

Документ подписан простой электронной подписью.
Информация о владельце:

ФИО: Кандрашина Елена Александровна

Должность: Врио ректора ФГАОУ ВО «Самарский государственный экономический университет»

Дата подписания: 07.07.2023 14:45:32

Уникальный программный ключ:

b2fd765521f4c570b8c6e8e502a10b4f1de8ae0d

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 № 11 от 30 мая 2023 г.)

РАБОЧАЯ ПРОГРАММА

Наименование дисциплины Б1.О.26 Алгоритмизация и программирование

Основная профессиональная образовательная программа 09.03.03 Прикладная информатика программа
Интеллектуальные цифровые системы и сервисы
в управлении

Квалификация (степень) выпускника Бакалавр

Самара 2023

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 Алгоритмизация и программирование входит в обязательную часть блока Б1. Дисциплины (модули)

Последующие дисциплины по связям компетенций: Облачные технологии и сервисы, Основы финансового и экономического анализа, Облачные ресурсы для разработки интеллектуальных сервисов, Программирование, Проектирование баз данных, Теория вероятности и математическая статистика, Дискретная математика, Имитационное моделирование

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

Изучение дисциплины Алгоритмизация и программирование в образовательной программе направлено на формирование у обучающихся следующих компетенций:

Универсальные компетенции (УК):

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

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
УК-1	УК-1.1: Знать:	УК-1.2: Уметь:	УК-1.3: Владеть (иметь навыки):
	основные методы критического анализа и основы системного подхода как общенаучного метода, критерии сопоставления различных вариантов решения поставленной задачи, принципы, критерии, правила построения суждения и оценок	анализировать поставленную задачу, используя основы критического анализа и системного подхода, осуществлять критический анализ собранной информации на соответствие ее условиям и критериям решения поставленной задачи, сопоставлять и оценивать различные варианты решения поставленной задачи, формируя собственные суждения и оценки	методами установления причинно-следственных связей и определения наиболее значимых среди них; методиками постановки цели и определения способов ее достижения; методиками разработки стратегий действий при проблемных ситуациях

Общепрофессиональные компетенции (ОПК):

ОПК-7 - Способен разрабатывать алгоритмы и программы, пригодные для практического применения

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
ОПК-7	ОПК-7.1: Знать:	ОПК-7.2: Уметь:	ОПК-7.3: Владеть (иметь навыки):
	основы алгоритмизации, языки программирования и работы с базами данных	применять языки программирования и работы с базами данных при решении прикладных	навыками использования технологии работы с базами данных, алгоритмизации и программирования для

		задач	создания программных продуктов в прикладных областях деятельности
--	--	-------	---

ОПК-1 - Способен применять естественнонаучные и общетеоретические знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
	ОПК-1.1: Знать:	ОПК-1.2: Уметь:	ОПК-1.3: Владеть (иметь навыки):
ОПК-1	основы математики, вычислительной техники и программирования	решать стандартные профессиональные задачи с применением естественнонаучных и общетеоретических знаний, методов математического анализа и моделирования	навыками теоретического и экспериментального исследования объектов профессиональной деятельности

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 1
Контактная работа, в том числе:	74.3/2.06
Занятия лекционного типа	36/1
Лабораторные работы (лабораторный практикум)	36/1
Индивидуальная контактная работа (ИКР)	0.3/0.01
Групповая контактная работа (ГКР)	2/0.06
Самостоятельная работа:	35.7/0.99
Промежуточная аттестация	34/0.94
Вид промежуточной аттестации:	
Экзамен	Экз
Общая трудоемкость (объем части образовательной программы): Часы	144
Зачетные единицы	4

4. Содержание дисциплины

4.1. Разделы, темы дисциплины и виды занятий:

Тематический план дисциплины Алгоритмизация и программирование 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		
			Лаборат. работы	ГКР			
1.	Парадигмы	12	12	0.15	1	17.85	УК-1.1, УК-1.2, УК

	программирования. Общие принципы разработки программного обеспечения Понятие алгоритма. Типы данных. Структура программы.						-1.3, ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-1.1, ОПК-1.2, ОПК-1.3
2.	Базовые понятия императивного программирования. Основы программирования на языке Python. Массивы, циклы, условные операторы, ввод и вывод данных.	24	24	0.15	1	17.85	УК-1.1, УК-1.2, УК-1.3, ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-1.1, ОПК-1.2, ОПК-1.3
Контроль		34					
Итого		36	36	0.3	2	35.7	

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.	Парадигмы программирования. Общие принципы разработки программного обеспечения Понятие алгоритма. Типы данных. Структура программы.	лекция	Основные конструкции структурной декомпозиции
		лекция	Архитектура программ с глобальными, локальными данными, с модульной и объектно – ориентированной структурой.
		лекция	Описания, базовые структуры и этапы анализа систем.
		лекция	Характеристики модели реализации. Сцепление. Связность. Реализация взаимодействия между приложениями.
2.	Базовые понятия императивного программирования. Основы программирования на языке Python. Массивы, циклы, условные операторы, ввод и вывод данных.	лекция	Введение в программирование на языке Python. Программирование линейных алгоритмов на языке Python.
		лекция	Программирование разветвляющихся и циклических процессов на языке Python. Структуры данных языка Python.
		лекция	Пользовательские функции в языке Python. Модули и пакеты языка Python.
		лекция	Итераторы, контейнеры и перечисления в языке Python. Работа с файлами и каталогами в языке Python

*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.	Парадигмы программирования. Общие принципы разработки программного обеспечения Понятие алгоритма. Типы	лабораторные работы	Программирование линейных алгоритмов на языке Python. Написание арифметических выражений и программная реализация линейных алгоритмов. Введение в программирование на языке Python. Знакомство со средой разработки IDLE.

	данных. Структура программы.		Структура программы. Комментарии. Основные типы данных и операций в языке Python. Основные типы переменных. Преобразование типов переменных.
2.	Базовые понятия императивного программирования. Основы программирования на языке Python. Массивы, циклы, условные операторы, ввод и вывод данных.	лабораторные работы	Программирование разветвляющихся и циклических процессов на языке Python. Разработка программ, включающих ветвление и циклические процессы.
		лабораторные работы	Операции над строками в языке Python. Разработка программ с использованием регулярных выражений.
		лабораторные работы	Структуры данных языка Python. Работа с массивами. Разработка программ, использующих динамические списки.
		лабораторные работы	Пользовательские функции в языке Python. Написание процедур и их использование в программном коде.
		лабораторные работы	Модули и пакеты языка Python. Разработка и тестирование модулей. Создание пакетов.
		лабораторные работы	Работа с файлами и каталогами в языке Python. Создание программ на языке Python, с использованием файлового ввода-вывода

** 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.	Парадигмы программирования. Общие принципы разработки программного обеспечения. Понятие алгоритма. Типы данных. Структура программы.	- тестирование
2.	Базовые понятия императивного программирования. Основы программирования на языке Python. Массивы, циклы, условные операторы, ввод и вывод данных.	- 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. Трофимов, В. В. Алгоритмизация и программирование : учебник для вузов / В. В. Трофимов, Т. А. Павловская ; под редакцией В. В. Трофимова. — Москва : Издательство Юрайт, 2020. — 137 с. — (Высшее образование). — ISBN 978-5-534-07834-3. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/452333>

Дополнительная литература

1. Кудрина, Е. В. Основы алгоритмизации и программирования на языке С# : учебное пособие для вузов / Е. В. Кудрина, М. В. Огнева. — Москва : Издательство Юрайт, 2020. — 322 с. — (Высшее образование). — ISBN 978-5-534-09796-2. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/456182>

2. Подбельский, В. В. Программирование. Базовый курс С# : учебник для вузов / В. В. Подбельский. — Москва : Издательство Юрайт, 2020. — 369 с. — (Высшее образование). — ISBN 978-5-534-10616-9. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/450868>

3. Казанский, А. А. Прикладное программирование на Excel 2019 : учебное пособие для вузов / А. А. Казанский. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2021. — 171 с. — (Высшее образование). — ISBN 978-5-534-12022-6. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/470200>

Литература для самостоятельного изучения

1.

5.2. Перечень лицензионного программного обеспечения

1. Microsoft Windows 10 Education / Microsoft Windows 7 / Windows Vista Business 2. Office 365 ProPlus, Microsoft Office 2019, Microsoft Office 2016 Professional Plus (Word, Excel, Access, PowerPoint, Outlook, OneNote, Publisher) / Microsoft Office 2007 (Word, Excel, Access, PowerPoint)

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

6. Фонд оценочных средств по дисциплине Алгоритмизация и программирование:

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

Универсальные компетенции (УК):

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

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
	УК-1.1: Знать:	УК-1.2: Уметь:	УК-1.3: Владеть (иметь навыки):
	основные методы критического анализа и основы системного подхода как общенаучного метода, критерии сопоставления различных вариантов решения поставленной задачи, принципы, критерии, правила построения суждения и оценок	анализировать поставленную задачу, используя основы критического анализа и системного подхода, осуществлять критический анализ собранной информации на соответствие ее условиям и критериям решения поставленной задачи, сопоставлять и оценивать различные варианты решения поставленной задачи, формируя собственные суждения и оценки	методами установления причинно-следственных связей и определения наиболее значимых среди них; методиками постановки цели и определения способов ее достижения; методиками разработки стратегий действий при проблемных ситуациях
Пороговый	возможности обработки собранной информации для решения профессиональных задач.	систематизировать и интерпретировать полученную информацию для решения профессиональных задач.	приемами решения профессиональных задач на основе результатов, полученных в результате анализа и обработки собранной информации
Стандартный (в дополнение к пороговому)	возможности обработки собранной информации для решения профессиональных задач с использованием прикладного программного обеспечения.	систематизировать и интерпретировать полученную информацию для решения профессиональных задач с использованием прикладного программного обеспечения.	приемами решения профессиональных задач на основе результатов, полученных в результате анализа и обработки собранной информации с использованием прикладного программного обеспечения
Повышенный (в дополнение к пороговому, стандартному)	возможности обработки собранной информации для решения профессиональных задач с использованием цифровых технологий.	систематизировать и интерпретировать полученную информацию для решения профессиональных задач с использованием цифровых технологий.	приемами решения профессиональных задач на основе результатов, полученных в результате анализа и обработки собранной информации с использованием цифровых технологий

Общепрофессиональные компетенции (ОПК):

ОПК-7 - Способен разрабатывать алгоритмы и программы, пригодные для практического применения

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
	ОПК-7.1: Знать:	ОПК-7.2: Уметь:	ОПК-7.3: Владеть (иметь навыки):
	основы алгоритмизации, языки программирования	применять языки программирования и	навыками использования технологии работы с

	и работы с базами данных	работы с базами данных при решении прикладных задач	базами данных, алгоритмизации и программирования для создания программных продуктов в прикладных областях деятельности
Пороговый	основные языки	применять языки	навыками программирования на
Стандартный (в дополнение к пороговому)	современные программные среды разработки информационных систем.	применять языки программирования, реализующие RAD – технологию.	навыками программирования на языках, реализующие RAD – технологию, методами функционального тестирования прототипов программно-технических комплексов задач
Повышенный (в дополнение к пороговому, стандартному)	современные программные среды разработки информационных технологий.	применять языки программирования, реализующие принципы объектно-ориентированного программирования.	навыками программирования на языках, реализующие принципы объектно-ориентированного программирования

ОПК-1 - Способен применять естественнонаучные и общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
	ОПК-1.1: Знать: основы математики, вычислительной техники и программирования	ОПК-1.2: Уметь: решать стандартные профессиональные задачи с применением естественнонаучных и общеинженерных знаний, методов математического анализа и моделирования	ОПК-1.3: Владеть (иметь навыки): навыками теоретического и экспериментального исследования объектов профессиональной деятельности
Пороговый	современные информационные технологии	применять в практической деятельности имеющиеся современные информационные технологии	практическими навыками применения современных информационных технологий
Стандартный (в дополнение к пороговому)	современные программные средства в профессиональной деятельности.	применять в практической деятельности имеющиеся современные программные средства	практическими навыками применения современных программных средств.
Повышенный (в дополнение к пороговому, стандартному)	основные принципы применения современных информационных технологий и программных средств и в профессиональной деятельности.	Настраивать и сопровождать современные программные средства	практическими навыками сопровождения и настройки современных программных средств.

6.3. Паспорт оценочных материалов

№	Наименование темы	Контролируемые	Вид контроля/используемые
---	-------------------	----------------	---------------------------

п/п	(раздела) дисциплины	планируемые результаты обучения в соотношении с результатами обучения по программе	оценочные средства	
			Текущий	Промежуточный
1.	Парадигмы программирования. Общие принципы разработки программного обеспечения. Понятие алгоритма. Типы данных. Структура программы.	УК-1.1, УК-1.2, УК- 1.3, ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-1.1, ОПК-1.2, ОПК-1.3	Оценка практических работ Тестирование	Экзамен
2.	Базовые понятия императивного программирования. Основы программирования на языке Python. Массивы, циклы, условные операторы, ввод и вывод данных.	УК-1.1, УК-1.2, УК- 1.3, ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-1.1, ОПК-1.2, ОПК-1.3	Оценка практических работ 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=2028>

Какому оператору соответствует запись $a:=5+b$;
 присваивания
 сравнения
 вычисления
 расчета

Отличительной чертой структурного программирования является:

полный перебор возможных решений задач
 обязательное наличие распределенной БД
 декомпозиция задачи на подзадачи
 использование статистической обработки данных

Программирование сверху вниз – это:

метод сведения трудной задачи к последовательности более простых
 исследование древовидной модели пространства решений и ориентация на поиск оптимального решения
 процесс, при котором от начального предложения осуществляется движение по направлению к лучшим решениям
 процесс пошагового разбиения алгоритма на все более мелкие части с целью получения таких элементов, для которых можно написать конкретные команды

Какой тип операторов используется для программирования повторяющихся действий:

присваивания
 ветвления
 цикла
 ввода

Проектирование программ путем последовательного разбиения большой задачи на меньшие подзадачи соответствует:

объектно – ориентированному проектированию
концептуальному моделированию
восходящему («снизу – вверх») проектированию
нисходящему («сверху – вниз») проектированию

Программа – это:

законченное минимальное смысловое выражение на языке программирования
алгоритм, записанный на языке программирования
протокол взаимодействия компонентов компьютерной сети
набор команд операционной системы компьютера

Разложение целого на части (структурное разбиение) в процессе моделирования называется:

агрегированием
декомпозицией
структурированием
абстрагированием

Модульная структура программы отражает одну из особенностей какого программирования.

структурного
логического
эвристического
динамического

Чему равен результат вычисления $a \bmod b$, если $a=13$, $b=5$

3

0,6

2

будет ошибка

Одна из основных идей структурного программирования состоит в том, что:

используется инкапсуляция и наследование объектов
повторяющиеся фрагменты программы могут оформляться в виде подпрограммы
структура системы описывается в терминах объектов и связей между ними, а поведение -системы в терминах обмена сообщениями между объектами
при написании программ не используются подпрограммы

Чему равен результат вычисления $a \div b$, если $a=24$, $b=5$

4

0,8

4

будет ошибка

К основополагающим принципам структурного программирования не относится:

достижение абсолютного минимума символов и строк в тексте программы
разбиение проект на модули, каждый из которых имеет один вход и один выход
программирование сверху вниз
логика программы допускает только три основные алгоритмические конструкции: следование, ветвление и повторение

При разработке программного продукта описание последовательности действий, ведущих к решению поставленной задачи, относится к этапу:

выбора метода решения задачи
анализа и формализации описания задачи
кодирования программы
разработки алгоритма

Главная идея структурного программирования состоит в том, что:

при написании программ не используются подпрограммы

основными для написания программ являются три типа операторов: линейный, ветвления, организации цикла
используется инкапсуляция и наследование объектов
структура системы описывается в терминах объектов и связей между ними, а поведение системы – в терминах обмена сообщениями между объектами

К основным алгоритмическим конструкциям не относятся:

замещение
цикл с постусловием
цикл с предусловием
ветвление

Если элементы массива $D[1..5]$ равны соответственно 3, 4, 5, 1, 2, то значение выражения $D[D[5]]-D[D[3]]$ равно:

+1
+2
-1
-3

Элементы массива в памяти компьютера упорядочены по:

возрастанию индексов элементов
возрастанию значений элементов
частотным характеристикам
алфавиту

Тип данных языка программирования характеризуется:

правилами преобразования значений, заданными в описании языка программирования
размером кластера
набором методов обработки данных
множеством допустимых значений и набором допустимых над этими значениями операций

Конкретные значения входят в состав команд языка программирования в виде:

операторов
функций
констант
инструкций

Отметьте из каких элементов состоит алгоритмический язык

выражения
символы
слова
операторы

Тематика контрольных работ

Раздел дисциплины	Темы
Парадигмы программирования. Общие принципы разработки программного обеспечения. Понятие алгоритма. Типы данных. Структура программы.	1. Установить интегрированную среду разработки Python (рекомендуется скачать установщик с официального сайта www.python.org). 2. Ознакомиться с базовой документацией по языку программирования Python. 3. Протестировать среду разработки IDLE в двух режимах: в интерактивном и сценарном посредством разработки приложения, образец которого приведен в разделе «Пример написания простейшей программы в среде разработки IDLE». 4. Изучить основные горячие клавиши для последующей быстрой разработки программ в сценарном режиме. 5. Установить специализированную среду разработки JetBrains PyCharm

	<p>Community Edition.</p> <p>6. В установленной среде разработки создать программу, выводящую на экран адрес расположения заданий для лабораторных работ на сервере и адрес любого электронного ресурса, содержащего программное обеспечение для разработки на языке программирования Python.</p>
<p>Базовые понятия императивного программирования. Основы программирования на языке Python. Массивы, циклы, условные операторы, ввод и вывод данных.</p>	<p>Задание для самостоятельной работы №1</p> <ol style="list-style-type: none"> 1. Вывести на экран все нечетные числа в диапазоне $1 < n \leq 4$. 2. Вывести таблицу умножения на p. 3. Вывести таблицу деления числа a на числа от $-b$ до b включительно. 4. Вывести на экран таблицу Пифагора. 5. Вывести на экран таблицу сложения чисел от 1 до 9. 6. Написать программу, выводящую на экран таблицу, в которой количество строк и столбцов задается пользователем. В каждой ячейке выводится ее порядковый номер. 7. Вывести на экран таблицу, ячейки которой закрашены в разные цвета, а номера цветов отображаются в соответствующих ячейках. Номера цветов изменяются от 555555 до 999999 с шагом 1111 8. Вывести на экран строку разных размеров и цветов. Цвет изменяется от 111111 до 999999 с шагом 111111. 9. Вывести на экран таблицу умножения для шестнадцатеричной системы счисления. 10. Вывести на экран таблицу сложения для шестнадцатеричной системы счисления. 11. Вывести на экран таблицу умножения для восьмеричной системы счисления. 12. Вывести на экран таблицу сложения для восьмеричной системы счисления. 13. Вывести на экран таблицу, текст в ячейках которой закрашен в разные цвета, а номера цветов отображаются в соответствующих ячейках. Номера цветов изменяются от 555555 до 999999 с шагом 1111. 14. Вывести на экран один и тот же текст шестнадцатью случайными цветами. 15. Получить сумму n случайных чисел из диапазона $[-50; 50]$. <p>Задание для самостоятельной работы №2</p> <ol style="list-style-type: none"> 1. Создайте многомерный массив, содержащий названия фильмов, организованных по жанрам: ассоциативный массив, в котором имена полей будут разными жанрами («комедия», «мелодрама», «детектив» и др.), а элементами — названия фильмов. Выведите информацию. 2. Создайте ассоциативный массив, аналогичный телефонному справочнику. Отсортируйте массив по фамилиям абонентов в алфавитном порядке. 3. Создайте ассоциативный многомерный массив, содержащий информацию о пользователях (ФИО, возраст, количество посещений страницы). Выведите всю информацию, начиная с пользователей, у которых количество посещений страницы больше. 4. Создайте массив, содержащий сведения об учениках класса (фамилия, рост, вес, средний балл). Найдите самого высокого ученика и выведите всю информацию о нем. 5. Создайте массив, содержащий сведения о ваших друзьях. Отсортируйте его по фамилиям друзей в алфавитном порядке и выведите всю информацию. 6. Создайте массив, содержащий сведения о ваших друзьях. Отсортируйте его по возрасту друзей и выведите всю информацию. 7. Создайте массив, содержащий сведения о продукции фирмы: номер товара, название, цена. Отсортируйте массив по названиям в

	<p>алфавитном порядке. Среди товаров с одинаковым названием сначала идут более дешевые.</p> <p>8. Создайте массив, содержащий сведения о картинках: местонахождение и имя файла, хранящего картинку, ее размеры, название. Выведите все картинки на экран с полной информацией о них.</p> <p>9. Создайте многомерный массив, содержащий названия музыкальных произведений, организованных по жанрам: ассоциативный массив, в котором имена полей будут разными жанрами («рок», «поп», «джаз» и др.), а элементами — названия песен. Выведите информацию.</p> <p>10. Описать массив расписание (день недели, количество пар в этот день, время начала и конца пары, название предмета, фамилия преподавателя) Вывести полную информацию о занятиях, относящихся к предметной области «Информатика».</p> <p>11. В библиотеке имеются книги, газеты, журналы (название, год выпуска, автора, редактора (для газеты), объем.) Вывести информацию об изданиях, вышедших в заданном году.</p> <p>12. Опишите массив, содержащий информацию о движении электропоездов из вашего города: направление; время отправления электропоездов, время в пути до конечного пункта, стоимость билетов по зонам. Вывести перечень электропоездов, следующих в заданном направлении.</p> <p>13. Описать массив экзаменационная ведомость (предмет, номер группы, номер зачетной книжки, фамилия, имя, отчество студента, его оценки по итогам текущей сессии). Определить отличников, хорошистов, троечников и двоечников.</p> <p>14. Описать массив служащих, включающий имена, фамилии, отчества служащих, даты рождения, полученное образование, домашние адреса, профессии. Определить имена людей с высшим образованием. Выдать данные о служащем, который имеет ту или иную профессию.</p> <p>15. При поступлении в университет лица, получившие оценку «неудовлетворительно» на первом экзамене, ко второму экзамену не допускаются. Считая фамилии абитуриентов и их оценки после первого экзамена исходными данными, составить список абитуриентов, допущенных ко второму экзамену.</p>
--	--

6.5. Оценочные материалы для промежуточной аттестации

Фонд вопросов для проведения промежуточного контроля в форме экзамена

Раздел дисциплины	Вопросы
Парадигмы программирования. Общие принципы разработки программного обеспечения Понятие алгоритма. Типы данных. Структура программы.	<ol style="list-style-type: none"> 1. Операторы ввода и вывода данных. Оператор присваивания. 2. Встроенные функции и методы для работы с числами. 3. Модуль Math. Математические функции. 4. Операторы для работы с последовательностью. 5. Приоритет выполнения операторов. Генерация случайных чисел. Модуль random. 6. Операции сравнения. Операторы условного перехода. 7. Операторы перехода на следующую итерацию и прерывания цикла. 8. Функции range() и enumerate(). 9. Вложенные циклы. 10. Строки и двоичные данные. Создание строки. 11. Специальные символы. Операции над строками. 12. Форматирование строк. Метод format(). 13. Функции и методы для работы со строками и символами. 14. Регулярные выражения: синтаксис, поиск по шаблону. 15. Списки. Создание списка. Операции над списками. 16. Многомерные списки. Перебор элементов списка.

	<p>17. Генераторы списков и выражения-генераторы.</p> <p>18. Функции для работы со списками. Добавление и удаление элементов списка.</p> <p>19. Добавление и удаление элементов списка.</p> <p>20. Поиск элемента в списке и получение сведений о значениях, входящих в список.</p> <p>переворачивание и перемешивание списка.</p>
<p>Базовые понятия императивного программирования. Основы программирования на языке Python. Массивы, циклы, условные операторы, ввод и вывод данных.</p>	<p>21. Выбор элемента списка случайным образом. Сортировка списка.</p> <p>22. Заполнение списка числами. Преобразование списка в строку.</p> <p>23. Кортежи. Операции над кортежами.</p> <p>24. Множества. Операции над множествами.</p> <p>25. Диапазоны. Операции над диапазонами.</p> <p>26. Словари. Создание словаря. Операции над словарями.</p> <p>27. Методы для работы над словарями. Генераторы словарей.</p> <p>28. Определение функции и ее вызов. Необязательные параметры функций и сопоставление по ключам.</p> <p>29. Переменное число параметров в функции.</p> <p>30. Анонимные функции.</p> <p>31. Функции-генераторы.</p> <p>32. Декораторы функций.</p> <p>33. Глобальные и локальные переменные.</p> <p>34. Рекурсивные функции. Вложенные функции.</p> <p>35. Модули. Понятие модуля. Подключение модуля: инструкции import и from.</p> <p>36. Пути поиска модулей. Повторная загрузка модулей.</p> <p>37. Пакеты. Понятие пакета. Работа с пакетами.</p> <p>38. Понятие класса, атрибута и метода.</p> <p>39. Определение класса и создание экземпляра класса. Конструкторы и деструкторы.</p> <p>40. Наследование. Множественное наследование.</p> <p>41. Понятие примесей и их использование.</p> <p>42. Специальные методы классов. Перегрузка операторов.</p> <p>43. Статические методы и методы классов.</p> <p>44. Абстрактные методы.</p> <p>45. Ограничение доступа к идентификаторам внутри класса.</p> <p>46. Свойства классов. Декораторы классов.</p> <p>47. Итератор класса. Понятие итератора класса и его использование.</p> <p>48. Понятие контейнера. Контейнеры-последовательности. Контейнеры-словари.</p> <p>49. Перечисления. Атрибуты и методы перечислений.</p> <p>50. Открытие файла. Методы для работы с файлами.</p> <p>51. Классы и модули используемые для доступа к файлам.</p> <p>52. Права доступа к файлам и каталогам.</p> <p>53. Функции для манипулирования файлами.</p> <p>54. Python. Преобразование пути к файлам и каталогам.</p> <p>55. Перенаправление ввода/вывода.</p> <p>56. Сохранение объектов в файл.</p> <p>57. Функции для работы с каталогами.</p>

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 УК-1.1, УК-1.2, УК-1.3, ОПК-7.1, ОПК-7.2, ОПК-7.3,

	ОПК-1.1, ОПК-1.2, ОПК-1.3
«хорошо»	Стандартный УК-1.1, УК-1.2, УК-1.3, ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-1.1, ОПК-1.2, ОПК-1.3
«удовлетворительно»	Пороговый УК-1.1, УК-1.2, УК-1.3, ОПК-7.1, ОПК-7.2, ОПК-7.3, ОПК-1.1, ОПК-1.2, ОПК-1.3
«неудовлетворительно»	Результаты обучения не сформированы на пороговом уровне