств и проектов	ОПК-8.2: Уметь: осуществлять эффективное управление разработкой программных средств и проектов	ОПК-8.3: Владеть (иметь навыки): навыками эффективного управления разработкой программных средств и проектов
Пороговый	Принципы коллективной разработки ПО	Работать над сложным программным проектом в команде	Навыками по разработке программных продуктов в коллективе
Стандартный (в дополнение к пороговому)	Принципы структурного и модульного программирования	Разработать алгоритм решения поставленной	навыками по нахождению алгоритмического решения поставленной

		задачи, разбить программу на модули	задачи
Повышенный (в дополнение к пороговому, стандартному)	Методиками освоения новых сред программирования	применять в профессиональной деятельности библиотеки и пакеты программ, современные профессиональные стандарты информационных технологий	Навыками применения в профессиональной деятельности библиотек и пакетов программ, современных профессиональных стандартов информационных технологий

Профессиональные компетенции (ПК):

ПК-3 - Способен использовать и развивать методы научных исследований и инструментарий в области управления проектами в области ИТ

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
	ПК-3.1: Знать:	ПК-3.2: Уметь:	ПК-3.3: Владеть (иметь навыки):
	методы научных исследований и инструментарий в области управления проектами в области ИТ	использовать и развивать методы научных исследований и инструментарий в области управления проектами в области ИТ	навыками использования и применения методов научных исследований и инструментария в области управления проектами в области ИТ
Пороговый	Современные интеллектуальные технологии для решения профессиональных задач	Решать задачи профессиональной деятельности с помощью современных интеллектуальных технологий	Приемами решения профессиональных задач на основе интеллектуальных технологий
Стандартный (в дополнение к пороговому)	методы выбора современных интеллектуальных технологий и программной среды	принимать решение по обоснованию выбора интеллектуальных технологий и программной среды при разработке оригинальных программных средств	приемами обоснования выбора интеллектуальных средств для разработки оригинальных программных средств

Повышенны й (в дополнение к пороговому, стандартном у)	необходимые языки программирования и библиотеки для решения производственных задач	применять необходимые языки программирования и библиотеки для решения производственных задач	языками программирования и библиотеками для решения производственных задач
--	--	--	--

ПК-1 - Способен разрабатывать план конфигурационного управления

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
	ПК-1.1: Знать:	ПК-1.2: Уметь:	ПК-1.3: Владеть (иметь навыки):
	системы контроля версий и поддержки конфигурационного управления	разрабатывать план конфигурационного управления	навыками разработки плана конфигурационного управления
Пороговый	способы сбора и эффективного анализа информации для дальнейшего использования в программировании поставленных задач	собирать и анализировать информацию для дальнейшего использования в программировании поставленных задач	сбора и эффективного анализа информации для дальнейшего использования в программировании поставленных задач
Стандартный (в дополнение к пороговому)	способы модернизации и адаптации программного обеспечения для решения профессиональных задач	модернизировать программное и аппаратное обеспечение интеллектуальных информационных систем	Навыками модернизации современного программного и аппаратного обеспечения на основе современных технологий,
Повышенны й (в дополнение к пороговому, стандартном у)	Современное программное и аппаратное обеспечение интеллектуальных информационных систем	Разрабатывать современное программное и аппаратное обеспечение	Навыками разработки современного программного и аппаратного обеспечения на основе современных технологий

6.3. Паспорт оценочных материалов

№ п/п	Наименование темы (раздела) дисциплины	Контролируемые планируемые	Вид контроля/используем
-------	--	----------------------------	-------------------------

		результаты обучения в соотношении с результатами обучения по программе	ые оценочные средства	
			Текущий	Промежуточный
1.	Основы программирования на языке Python	ОПК-1.1, ОПК-1.2, ОПК-1.3, ОПК-2.1, ОПК-2.2, ОПК-2.3, ОПК-4.1, ОПК-4.2, ОПК-4.3, ОПК-5.1, ОПК-5.2, ОПК-5.3, ОПК-8.1, ОПК-8.2, ОПК-8.3, ПК-3.1, ПК- 3.2, ПК-3.3, ПК-1.1, ПК-1.2, ПК-1.3	Оценка контрольных работ (для заочной формы обучения) Тестирование	Экзамен
2.	Структуры данных. Объектно-ориентированный подход	ОПК-1.1, ОПК-1.2, ОПК-1.3, ОПК-2.1, ОПК-2.2, ОПК-2.3, ОПК-4.1, ОПК-4.2, ОПК-4.3, ОПК-5.1, ОПК-5.2, ОПК-5.3, ОПК-8.1, ОПК-8.2, ОПК-8.3, ПК-3.1, ПК- 3.2, ПК-3.3, ПК-1.1, ПК-1.2, ПК-1.3	Оценка контрольных работ (для заочной формы обучения) Тестирование	Экзамен

6.4.Оценочные материалы для текущего контроля

Задания для тестирования по дисциплине для оценки сформированности компетенций (min 20, max 50 + ссылку на ЭИОС с тестами) <https://lms2.sseu.ru/course/index.php?categoryid=1863>

Языком программирования называется...

совокупность средств и правил перевода текста с естественного языка на формальный

совокупность средств и правил перевода текста с формального языка на естественный

совокупность средств и правил представления алгоритма в виде, пригодном для

выполнения вычислительной машиной

язык, понятный как программисту, так и компьютеру

... преобразует программу на высокоуровневом языке программирования в машинный код

и формирует исполняемый файл приложения.

транслятор

компилятор

интерпретатор

парсер

... выполняет исходный код на языке высокого уровня «на лету» непосредственно во время работы программы, без предварительного формирования исполняемого файла.

транслятор

компилятор

интерпретатор

парсер

Константами называются элементы данных,

... не имеющие значений

значения которых в процессе выполнения программы могут изменяться или не изменяться

в зависимости от условия

значения которых в процессе выполнения программы изменяются

значения которых в процессе выполнения программы не изменяются

Программные инструментальные средства разработки ПО – это:

программы, позволяющие выполнить все работы, определенные методологией проектирования ПО

системное программное обеспечение, позволяющее сопровождать офисные программные пакеты средства создания текстовых документов

программное обеспечение, используемое на всех стадиях разработки нового ПО

программное обеспечение для настройки офисных приложений на условия конкретного применения программы, которые используются в ходе разработки, корректировки или развития других прикладных или системных программ

устройство компьютера, специально предназначенное для поддержки разработки программных средств

средства создания и редактирования текстовых документов

Транслятор – это:

программа, выполняющая перевод программы с одного языка программирования на другой

комплекс программ мультимедийных технологий

программа, которая выполняет перевод программы с одного языка программирования на машинные коды

программа-переводчик с одного иностранного языка на другой

техническое устройство передачи и преобразования аудио и видеосигналов

техническое устройство для кодирования и декодирования информации

программное обеспечение для обеспечения защиты информации на компьютере

одно из основных средств автоматизации программирования для преобразования программы, написанной на машинно-независимом языке, в программу на машинном языке конкретной ЭВМ

Компилятор – это:

один из видов трансляторов

прикладное программное обеспечение

специальная утилита системного ПО

операционная оболочка

переводит в коды сразу всю программу и создает независимый исполняемый файл

программное обеспечение, используемое в издательских системах

программа, которая переводит программу, написанную на языке программирования высокого уровня в программу на машинном языке не участвуя в ее исполнении

переводит в машинные коды 1 строчку программы и сразу ее выполняет

Интерпретатор:

программа для создания и редактирования электронных таблиц

программа, анализирующая команды или операторы исходной программы и немедленно выполняющая их

переводит в коды сразу всю программу и создает независимый исполняемый файл

переводит в машинные коды 1 строчку программы и сразу ее выполняет

программа для создания и редактирования текстовых документов

один из видов трансляторов

программа создания и управления базами данных

программа создания файлов мультимедиа

Компоновщик – это:

программа для компоновки и оформления тестовых документов

редактор связей

комплекс программ, для создания и ведения баз данных

программа, которая из одного или нескольких объектных модулей с привлечением библиотечных программ и стандартных подпрограмм формирует загрузочный модуль

программное обеспечение для создания презентаций

программа сборки загрузочного модуля из полученных в результате отдельной компиляции объектных модулей с автоматическим поиском и присоединением библиотечных подпрограмм и процедур

программа для поиска синтаксических и семантических ошибок в программе программа
Отладчик - это:

программа, облегчающая программисту выполнение отладки разрабатываемых им программ

программа для создания системы защиты файла

программа создания системы защиты от вирусных атак

программа, помогающая анализировать поведение отлаживаемой программы, обеспечивая ее трассировку

операционная оболочка для создания и управления файловыми структурами

системное программное обеспечение для настройки операционной системы

программа создания и редактирования графических файлов

программа, позволяющая выполнять остановы в заданных точках, просмотреть текущие значения переменных и изменять их значения

Укажите низкоуровневый язык программирования

Assembler

C

C++

Pascal

Укажите среды программирования, поддерживающие разработку и выполнение программ на языке Python.

Embarcadero RAD Studio

MS Visual Studio

PyCharm

IntelliJ IDEA

Jupyter Notebook

Notepad++

Язык Python является языком...

низкого уровня

высокого уровня

сверхвысокого уровня

Язык Python является...

языком со строгой типизацией данных

языком с динамической типизацией данных

Язык Python является...

интерпретируемым

компилируемым

Язык Python реализует ... парадигму программирования.

объектно-ориентированную

функциональную

логическую

Язык Python является...

кроссплатформенным

исполняемым под ОС Windows

исполняемым под ОС Linux

браузерным

IEEE – это:

коммерческая организация ученых и исследователей

просто принятое обозначение, расшифровки не имеет

обозначение всемирной компьютерной сети

всемирная некоммерческая техническая профессиональная ассоциация ученых и исследователей

такая аббревиатура нигде не используется

institute Of Electrical and Electronic Engineers, Inc

американская организация ученых-экономистов

институт инженеров радиоэлектроники и электротехники

Первый этап развития технологий программирования

объектный подход к программированию

компонентный подход и CASE-технологии
«стихийное» программирование
структурный подход к программированию
Второй этап развития технологий программирования
объектный подход к программированию
компонентный подход и CASE-технологии
«стихийное» программирование
структурный подход к программированию
Третий этап развития технологий программирования
объектный подход к программированию
компонентный подход и CASE-технологии
«стихийное» программирование
структурный подход к программированию
Четвертый этап развития технологий программирования
объектный подход к программированию
компонентный подход и CASE-технологии
«стихийное» программирование
структурный подход к программированию
Сцепление модулей по данным предполагает
что модули обмениваются данными, представленными скалярными значениями
что модули обмениваются данными, объединенными в структуры
что один модуль посылает другому некоторый информационный объект (флаг),
предназначенный для управления внутренней логикой модуля
что модули работают с общей областью данных
Сцепление модулей по образцу предполагает
что модули обмениваются данными, представленными скалярными значениями
что модули обмениваются данными, объединенными в структуры
что один модуль посылает другому некоторый информационный объект (флаг),
предназначенный для управления внутренней логикой модуля
что модули работают с общей областью данных
При сцеплении модулей по управлению предполагается
что модули обмениваются данными, представленными скалярными значениями
что модули обмениваются данными, объединенными в структуры
что один модуль посылает другому некоторый информационный объект (флаг),
предназначенный для управления внутренней логикой модуля
что модули работают с общей областью данных

Тематика контрольных работ

Раздел дисциплины	Темы																				
Основы программирования на языке Python	Откройте рейтинг языков программирования от IEEE .																				
	Укажите, на основе чего строится данный индекс популярности и заполните следующую таблицу:																				
	<table border="1"> <thead> <tr> <th></th> <th colspan="4">Тип ПО</th> </tr> <tr> <th>Место в рейтинге</th> <th>Web</th> <th>Mobile</th> <th>Enterprise</th> <th>Embedded</th> </tr> </thead> <tbody> <tr> <td>1</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>2</td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>		Тип ПО				Место в рейтинге	Web	Mobile	Enterprise	Embedded	1					2				
	Тип ПО																				
Место в рейтинге	Web	Mobile	Enterprise	Embedded																	
1																					
2																					

3

4

5

По результатам заполнения таблицы ответьте на следующие вопросы:

- какой язык программирования является наиболее универсальным (охватывает больше типов ПО)?
- какой язык программирования наименее универсален?

На сайте TIOBE имеется собственный [рейтинг популярности языков](#).

Укажите, на основе чего строится данный индекс популярности, после чего возьмите первые 15 языков программирования и заполните следующую таблицу, используя информацию из сети Интернет:

№	Язык	Тип транслятора
1	Java	Гибрид
.	.	.

После заполнения:

- определите количество языков программирования с выбранным типом транслятора (например, компилируемых - 6, гибридов - 2 и т.д.);
- ответьте, чем может объясняться такое соотношение.

Для запуска интерпретатора Python откройте терминал и введите команду `python` (или `python3`), после чего интерпретатор будет запущен и перейдет в интерактивный режим, ожидая ввода команд (Рисунок 2.2.12).

```
C:\WINDOWS\system32\cmd.exe - python
Microsoft Windows [Version 10.0.16299.192]
(c) Корпорация Майкрософт (Microsoft Corporation), 2017. Все права защищены.

C:\Users\Yuri>python
Python 3.5.3 (v3.5.3:1880cb95a742, Jan 16 2017, 16:02:32) [MSC v.1900 64-bit
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Рисунок 2.2.12 - Запуск интерпретатора Python

После ввода команды ее необходимо подтвердить нажатием клавиши <ENTER>. При этом, если команда подразумевает возврат какого-либо результата, интерпретатор автоматически отобразит его в консоли (Рисунок 2.2.13).

Рисунок 2.2.13 - Ввод команд и отображение результата

Введите в консоли по очереди команды, приведенные в Листинге 2.2.1. Комментарии вводить не обязательно. Один из возможных результатов выполнения приведен на Рисунке 2.2.14.

Листинг 2.2.1 - Пример программы на языке Python | [скачать](#)

```
print("Решение кв. уравнения |  $ax^2 + bx + c = 0$ ")

a = int(input("a=")) # Вводим с клавиатуры целое число и
b = int(input("b=")) # связываем с определенной переменной
c = int(input("c="))

d = b**2 - 4*a*c # Рассчитываем дискриминант, [d] указывает на результат
d # Отображаем полученное значение (только в интерактивной консоли!)

# Выводим решение в зависимости от значения дискриминанта
if d < 0:
    print("Решений нет")
elif d == 0:
    x = -b / (2*a)
    print("x =", x)
else:
    x1 = (-b - d**0.5) / (2*a)
    x2 = (-b + d**0.5) / (2*a)
    print("x1 =", x1, " x2 =", x2)
```

```
C:\WINDOWS\system32\cmd.exe - python
>>> print("Решение кв. уравнения |  $ax^2 + bx + c = 0$ ")
Решение кв. уравнения |  $ax^2 + bx + c = 0$ 
>>> a = int(input("a=")) # Вводим с клавиатуры целое число и
a=5
>>> b = int(input("b=")) # связываем с определенной переменной
b=8
>>> c = int(input("c="))
c=1
>>> d = b**2 - 4*a*c # Рассчитываем дискриминант, [d] указывает на рез
>>> d # Отображаем полученное значение (только в интерактивной консоли
44
>>> # Выводим решение в зависимости от значения дискриминанта
...
>>> if d < 0:
...     print("Решений нет")
... elif d == 0:
...     x = -b / (2*a)
...     print("x =", x)
... else:
...     x1 = (-b - d**0.5) / (2*a)
...     x2 = (-b + d**0.5) / (2*a)
...     print("x1 =", x1, " x2 =", x2)
...
x1 = -1.4633249580710799 x2 = -0.13667504192892005
>>>
```

Рисунок 2.2.14 - Результат ввода программы в интерактивном режиме. Даже если не знать синтаксис Python, благодаря его «простоте», можно понять, что происходит в программе.

2.2.2.2. IDE

В случае, когда программу требуется сохранить и периодически изменять, удобнее будет работать в IDE.

2.2.2.2.1. Atom

Для создания и запуска программы необходимо выполнить нескольких шагов:

1. Для создания файла выберите меню *File -> New File (Ctrl + N)*.
2. Сохраните файл с расширением `.py` (например, `main.py`), используя меню *File -> Save (Ctrl + S)*. Убедитесь, что кодировка файла установлена в UTF-8.
3. Наберите текст программы из Листинга 2.2.1.
4. Для запуска программы нажмите клавишу *F5*.

5. Протестируйте работу программы, запустив ее несколько раз, введя различные входные данные.
6. Проверьте, исправьте возможные стилевые ошибки (до исчезновения соответствующего окна сообщений) и сохраните окончательный вариант программы (Рисунок 2.2.15).

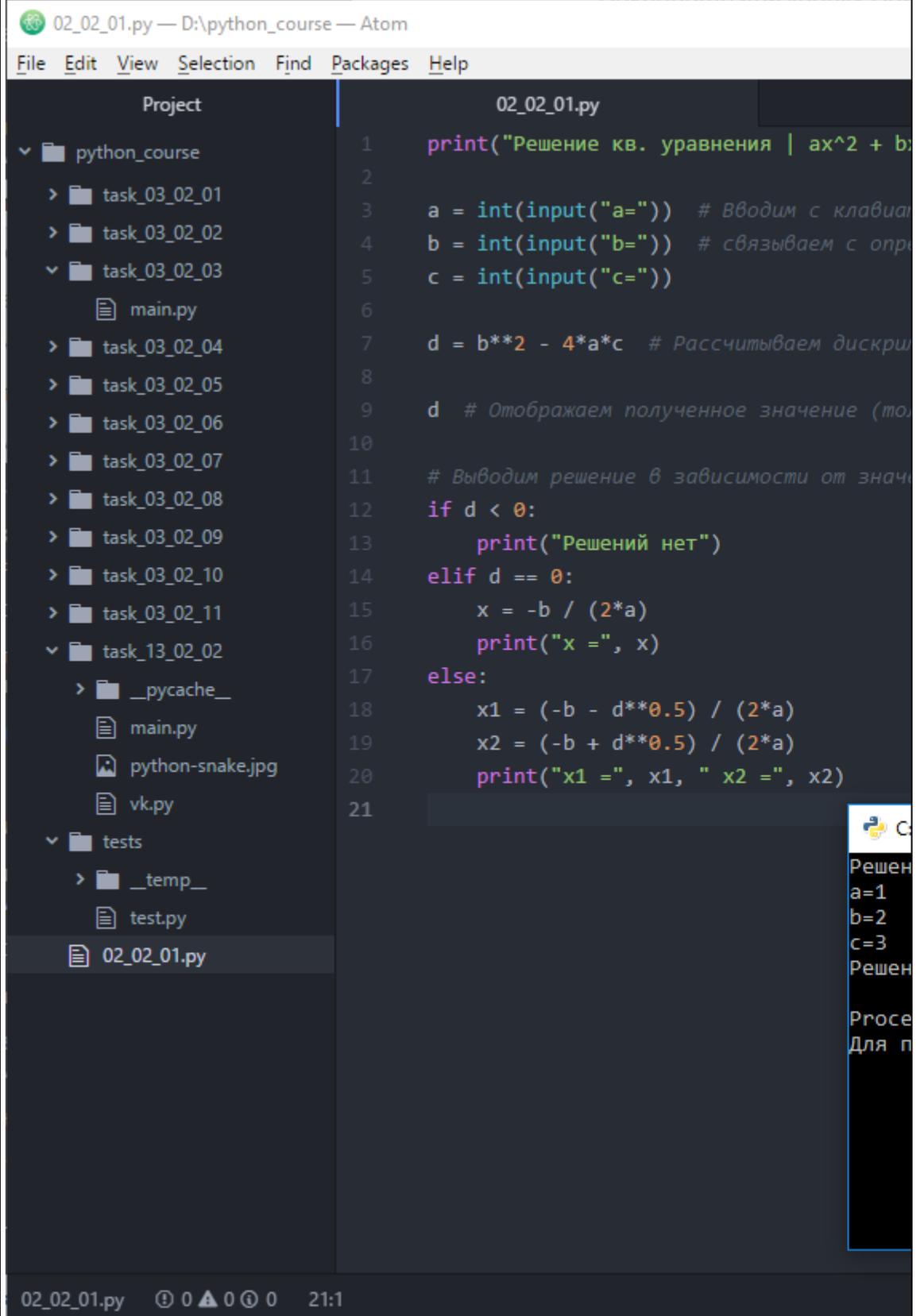


Рисунок 2.2.15 - Работа в IDE Atom

2.2.2.2.2. Geany

Для создания и запуска программы необходимо выполнить нескольких шагов:

1. Для создания файла выберите меню *Файл -> Создать* (*Ctrl + N*).
2. Сохраните файл с расширением `.py` (например, `'main.py'`), используя меню *Файл -> Сохранить* (*Ctrl + S*). Убедитесь, что кодировка файла установлена в UTF-8.
3. Наберите текст программы из Листинга 2.2.1.
4. Для запуска программы нажмите клавишу *F5* или кнопку *Выполнить* на панели инструментов.
5. Протестируйте работу программы, запустив ее несколько раз, введя различные входные данные.
6. Проверьте, исправьте возможные стилевые ошибки, выбрав меню *Сборка -> PEP8* и сохраните окончательный вариант программы (Рисунок 2.2.16).

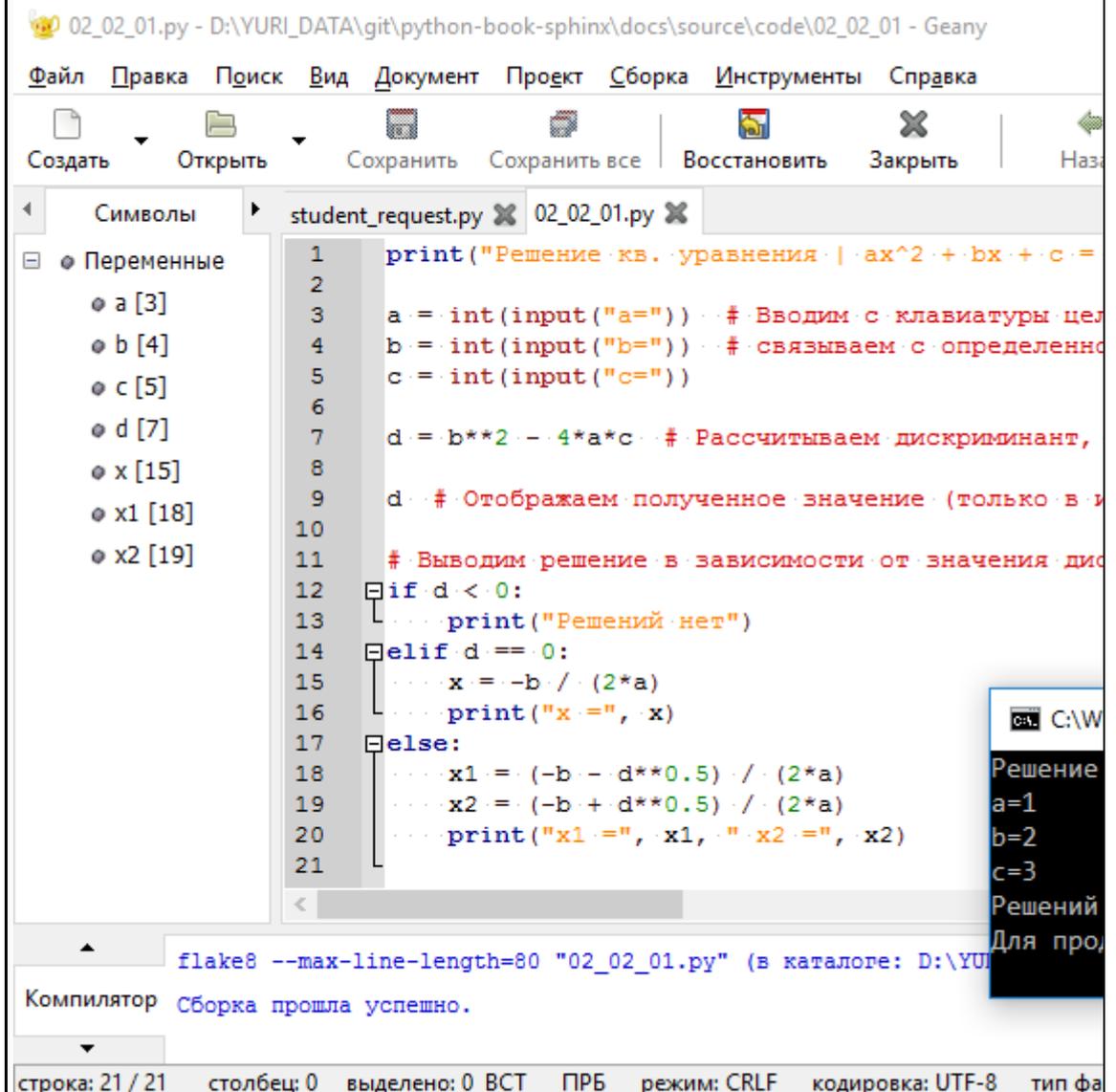


Рисунок 2.2.16 - Работа в IDE Geany

2.2.2.2.3. PyCharm

PyCharm использует концепцию проектов, поэтому в первую очередь необходимо создать пустой проект, и, введя его название, нажать кнопку ОК (Рисунок 2.2.17).

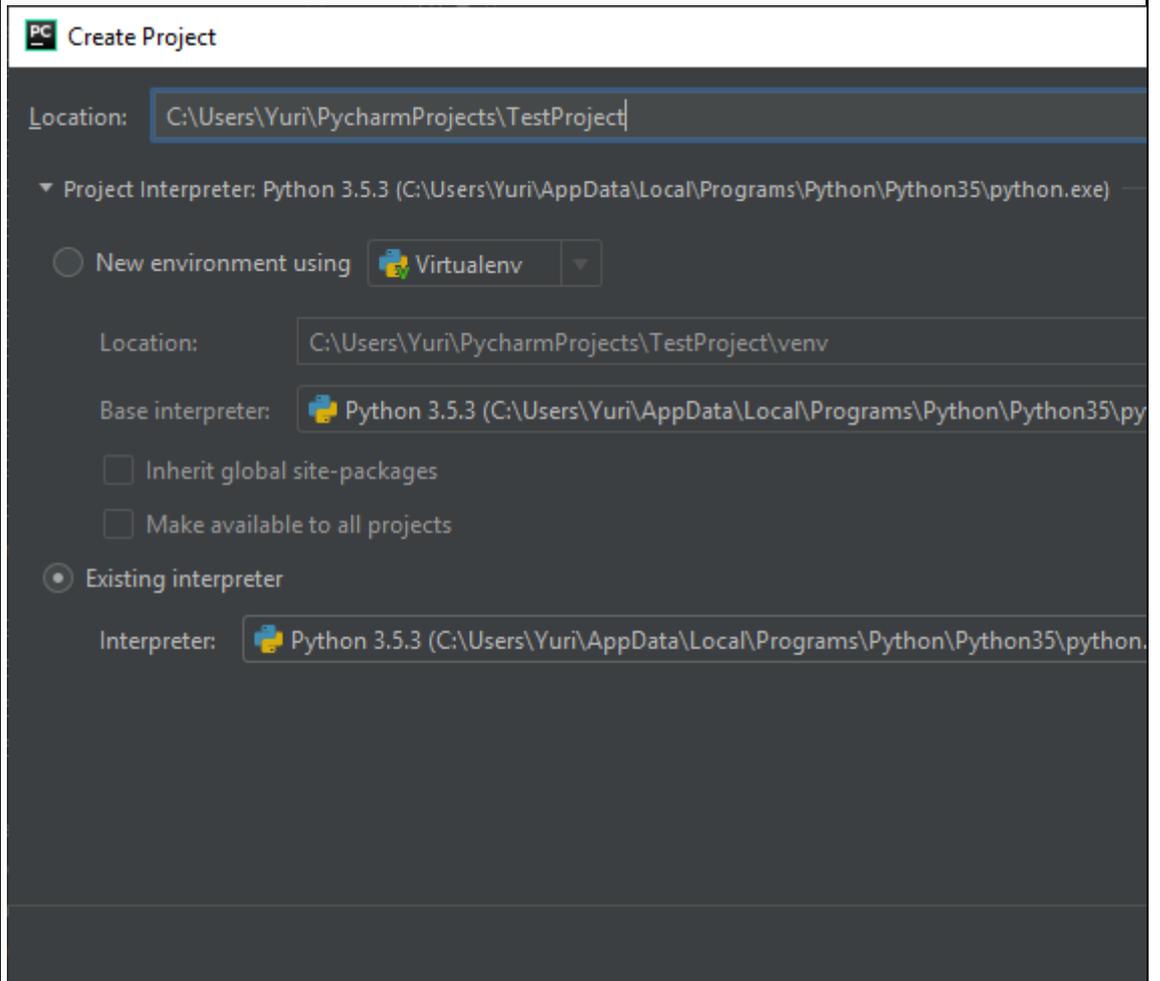


Рисунок 2.2.17 - Создание проекта IDE PyCharm

После открытия проекта необходимо выполнить нескольких шагов:

1. Для создания файла выберите меню *File -> New (Alt + Ins)*.
2. Выберите *Python File* и введите имя файла (например, `'main.py'`), после чего подтвердите создание нажатием кнопки ОК.
3. Наберите текст программы из Листинга 2.2.1.
4. Для запуска программы нажмите клавишу *F5*.
5. Протестируйте работу программы, запустив ее несколько раз, введя различные входные данные.
6. Проверьте, исправьте возможные стилевые ошибки (до исчезновения подчеркиваний в редакторе кода) и сохраните окончательный вариант программы (Рисунок 2.2.18).

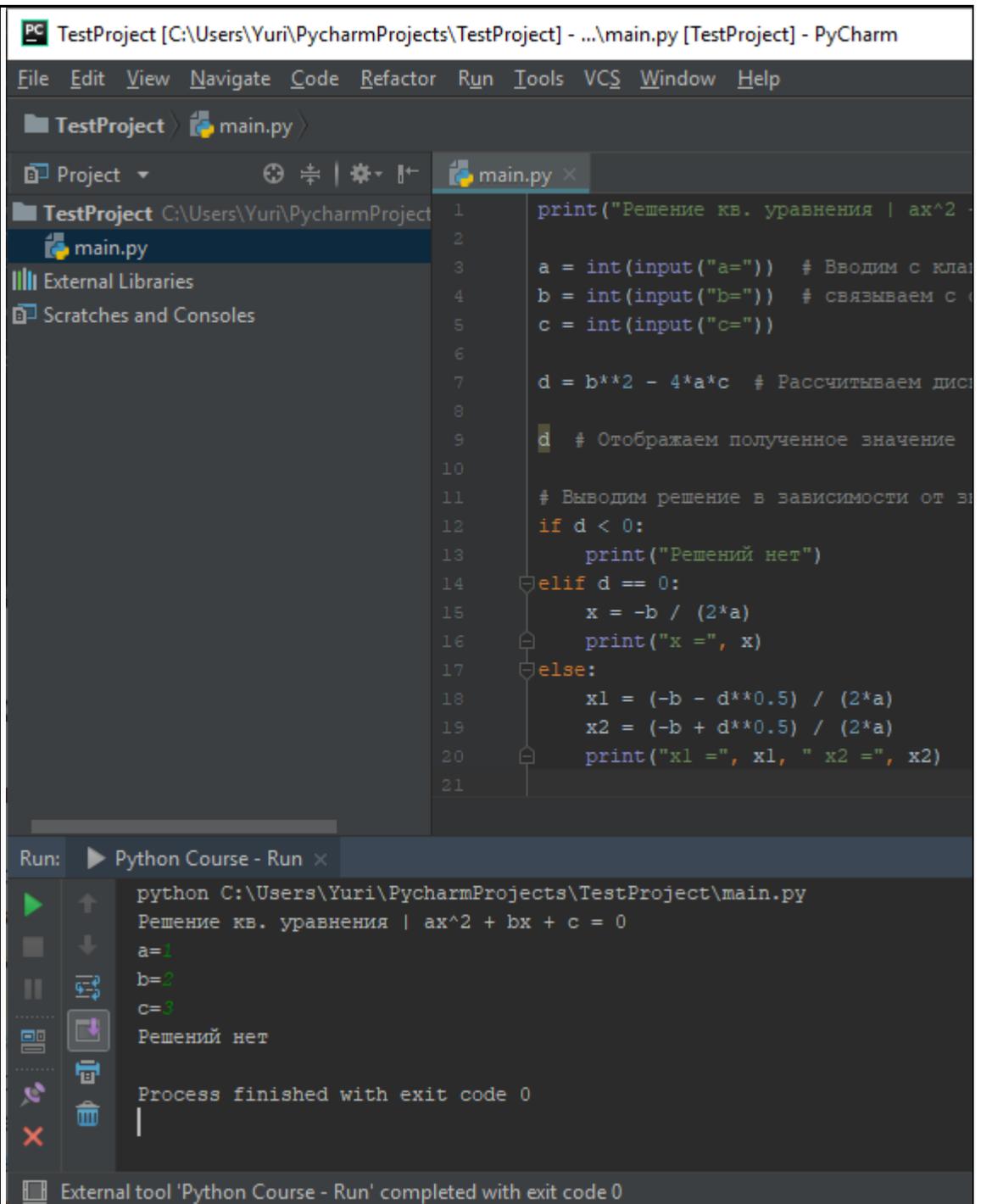


Рисунок 2.2.18 - Работа в IDE PyCharm

2.2.2.2.4. Автоматическая проверка заданий

Начиная с 3-й темы заключительным этапом решения задачи должна быть автоматическая проверка (клавиша *F6* в Atom и PyCharm или меню *Сборка -> Проверить* в Geany). Подробнее см. раздел [Выполнение практикума и самостоятельной работы](#).

2.2.3. Получение справочной информации

Исчерпывающая документация по языку программирования Python

находится на [официальном сайте в разделе документации](#) или в папке `Doc` (куда был установлен Python). Воспользуйтесь поиском, когда необходимо найти описание неизвестной функции или ознакомиться с ее поведением (Рисунок 2.2.19).

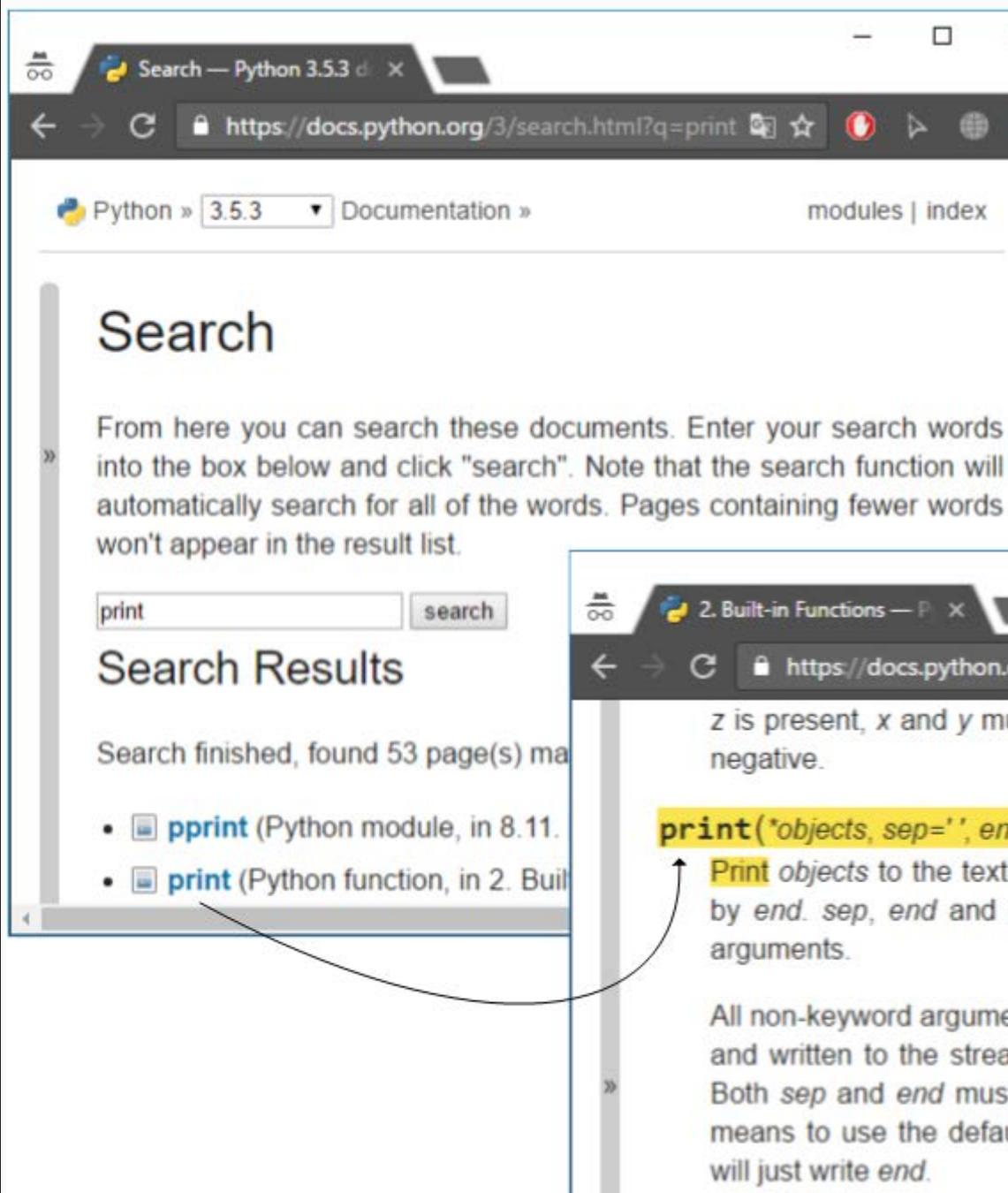


Рисунок 2.2.19 - Поиск функции `print()` в онлайн-документации

help

Еще одним (и возможно более простым) способом получения справочной информации является использование функции `help()` (Рисунок 2.2.20)

```
C:\WINDOWS\system32\cmd.exe - python
(c) Корпорация Майкрософт (Microsoft Corporation), 2017. Все права защищены.

C:\Users\Yuri>python
Python 3.5.3 (v3.5.3:1880cb95a742, Jan 16 2017, 16:02:32) [MSC v.1900 64-bit
(MD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> help(print)
Help on built-in function print in module builtins:

print(...)
    print(value, ..., sep=' ', end='\n', file=sys.stdout, flush=False)

    Prints the values to a stream, or to sys.stdout by default.
    Optional keyword arguments:
    file: a file-like object (stream); defaults to the current sys.stdout.
    sep: string inserted between values, default a space.
    end: string appended after the last value, default a newline.
    flush: whether to forcibly flush the stream.

>>> _
```

Рисунок 2.2.20 - Использование функции `help()` для получения справки
примечание

Используя заготовку, сохраните как:

- `desktop_python_terminal.png`: скриншот рабочего стола с запущенным интерпретатором Python с указанием версии и набранной программой из Листинга 2.2.1;
- `desktop_ide_terminal.png`: скриншот рабочего стола с запущенной IDE (с набранной программой из Листинга 2.2.1), а также терминала с результатом запуска;
- `task_02_02_01.py`: файл с текстом программы из Листинга 2.2.1;
- `help.png`: скриншот с найденной справочной информацией для функции `input()` (используя используя функцию `help()` и официальный сайт с документацией).

Структуры данных. Объектно-ориентированный подход

10.2.1. Римское число

Создайте класс `Roman` (`РимскоеЧисло`), представляющий римское число и поддерживающий операции `+`, `-`, `*`, `/`.

ОТВЕТ

При реализации класса следуйте рекомендациям:

- операции `+`, `-`, `*`, `/` реализуйте как специальные методы

(`__add__` и др.);

- методы преобразования имеет смысл реализовать как статические методы, позволяя не создавать экземпляр объекта в случае, если необходимо выполнить только преобразования чисел.

10.2.2. Пиццерия

Пиццерия предлагает клиентам три вида пиццы: `Пепперони`, `Барбекю` и `Дары Моря`, каждая из которых определяется тестом, соусом и начинкой.

Требуется спроектировать и реализовать приложение для терминала, позволяющее обеспечить обслуживание посетителей.

Дополнительная информация

В бизнес-процессе работы пиццерии в контексте задачи можно выделить 3 сущности (объекта):

- `Терминал`: отвечает за взаимодействие с пользователем:
 - вывод меню на экран;
 - прием команд от пользователя (выбор пиццы, подтверждение заказа, оплата и др.);
- `Заказ`: содержит список заказанных пицц, умеет подсчитывать свою стоимость;
- `Пицца`: содержит заявленные характеристики пиццы, а также умеет себя приготовить (замесить тесто, собрать ингредиенты и т.д.), испечь, порезать и упаковать.

Т.к. пиццерия реализует несколько видов пиццы, которые различаются характеристиками, логично будет сделать общий класс `Пицца`, а в дочерних классах (например, классе `ПиццаБарбекю`) уточнить характеристики конкретной пиццы.

Диаграмма указанных классов в нотации UML приведена на Рисунке 10.2.1.

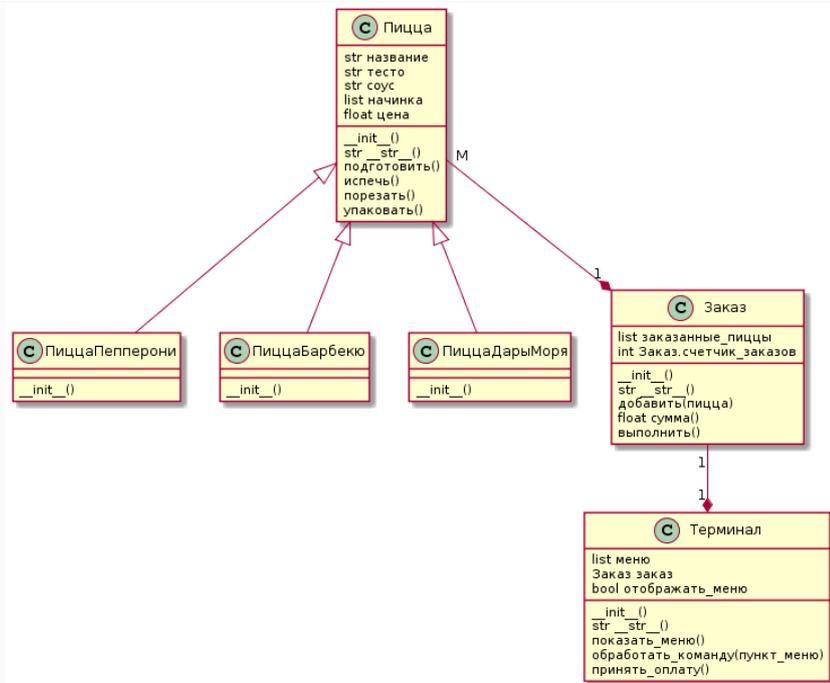


Рисунок 10.1.1 - UML-диаграмма классов приложения

Алгоритм работы пользователя с терминалом может выглядеть следующим образом:

1. Терминал отображает список меню.
2. Терминал создает новый заказ.
3. Клиент вводит номер пиццы из меню.
4. Заказ добавляет в список выбранную пиццу.
5. Действия 3-4 повторяются до подтверждения или отмены.
6. Клиент подтверждает заказ (или отменяет).
7. Терминал выставляет счет, отображая информацию о заказе.
8. Терминал принимает оплату.
9. Заказ отдается на выполнение.

10.2.3. Банковские вклады

Банк предлагает ряд вкладов для физических лиц:

- Срочный вклад: расчет прибыли осуществляется по формуле простых процентов;
- Бонусный вклад: бонус начисляется в конце периода как % от прибыли, если вклад больше определенной суммы;
- Вклад с капитализацией процентов.

Реализуйте приложение, которое бы позволило подобрать клиенту вклад по заданным параметрам.

6.5. Оценочные материалы для промежуточной аттестации

Фонд вопросов для проведения промежуточного контроля в форме экзамена

Раздел дисциплины	Вопросы
Основы программирования на языке Python	<ol style="list-style-type: none"> 1. Понятия программы, языка и среды программирования. 2. Виды языков программирования. 3. Парадигмы программирования. 4. Сравнение наиболее популярных языков программирования. 5. Правила оформления кода. Code Style. PEP 8. 6. Python. Переменные. Правила именования. Оператор присваивания. 7. Python. Типы данных. Динамическая типизация. 8. Python. Вычисления. Арифметические операторы. 9. Python. Условный оператор if, операторы сравнения, логические операторы. 10. Python. Цикл for. 11. Python. Цикл while. 12. Python. Итераторы и генераторы. 13. Python. Строки. Срезы. 14. Python. Структуры данных: списки. 15. Python. Структуры данных: множества. 16. Python. Структуры данных: словари. 17. Python. Структуры данных: словари.
Структуры данных. Объектно-ориентированный подход	<ol style="list-style-type: none"> 18. Python. Объявление и вызов функций. 19. Python. Рекурсивный вызов функций. 20. Python. Пространства имен и области видимости. 21. Python. Объекты. Классы. 22. Python. Наследование классов. 23. Python. Модули и импорт. 24. Python. Файловый ввод-вывод. 25. Jupyter Notebook для подготовки исследовательских отчетов. 26. Библиотека для анализа данных NumPy. 27. Python. Поиск и подключение научно-исследовательских пакетов 28. Организация отладки и тестирования программных продуктов. 29. Методики оценки качества программного обеспечения. 30. Организация документирования программных средств.

6.6. Шкалы и критерии оценивания по формам текущего контроля и промежуточной аттестации

Шкала и критерии оценивания

Оценка	Критерии оценивания для мероприятий контроля с применением 4-х балльной системы
«отлично»	Повышенный ОПК-1.1, ОПК-1.2, ОПК-1.3, ОПК-2.1, ОПК-2.2, ОПК-2.3, ОПК-4.1, ОПК-4.2, ОПК-4.3, ОПК-5.1, ОПК-5.2, ОПК-5.3, ОПК-8.1, ОПК-8.2, ОПК-8.3, ПК-3.1, ПК-3.2, ПК-3.3, ПК-1.1, ПК-1.2, ПК-1.3
«хорошо»	Стандартный ОПК-1.1, ОПК-1.2, ОПК-1.3, ОПК-2.1, ОПК-2.2, ОПК-2.3, ОПК-4.1, ОПК-4.2, ОПК-4.3, ОПК-5.1, ОПК-5.2, ОПК-5.3, ОПК-8.1, ОПК-8.2, ОПК-8.3, ПК-3.1, ПК-3.2, ПК-3.3, ПК-1.1, ПК-1.2, ПК-1.3
«удовлетворительно»	Пороговый ОПК-1.1, ОПК-1.2, ОПК-1.3,

	ОПК-2.1, ОПК-2.2, ОПК- 2.3, ОПК-4.1, ОПК-4.2, ОПК-4.3, ОПК-5.1, ОПК-5.2, ОПК-5.3, ОПК-8.1, ОПК-8.2, ОПК-8.3, ПК-3.1, ПК-3.2, ПК-3.3, ПК-1.1, ПК- 1.2, ПК-1.3
«неудовлетворительно»	Результаты обучения не сформированы на пороговом уровне