Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Выборнова Любовь Алексеевна
Должность: Ректор
Дата подписания: 30.10.2025 09:42:49
Уникальный программный ключ:
c3b3b9c625f6c113afa2a2c42baff9e05a38b76e

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение высшего образования

«Поволжский государственный университет сервиса» (ФГБОУ ВО «ПВГУС»)

Колледж креативных индустрий и предпринимательства

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ОП.04 «ОСНОВЫ АЛГОРИТМИЗАЦИИ И ПРОГРАММИРОВАНИЯ»

Специальность

09.02.07 «Информационные системы и программирование»

Тольятти 2024

Рабочая программа дисциплины «*Основы алгоритмизации и программирования»* разработана в соответствии с Федеральным государственным образовательным стандартом среднего профессионального образования по специальность 09.02.07 Информационные системы и программирование, утвержденным приказом Министерства образования и науки Российской Федерации от 9 декабря 2016 года № 1547.

Разработчик РПД:

преподаватель	Ляпина К.В.
(должность, ученая степень, ученое звание)	(ФИО)

Рассмотрена на заседании предметно-цикловой комиссии по образовательной программе 09.02.07 Информационные системы и программирование
Протокол от «_22_» _ноября__2024 № _3
Председатель ПЦК _ преподаватель Колледжа креативных индустрий и предпринимательства Ефимов Д.М.

1. ПЕРЕЧЕНЬ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫХ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

1.1. Цельосвоения дисциплины

Целью освоения дисциплины является формирование у обучающихся следующих компетенций:

Код компетенции	Наименование компетенции
ОК 02	Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности
ПК 3.1	Осуществлять ревьюирование программного кода в соответствии с технической документацией
ПК 3.3	Производить исследование созданного программного кода с использованием специализированных программных средств с целью выявления ошибок и отклонения от алгоритма

1.2. Планируемые результаты освоения дисциплины

В результате освоения дисциплины обучающийся должен:

иметь практический опыт:

Оптимизировать программный код с использованием специализированных программных средств. Использовать основные методологии процессов разработки программного обеспечения

уметь:

Использовать современное программное обеспечение; использовать различные цифровые средства для решения профессиональных задач.

Выполнять оптимизацию программного кода с использованием специализированных программных средств. Использовать методы и технологии тестирования и ревьюирования кода и проектной документации.

знать:

Номенклатуру информационных источников, применяемых в профессиональной деятельности; приемы структурирования информации; формат оформления результатов поиска информации, современные средства и устройства информатизации; порядок их применения и программное обеспечение в профессиональной деятельности в том числе с использованием цифровых средств.

1.3. Место дисциплиныв структуре образовательной программы

Дисциплина «Основы алгоритмизации и программирования» относится к циклу общеопрофессиональных дисциплин основной профессиональной образовательной программы.

2. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

2.1. Объём учебной дисциплиныи виды учебной работы

Общая трудоёмкость дисциплины составляет **180 час.** Их распределение по видам работ представлено в таблице:

Виды учебных занятий и работы обучающихся	Трудоемкость, час		
	всего	3 семестр	4 семестр
Общая трудоемкость	**180/180**	**72/72**	**108**
Объем работы обучающихся во взаимодействии с преподавателем по видам учебных занятий (всего), в т.ч.:	**136/24**	**44/12**	**92/12**
лекции	50/8	18/4	32/4
лабораторные работы	26/4	6/2	20/2
практические занятия	56/8	18/4	38/4
курсовое проектирование (консультации)	-	-	-
Самостоятельная работа	**44/156**	**28/60**	**16/96**
Контроль (часы на зачет, контрольную работу)	**4/4**	**2/2**	**2/2**
Консультация перед экзаменом	**-**	**-**	**-**
Промежуточная аттестация		**контрольная работа**	**дифференцированный зачет**

Примечание: *-/- объем часов соответственно для очной, заочной форм обучения*

2.2. Содержание дисциплины, структурированное по темам

Коды компетенций, формированию которых способствует элемент программы	Наименование разделов, тем	Виды учебной работы				Формы текущего контроля (наименование оценочного средства)
		Работа во взаимодействии с преподавателем			Самостоятельная работа, час	
		Лекции, час	Лабораторные работы, час	Практические занятия, час		
3 семестр						
ОК 2, ПК 3.1, ПК 3.3	**Тема 1. Основы алгоритмизации** Содержание темы: 1.Алгоритмы. 2.Свойства алгоритмов. 3.Способы описания алгоритмов. 4.Основные конструкции алгоритмического языка: линейный алгоритм, ветвление, цикл	7/1				
	Практическая работа № 1. Проектирование алгоритмов. Блок-схема линейного алгоритма Практическая работа № 2. Проектирование алгоритмов. Блок-схема алгоритма ветвления Практическая работа № 3. Проектирование алгоритмов. Блок-схема цикличного алгоритма			12/1		Отчёт по практической работе
	Самостоятельная работа				4/9	Самостоятельное изучение учебных материалов
ОК 2, ПК 3.1, ПК 3.3	**Тема 2. Структуры данных** Содержание темы: 1. Типы структур данных (массивы, списки, деревья, графы и т.д.) 2. Операции над структурными данными	2/0,5				Тестирование по темам лекционных занятий
	Самостоятельная работа				4/9	Самостоятельное изучение учебных материалов
ОК 2, ПК 3.1, ПК 3.3	**Тема 3. Алгоритмы сортировки** Содержание темы: 1. Определение и importance сортировки 2. Виды алгоритмов сортировки (пузырьковая, быстрая, слиянием и т.д.)	1/0,5				

Коды компетенций, формированию которых способствует элемент программы	Наименование разделов, тем	Виды учебной работы				Формы текущего контроля (наименование оценочного средства)
		Работа во взаимодействии с преподавателем			Самостоятельная работа, час	
		Лекции, час	Лабораторные работы, час	Практические занятия, час		
	3. Сравнительный анализ алгоритмов сортировки					
	Самостоятельная работа				4/9	Самостоятельное изучение учебных материалов
ОК 2, ПК 3.1, ПК 3.3	**Тема 4. Алгоритмы поиска** Содержание темы: 1. Определение и importance поиска 2. Виды алгоритмов поиска (линейный, бинарный, хеш-поиск и т.д.) 3. Сравнительный анализ алгоритмов поиска	1/0,5				
	Самостоятельная работа				4/9	Самостоятельное изучение учебных материалов
ОК 2, ПК 3.1, ПК 3.3	**Тема 5. Введение в Python** Содержание темы: 1. История создания и особенности языка Python 2. Установка и настройка окружения для программирования на Python 3. Синтаксис и основные конструкции языка Python	3/0.5				
	Лабораторная работа № 1. Ввод-вывод данных на языке Python		2/1			Отчёт по лабораторной работе
	Практическая работа № 4. Составление программ линейной структуры			4/0.5		
	Самостоятельная работа				4/9	Самостоятельное изучение учебных материалов
ОК 2, ПК 3.1, ПК 3.3	**Тема 6. Типы данных и переменные** Содержание темы: 1. Типы данных в Python (целочисленные, вещественные, строковые, логические и т.д.) 2. Операции над типами данных	2/0,5				

Коды компетенций, формированию которых способствует элемент программы	Наименование разделов, тем	Виды учебной работы				Формы текущего контроля (наименование оценочного средства)
		Работа во взаимодействии с преподавателем			Самостоятельная работа, час	
		Лекции, час	Лабораторные работы, час	Практические занятия, час		
	3. Переменные и их объявление 4. Оператор присваивания и приоритет операций					
	Самостоятельная работа				4/9	Самостоятельное изучение учебных материалов
ОК 2, ПК 3.1, ПК 3.3	**Тема 7. Условный оператор** Содержание темы: 1. Условные операторы (if, elif, else) 2. Вложенные условные операторы	2/0,5				Тестирование по темам лекционных занятий
	Лабораторная работа № 2. Условный оператор на языке Python		4/1			Отчёт по лабораторной работе
	Практическая работа № 5. Составление программ разветвляющей структуры			2/0,5		Отчёт по практической работе
	Самостоятельная работа				4/6	Самостоятельное изучение учебных материалов
	ИТОГО за 3 семестр	**18/4**	**6/2**	**18/2**	**28/60**	
4 семестр						
ПК 3.1, ПК 3.3	**Тема 8. Циклические операторы** Содержание темы: 1. Циклы (for, while) 2. Операторы break, continue и pass 3. Вложенные циклы	8/0,5				
	Лабораторная работа № 3. Работа с циклами в Python. Цикл for Лабораторная работа № 4. Работа с циклами в Python. Цикл while		8/0,5			Отчёт по лабораторной работе

Коды компетенций, формированию которых способствует элемент программы	Наименование разделов, тем	Виды учебной работы				Формы текущего контроля (наименование оценочного средства)
		Работа во взаимодействии с преподавателем			Самостоятельная работа, час	
		Лекции, час	Лабораторные работы, час	Практические занятия, час		
	Практическая работа № 6. Составление программ циклической структуры			5/1		Отчёт по практической работе
	Самостоятельная работа				4/15	Самостоятельное изучение учебных материалов
ПК 3.1, ПК 3.3	**Тема 9. Функции** Содержание темы: 1. Определение и объявление функций 2. Аргументы и параметры функции (обязательные, необязательные, по умолчанию, переменное количество) 3. Возвращение значения из функции 4. Рекурсивные функции 5. Встроенные функции Python	5/0,5				
	Практическая работа № 7. Организация процедур Практическая работа № 8. Организация функций Практическая работа № 9.Применение рекурсивных функций			12/0.5		Отчёт по практической работе
	Самостоятельная работа				2/15	Самостоятельное изучение учебных материалов
ПК 3.1, ПК 3.3	**Тема 10. Списки и кортежи** Содержание темы: 1. Определение и создание списков и кортежей 2. Операции над списками и кортежами (добавление, удаление, поиск и т.д.) 3. Методысписковикортежей (append, insert, remove, pop, sort, reverse ит.д.) 4. Работа со срезами	4/0,5				
	Лабораторная работа № 5. Создание списков и кортежей		4/0,5			Отчёт по лабораторной

Коды компетенций, формированию которых способствует элемент программы	Наименование разделов, тем	Виды учебной работы				Формы текущего контроля (наименование оценочного средства)
		Работа во взаимодействии с преподавателем			Самостоятельная работа, час	
		Лекции, час	Лабораторные работы, час	Практические занятия, час		
						работе
	Практическая работа № 10. Обработка одномерных массивов Практическая работа № 11. Обработка двумерных массивов			8/0,5		Отчёт по практической работе
	Самостоятельная работа				2/15	Самостоятельное изучение учебных материалов
ПК 3.1, ПК 3.3	**Тема 11. Словари и множества** Содержание темы: 1. Определение и создание словарей и множеств 2. Операции над словарями и множествами (добавление, удаление, поиск и т.д.) 3. Методы словарей и множеств (keys, values, items, add, remove, pop и т.д.) 4. Работа со словарями как с ассоциативными массивами	4/0,5				
	Лабораторная работа № 6.Словари, кортежи, множества		4/0,5			Отчёт по лабораторной работе
	Практическая работа № 12. Работа с данными типа множество			4/0,5		Отчёт по практической работе
	Самостоятельная работа				2/15	Самостоятельное изучение учебных материалов
ПК 3.1, ПК 3.3	**Тема 12. Строки** Содержание темы: 1. Определение и создание строк 2. Операции над строками (конкатенация, повторение, поиск подстроки, замена подстроки и т.д.) 3. Методыстрок (lower, upper, capitalize, title, strip, replaceит.д.)	5/0,5				Тестирование по темам лекционных занятий

Коды компетенций, формированию которых способствует элемент программы	Наименование разделов, тем	Виды учебной работы				Формы текущего контроля (наименование оценочного средства)
		Работа во взаимодействии с преподавателем			Самостоятельная работа, час	
		Лекции, час	Лабораторные работы, час	Практические занятия, час		
	4. Форматирование строк с помощью метода format()					
	Практическая работа № 13. Работа со строками			4/0,5		Отчёт по практической работе
	Самостоятельная работа				2/15	Самостоятельное изучение учебных материалов
ПК 3.1, ПК 3.3	**Тема 13. Модули и пакеты** Содержание темы: 1. Определение и importance модулей и пакетов 2. Импорт модулей и пакетов 3. Создание собственных модулей и пакетов 4. Подключение модулей и пакетов через переменную sys.path	3/0,5				
	Лабораторная работа № 7. Создание собственных модулей и пакетов		4/0,5			Отчёт по лабораторной работе
	Самостоятельная работа				2/10	Самостоятельное изучение учебных материалов
ПК 3.1, ПК 3.3	**Тема 14. Работа с файлами** Содержание темы: 1. Чтение и запись файлов в текстовом и двоичном режиме 2. Работа с файлами в режиме добавления и перезаписи 3. Исключения, возникающие при работе с файлами 4. Работа с файлами через контекстный менеджер with	3/1				Тестирование по темам лекционных занятий
	Практическая работа № 14. Составление программ с использованием текстовых файлов			5/1		Отчёт по практической работе

Коды компетенций, формированию которых способствует элемент программы	Наименование разделов, тем	Виды учебной работы				Формы текущего контроля (наименование оценочного средства)
		Работа во взаимодействии с преподавателем			Самостоятельная работа, час	
		Лекции, час	Лабораторные работы, час	Практические занятия, час		
	Самостоятельная работа				2/11	Самостоятельное изучение учебных материалов
	ИТОГО за 4 семестр	**32/4**	**20/2**	**38/4**	**16/96**	
	ИТОГО	**50/8**	**26/4**	**56/8**	**44/156**	

Примечание: *-/- объем часов соответственно для очной, заочной форм обучения*

2.3.Формы и критерии текущего контроля успеваемости

3 семестр:

Формы текущего контроля	Количество контрольных точек	Количество баллов за 1 контр.точку	Макс. возм. кол-во баллов
Отчёт по практической работе	5	10	50
Отчёт по лабораторной работе	2	10	20
Тестирование по темам лекционных занятий	2	15	30
		Итого по дисциплине	**100 баллов**

4 семестр:

Формы текущего контроля	Количество контрольных точек	Количество баллов за 1 контр.точку	Макс. возм. кол-во баллов
Отчёт по практической работе	9	5	45
Отчёт по лабораторной работе	5	5	25
Тестирование по темам лекционных занятий	2	15	30
		Итого по дисциплине	**100 баллов**

2.4. Шкала оценки результатов освоения дисциплины, сформированности результатов обучения

Форма проведения промежуточной аттестации	Условия допуска	Шкалы оценки уровня сформированности результатов обучения		Шкала оценки уровня освоения дисциплины		
		Уровневая шкала оценки компетенций	100 бальная шкала, %	100 бальная шкала, %	5-балльная шкала, дифференцированная оценка/балл	недифференцированная оценка
Контрольная работа Дифференцированный зачет	допускаются все студенты	допороговый	ниже 61	ниже 61	«неудовлетворительно» / 2	не зачтено
		пороговый	61-85,9	61-69,9	«удовлетворительно» / 3	зачтено
				70-85,9	«хорошо» / 4	зачтено
		повышенный	86-100	86-100	«отлично» / 5	зачтено

3.МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

3.1. Общие методические рекомендации по освоению дисциплины, образовательные технологии

Дисциплина реализуется посредством проведения контактной работы с обучающимися (включая проведение текущего контроля успеваемости), самостоятельной работы обучающихся и промежуточной аттестации.

Контактная работа может быть аудиторной, внеаудиторной, а также проводиться в электронной информационно-образовательной среде университета (далее - ЭИОС).В случае проведения части контактной работы по дисциплине в ЭИОС (в соответствии с расписанием учебных занятий), трудоемкость контактной работа в ЭИОС эквивалентна аудиторной работе.

При проведении учебных занятий по дисциплине обеспечивается развитие у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств (включая проведение интерактивных лекций, групповых дискуссий, ролевых игр, тренингов, анализ ситуаций и имитационных моделей, преподавание дисциплины в форме курса, составленного на основе результатов научных исследований, проводимых университетом, в том числе с учетом региональных особенностей профессиональной деятельности выпускников и потребностей работодателей).

Преподавание дисциплины ведется с применением следующих видов образовательных технологий:

-*балльно-рейтинговая технология оценивания;*

- *электронное обучение;*

- *проблемное обучение;*

- *разбор конкретных ситуаций;*

Для оценки знаний, умений, навыков и уровня сформированности компетенции по дисциплине применяется балльно-рейтинговая система контроля и оценки успеваемости студентов. В основу балльно-рейтинговой системы положены принципы, в соответствии с которыми формирование рейтинга студента осуществляется в ходе текущего контроля успеваемости. Максимальное количество баллов в семестре – 100.

По итогам текущей успеваемости студенту может быть выставлена оценка по промежуточной аттестации в соответствии за набранными за семестр баллами.Студентам, набравшим в ходе текущего контроля успеваемости по дисциплине от 61 до 100 баллов и выполнившим все обязательные виды запланированных учебных занятий, по решению преподавателя без прохождения промежуточной аттестации выставляется оценка в соответствии со шкалой оценки результатов освоения дисциплины.

Результат обучения считается сформированным (повышенный уровень), если теоретическое содержание курса освоено полностью; при устных собеседованиях студент исчерпывающе, последовательно, четко и логически стройно излагает учебный материал; свободно справляется с задачами, вопросами и другими видами заданий, требующих применения знаний, использует в ответе дополнительный материал; все предусмотренные рабочей учебной программой задания выполнены в соответствии с установленными требованиями, студент способен анализировать полученные результаты; проявляет самостоятельность при выполнении заданий, качество их выполнения оценено числом баллов от 86 до 100, что соответствует повышенному уровню сформированности результатов обучения.

Результат обучения считается сформированным (пороговый уровень), если теоретическое содержание курса освоено полностью; при устных собеседованиях студент последовательно, четко и логически стройно излагает учебный материал; справляется с задачами, вопросами и другими видами заданий, требующих применения знаний; все предусмотренные рабочей учебной программой задания выполнены в соответствии с установленными требованиями, студент способен анализировать полученные результаты; проявляет самостоятельность при выполнении заданий, качество их выполнения оценено

числом баллов от 61 до 85,9, что соответствует пороговому уровню сформированности результатов обучения.

Результат обучения считается несформированным, если студент при выполнении заданий не демонстрирует знаний учебного материала, допускает ошибки, неуверенно, с большими затруднениями выполняет задания, не демонстрирует необходимых умений, качество выполненных заданий не соответствует установленным требованиям, качество их выполнения оценено числом баллов ниже 61, что соответствует допороговому уровню.

3.2. Методические указания по самостоятельной работе обучающихся

Самостоятельная работа обеспечивает подготовку обучающегося к аудиторным занятиям и мероприятиям текущего контроля и промежуточной аттестации по изучаемой дисциплине. Результаты этой подготовки проявляются в активности обучающегося на занятиях и в качестве выполненных практических заданий и других форм текущего контроля.

При выполнении заданий для самостоятельной работы рекомендуется проработка материалов лекций по каждой пройденной теме, а также изучение рекомендуемой литературы, представленной в Разделе 4.

В процессе самостоятельной работы при изучении дисциплины студенты могут использовать в специализированных аудиториях для самостоятельной работы компьютеры, обеспечивающему доступ к программному обеспечению, необходимому для изучения дисциплины, а также доступ через информационно-телекоммуникационную сеть «Интернет» к электронной информационно-образовательной среде университета (ЭИОС) и электронной библиотечной системе (ЭБС), где в электронном виде располагаются учебные и учебно-методические материалы, которые могут быть использованы для самостоятельной работы при изучении дисциплины.

Для обучающихся по заочной форме обучения самостоятельная работа является основным видом учебной деятельности.

3.3. Методические указания для выполнения курсового проекта / работы

Не предусмотрено учебным планом.

4. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

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

Основная литература:

1. Гуриков, С. Р. Основы алгоритмизации и программирования на Python : учеб.пособие для студентов СПО по специальностям: 09.02.01 "Компьютер. системы и комплексы", 09.02.02 "Компьютер. сети", 09.02.03 "Программирование в компьютер. системах", 09.02.04 "Информ. системы (по отраслям)", 09.02.05 "Приклад. информатика (по отрас / С. Р. Гуриков. – Документ read. – Москва : ИНФРА-М, 2025. – 343 с. – (Среднее профессиональное образование). – Прил. – URL: https://znanium.ru/read?id=457331 (дата обращения: 04.04.2025). - Режим доступа: для авториз. пользователей. – ISBN 978-5-16-109958-2. – Текст : электронный.

2. Канцедал, С. А. Алгоритмизация и программирование : учеб.пособие для сред. проф. образования по специальности "Информатика и вычисл. техника" / С. А. Канцедал. - Документ read. - Москва : ФОРУМ [и др.], 2021. - 352 с. : табл., схем. - (Профессиональное образование). - URL: https://znanium.com/read?id=364617 (дата обращения: 01.04.2025). - Режим доступа: для авториз. пользователей. - ISBN 978-5-8199-0727-6. - 978-5-16-100506-4. - Текст : электронный.

3. Колдаев, В. Д. Основы алгоритмизации и программирования : учеб.пособие для сред. проф. образования по группе специальностей 09.00.00 "Информатика и вычисл. техника" / В. Д. Колдаев ; под ред. Л. Г. Гагариной. - Документ read. - Москва : ФОРУМ [и др.], 2022. - 414 с. : ил., табл. - (Среднее профессиональное образование). - Прил. - Глоссарий. - URL:

https://znanium.ru/read?id=378685 (дата обращения: 13.02.2025). - Режим доступа: для авториз. пользователей. - ISBN 978-5-8199-0733-7. - 978-5-16-103967-0. - Текст : электронный.

4. Фризен, И. Г. Основы алгоритмизации и программирования (среда PascalABC.NET) : учеб.пособие для сред. проф. образования по специальностям 09.02.04 "Информ. системы (по отраслям)", 09.02.03 "Программир. в компьютер. системах" / И. Г. Фризен. - Документ read. - Москва : ФОРУМ [и др.], 2023. - 392 с. - (Среднее профессиональное образование). - URL: https://znanium.com/read?id=417233 (дата обращения: 20.03.2025). - Режим доступа: для авториз. пользователей. - ISBN 978-5-00091-005-4. - 978-5-16-105049-1. - Текст : электронный.

Дополнительная литература:

5. Андрианова, А. А. Алгоритмизация и программирование. Практикум : учеб.пособие / А. А. Андрианова, Л. Н. Исмагилов, Т. М. Мухтарова. - Документ Reader. - Санкт-Петербург [и др.] : Лань, 2022. - 236 с. - (Учебники для вузов. Специальная литература). - URL: https://reader.lanbook.com/book/206258 (дата обращения: 03.02.2025). - Режим доступа: для авториз. пользователей. - ISBN 978-5-8114-3336-0. - Текст : электронный.

6. Волк, В. К. Базы данных. Проектирование, программирование, управление и администрирование : учеб.для СПО / В. К. Волк. - Изд. 3-е, стер. - Документ read. - Санкт-Петербург [и др.] : Лань, 2024. - 338 с. - Прил. - URL: https://reader.lanbook.com/book/382310 (дата обращения: 27.03.2025). - Режим доступа: для авториз. пользователей. - ISBN 978-5-507-47482-0. - Текст : электронный.

7. Гвоздева, В. А. Информатика, автоматизированные информационные технологии и системы : учеб.для СПО по техн. специальностям / В. А. Гвоздева. - Документ read. - Москва : ФОРУМ [и др.], 2025. - 542 с. - (Среднее профессиональное образование). - URL: https://znanium.ru/read?id=451818 (дата обращения: 26.03.2025). - Режим доступа: для авториз. пользователей. - ISBN 978-5-8199-0856-3. - 978-5-16-107194-6. - Текст : электронный.

8. Голицына, О. Л. Языки программирования : учеб.пособие для сред. проф. образования / О. Л. Голицына, Т. Л. Партыка, И. И. Попов. - 3-е изд., перераб. и доп. - Документ read. - Москва : ФОРУМ [и др.], 2023. - 399 с. - Глоссарии терминов. - URL: https://znanium.ru/read?id=422756 (дата обращения: 25.03.2025). - Режим доступа: для авториз. пользователей. - ISBN 978-5-00091-613-1. - 978-5-16-102775-2. - Текст : электронный.

9. Канакова, С. Г. Основы алгоритмизации и программирования. Практикум : учеб.пособие / С. Г. Канакова. – Документ read. – Москва : ИНФРА-М, 2025. – 243 с. – (Среднее профессиональное образование). – URL: https://znanium.ru/read?id=451087 (дата обращения: 04.04.2025). - Режим доступа: для авториз. пользователей. – ISBN 978-5-16-110333-3. – Текст : электронный.

4.2. Профессиональные базы данных, информационно-справочные системы, интернет-ресурсы

1. eLIBRARY.RU : научная электронная библиотека : сайт. – Москва, 2000 - . - URL: https://elibrary.ru (дата обращения: 03.09.2024). – Режим доступа: для зарегистрир. пользователей. – Текст: электронный.
2. КонсультантПлюс : справочная правовая система : сайт / ЗАО «КонсультантПлюс». – Москва, 1992 - . - URL: http://www.consultant.ru (дата обращения: 03.09.2024). - Текст : электронный.
3. Министерство финансов России : сайт. – Москва, 1998 - . - URL: http://minfin.ru/ru/ (дата обращения: 03.09.2024). - Текст : электронный.
4. Министерство экономического развития РФ : сайт - URL: http:// www.Economy.gov.ru / (дата обращения: 03.09.2024). - Текст: электронный.
5. Федеральная служба государственной статистики : сайт. - Москва, 1999 - . - URL: http://www.gks.ru/ (дата обращения: 03.09.2024). - Текст: электронный.
6. Электронная библиотечная система Поволжского государственного университета сервиса : сайт / ФГБОУ ВО «ПВГУС». – Тольятти, 2010 - . - URL. :http://elib.tolgas.ru(дата обращения: 03.09.2024). - Режим доступа: для авториз. пользователей. - Текст : электронный.

7. Электронно-библиотечная система Znanium.com: сайт / ООО "ЗНАНИУМ". – Москва, 2011 - . - URL: https://znanium.com/ (дата обращения: 03.09.2024). - Режим доступа: для авториз. пользователей. - Текст : электронный.

8. Электронно-библиотечная система Лань: сайт / ООО "ЭБС ЛАНЬ". - Москва, 2011 - . - URL: https://e.lanbook.com/ (дата обращения: 03.09.2024). - Режим доступа: для авториз. пользователей. - Текст : электронный.

4.3. Программное обеспечение

Информационное обеспечениеучебногопроцесса по дисциплине осуществляется с использованиемследующегопрограммногообеспечения (лицензионного и свободно распространяемого), в том числе отечественного производства:

№ п/п	Наименование	Условия доступа
1	MicrosoftWindows	из внутренней сети университета (лицензионный договор)
2	MicrosoftOffice	из внутренней сети университета (лицензионный договор)
3	КонсультантПлюс	из внутренней сети университета (лицензионный договор)
4	СДО MOODLE	из любой точки, в которой имеется доступ к сети Интернет (лицензионный договор)

5. ОПИСАНИЕ МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЙ БАЗЫ, НЕОБХОДИМОЙ ДЛЯ ОСУЩЕСТВЛЕНИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ПО ДИСЦИПЛИНЕ (МДК)

Специальные помещения представляют собой учебные аудитории для проведения занятий всех видов, предусмотренных образовательной программой, в том числе групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы, мастерские и лаборатории, оснащенные оборудованием, техническими средствами обучения и материалами, учитывающими требования международных стандартов.

Занятия лекционного типа.Учебные аудитории для занятий лекционного типа укомплектованы мебелью и техническими средствами обучения, служащими для представления учебной информации (стационарные или переносные наборы демонстрационного оборудования (проектор, экран, компьютер/ноутбук), учебно-наглядные пособия (презентации по темам лекций), обеспечивающие тематические иллюстрации, соответствующие данной программе дисциплины.

Занятия семинарского типа. Для проведения практических занятий используется учебная аудитория:

- «Универсальная лаборатория компьютерных технологий (Г-404)», оснащённая следующим оборудованием: Компьютер в сборе iC D420\ i945\ DDR2 1Gb\HDD80\ LCD Xepox 17"-10 шт., ПК iCел 3,06 Ghz\1Gb\ 80Гб\LCD 17" -2 шт., стенды ЛогАнПК(ЭЛИК 50ИН "Информатика", блок питания с спец. Программа. Экран настенный ELITE SCREENS Manual M150 XWV 2,305x229см 4:3-1шт. Проектор Acer X1373. Сетевое оборудование D-Link DES-1016D, 16-портовый; локальная сеть 100 мб с доступом в Интернет 50 мб.
- «Научная лаборатория математического моделирования (Лаборатория схемотехники ЭВМ, цифровой и микропроцессорной техники и периферийных устройств) (Т-413)», оснащённая следующим оборудованием: Компьютер в сборе: Pentium(R) Dual-Core CPU E5300@ 2.60GHz/2гб/160гб, монитор ViewSonicVa 916-6 - 4 шт., панель отображения LED - SUPRA STV-LG46S990FL; сетевое оборудование- D-Link DES-1016D, 16-портовый; локальная сеть 100 мб с доступом в Интернет 50 мб; комплект учебной мебели на 10 посадочных мест, маркерная доска.

Лабораторные работы. Для проведения лабораторных работ используется учебная аудитория:

- «Универсальная лаборатория компьютерных технологий (Т-404)», оснащённая следующим оборудованием: Компьютер в сборе iC D420\ i945\ DDR2 1Gb\HDD80\ LCD Xepox 17"-10 шт., ПК iCел 3,06 Ghz\1Gb\ 80Гб\LCD 17" -2 шт., стенды ЛогАнПК(ЭЛИК 50ИН "Информатика", блок питания с спец. Программа. Экран настенный ELITE SCREENS Manual M150 XWV 2,305x229см 4:3-1шт. Проектор Acer X1373. Сетевое оборудование D-Link DES-1016D, 16- портовый; локальная сеть 100 мб с доступом в Интернет 50 мб.

- «Научная лаборатория математического моделирования (Лаборатория схемотехники ЭВМ, цифровой и микропроцессорной техники и периферийных устройств) (Т-413)», оснащённая следующим оборудованием: Компьютер в сборе: Pentium(R) Dual-Core CPU E5300@ 2.60GHz/2гб/160гб, монитор ViewSonicVa 916-6 - 4 шт., панель отображения LED - SUPRA STV-LG46S990FL; сетевое оборудование- D-Link DES-1016D, 16-портовый; локальная сеть 100 мб с доступом в Интернет 50 мб; комплект учебной мебели на 10 посадочных мест, маркерная доска.

Промежуточная аттестация. Для проведения промежуточной аттестации по дисциплине используются компьютерные классы, оснащенные компьютерной техникой с возможностью подключения к сети Интернет и обеспечением доступа в электронную информационно-образовательную среду университета и/или учебные аудитории, укомплектованные мебелью и техническими средствами обучения.

Самостоятельная работа. Помещения для самостоятельной работы оснащены компьютерной техникой с возможностью подключения к сети «Интеренет» и доступом к электронной информационно-образовательной среде университета. Для организации самостоятельной работы обучающихся используются:

компьютерные классы университета;

библиотека (медиазал), имеющая места для обучающихся, оснащенные компьютерами сдоступом к базам данных и сети Интернет.

Электронная информационно-образовательная среда университета (ЭИОС).Каждыйобучающийсявтечениевсегопериодаобученияобеспечениндивидуальнымнеограниченным доступом к электронной информационно-образовательной среде университета (ЭИОС)http://sdo.tolgas.ru/ из любой точки, в которой имеется доступ к информационно-телекоммуникационной сети "Интернет", как на территории университета, так и вне ее.

6. ОСОБЕННОСТИ ОРГАНИЗАЦИИ ОБУЧЕНИЯ ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ И ИНВАЛИДОВ

При необходимости рабочая программа может быть адаптирована для обеспечения образовательного процесса инвалидов и лиц с ограниченными возможностями здоровья, в том числе для дистанционного обучения. Для этого требуется заявление студента (его законного представителя) и заключение психолого-медико-педагогической комиссии (ПМПК).

К предметным результатам освоения дисциплины дополнительно относятся:

1) для слепых, слабовидящих обучающихся:

- сформированность навыков письма на брайлевской печатной машинке;

2) для глухих, слабослышащих, позднооглохших обучающихся:

- сформированность и развитие основных видов речевой деятельности обучающихся - слухозрительного восприятия (с использованием слуховых аппаратов и (или) кохлеарныхимплантов), говорения, чтения, письма;

3) для обучающихся с расстройствами аутистического спектра:

-овладение основными стилистическими ресурсами лексики и фразеологии языка, основными нормами литературного языка, нормами речевого этикета; приобретение опыта их использования в речевой и альтернативной коммуникативной практике при создании устных, письменных, альтернативных высказываний; стремление к возможности выразить собственные мысли и чувства, обозначить собственную позицию.

В случае необходимости, обучающимся из числа лиц с ограниченными возможностями здоровья (по заявлению обучающегося) а для инвалидов также в соответствии с индивидуальной программой реабилитации инвалида, могут предлагаться следующие варианты

восприятия учебной информации с учетом их индивидуальных психофизических особенностей, в том числе с применением электронного обучения и дистанционных технологий:

- для лиц с нарушениями зрения: в печатной форме увеличенным шрифтом; в форме электронного документа; в форме аудиофайла (перевод учебных материалов в аудиоформат); в печатной форме на языке Брайля; индивидуальные консультации с привлечением тифлосурдопереводчика; индивидуальные задания и консультации.

- для лиц с нарушениями слуха: в печатной форме; в форме электронного документа; видеоматериалы с субтитрами; индивидуальные консультации с привлечением сурдопереводчика; индивидуальные задания и консультации.

- для лиц с нарушениями опорно-двигательного аппарата: в печатной форме; в форме электронного документа; в форме аудиофайла; индивидуальные задания и консультации.

7.ОЦЕНОЧНЫЕ МАТЕРИАЛЫ (ФОНД ОЦЕНОЧНЫХ СРЕДСТВ) ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИИ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ

7.1. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта в ходе текущего контроля успеваемости

7.1.1. Типовые задания к практическим (семинарским) занятиям

Практическая работа№1.«Проектирование алгоритмов. Блок-схема линейного алгоритма».

Цели: Изучить структуру линейных алгоритмов и их виды.

Научиться конструировать блок-схемы линейных алгоритмов.

Приобрести навыки по составлению блок-схем линейных алгоритмов на основе данного описания.

Задание:

Дана задача: "Найти и вывести значение максимального элемента из списка A, состоящего из N целых чисел".

Составьте блок-схему линейного алгоритма для решения данной задачи.

Практическая работа № 2.«Проектирование алгоритмов. Блок-схема алгоритма ветвления».

Цели:Изучить структуру алгоритмов ветвления и решений.

Научиться конструировать блок-схемы алгоритмов ветвления.

Задание:Даны задачи с ветвлением. Вам нужно составить блок-схему алгоритма для каждой из задач.

Практическая работа№ 3.«Проектирование алгоритмов. Блок-схема цикличного алгоритма».

Цель: Изучить структуру циклических алгоритмов.

Научиться конструировать блок-схемы циклических алгоритмов.

Задание:Даны задачи с циклом. Вам нужно составить блок-схему алгоритма для каждой из задач.

Практическая работа№ 4.«Составление программ линейной структуры».

Цель:закрепить навыки написания программ с использованием линейных структур, а также развить навыки проектирования и отладки программ.

Задание: Написание программы для ввода и вывода простых чисел, например, для нахождения наибольшего простого числа в заданном диапазоне.

Практическая работа№ 5.«Составление программ разветвляющей структуры».

Цель:обучиться составлению программ с использованием разветвляющей структуры, а также развитию навыков по принятию решений на основе заданных условий.

Задания для практической работы могут включать в себя:

Написать программу для определения наибольшего числа из трех введенных чисел.

Составить программу, которая запрашивает у пользователя возраст и выдает сообщение о том, имеет ли пользователь право голосовать (если возраст больше или равен 18) или нет (если возраст меньше 18).

Практическая работа№ 6. «Составление программ циклической структуры».

Цель: обучиться оставлению программ с использованием циклических структур, а также развитию навыков по повторению действий в программе до наступления определенного условия.

Задания для практической работы могут включать в себя:

Написать программу для вывода чисел от 1 до 100, используя цикл for.

Составить программу для расчета факториала числа, используя цикл while.

Практическая работа№ 7.«Организация процедур».

Цель: обучитьсянаписанию и вызову процедур (функций) в программе, а также развитию навыков по организации кода на более высоком уровне.

Задания для практической работы могут включать в себя:

Написать процедуру для вычисления суммы двух чисел и вызвать её в программе.

Составить процедуру для вывода чисел от 1 до 100 и вызвать её в программе.

Практическая работа№ 8.«Организация функций».

Цель: закрепить знания о написании и использовании функций в программах, а также научить учащихся организации кода с использованием функций на более высоком уровне.

Задания для практической работы могут включать в себя:

Написать функцию для вычисления среднего арифметического массива чисел и вызвать её в программе.

Составить функцию для проверки, является ли заданное число простым, и вызвать её в программе.

Практическая работа№ 9.«Применение рекурсивных функций».

Цель: обучитьсяприменению рекурсии в программировании для решения задач, а также развитию навыков по написанию рекурсивных функций.

Задания для практической работы могут включать в себя:

Написать рекурсивную функцию для вычисления факториала числа.

Составить рекурсивную функцию для вычисления суммы чисел от 1 до заданного числа.

Практическая работа№ 10.«Обработка одномерных массивов».

Цель: научиться обрабатывать одномерные массивы в программе, а также развить навыки по применению различных алгоритмов и структур данных для решения задач, связанных с массивами.

Задания для практической работы могут включать в себя:

Написать программу для заполнения массива случайными числами и вывода его на экран.

Составить программу для поиска заданного элемента в массиве и вывода его индекса, если он найден.

Практическая работа№ 11. «Обработка двумерных массивов».

Цель: научиться работать с двумерными массивами (массивами массивов) в программе, а также применять различные алгоритмы и структуры данных для решения задач, связанных с двумерными массивами.

Задания для практической работы могут включать в себя:

Написать программу для заполнения двумерного массива случайными числами и вывода его на экран.

Составить программу для поиска заданного элемента в двумерном массиве и вывода его индексов, если он найден.

Написать программу для подсчета количества положительных, отрицательных и нулевых элементов в двумерном массиве.

Практическая работа№ 12.«Работа с данными типа множество».

Цель: научиться работать с данными типа множество в программе, а также применять различные операции и структуры данных для решения задач, связанных с множествами.

Задания для практической работы могут включать в себя:

Написать программу для создания множества из заданного количества случайных чисел и вывода его на экран.

Составить программу для поиска общих элементов в двух множествах.

Практическая работа№ 13.«Работа со строками».

Цель: научиться работать со строками в программе, а также применять различные алгоритмы и методы для обработки строк.

Задания для практической работы могут включать в себя:

Написать программу для вывода длины заданной строки.

Составить программу для вывода строки в обратном порядке.

Написать программу для перевода строки из нижнего регистра в верхний и наоборот.

Практическая работа№ 14.«Составление программ с использованием текстовых файлов».

Цель: научиться работать с текстовыми файлами в программе, а также применять различные методы для чтения, записи и обработки данных из файлов.

Задания для практической работы могут включать в себя:

Написать программу для создания нового текстового файла и записи в него заданного текста.

Составить программу для чтения содержимого текстового файла и вывода его на экран.

Написать программу для подсчета количества строк, слов и символов в текстовом файле.

7.1.2. Типовые задания к лабораторным занятиям

Лабораторная работа № 1. «Ввод-вывод данных на языке Python».

Цель: обучитьсяосновам ввода-вывода данных в программе на языке Python, а также научить их использовать различные типы данных и методы ввода-вывода.

Задания для лабораторной работы могут включать в себя:

Написать программу, которая запрашивает у пользователя его имя и выводит приветственное сообщение.

Составить программу для ввода и вывода чисел с плавающей точкой, используя различные форматы вывода.

Написать программу для ввода и вывода строковых данных, используя разные методы объединения строк.

Лабораторная работа № 2.«Условный оператор на языке Python».

Цель: научиться использовать условный оператор в программе на языке Python для принятия решений в зависимости от заданных условий.

Задания для лабораторной работы могут включать в себя:

Написать программу, которая запрашивает у пользователя возраст и выводит сообщение о том, можно ли ему голосовать (если возраст больше или равен 18) или нет (если возраст меньше 18).

Составить программу для определения наибольшего числа из трех введенных чисел.

Лабораторная работа № 3.«Работа с циклами в Python. Цикл for».

Цель: научиться работать с циклом for в программе на языке Python для повторения блоков кода заданное количество раз.

Задания для лабораторной работы могут включать в себя:

Написать программу для вывода чисел от 1 до 100 с помощью цикла for.

Составить программу для вывода чисел от 100 до 1 с шагом -1 с помощью цикла for.

Написать программу для вывода таблицы умножения от 1 до 10 с помощью цикла for.

Лабораторная работа № 4. «Работа с циклами в Python. Цикл while».

Цель: научиться работать с циклом while в программе на языке Python для повторения блоков кода до тех пор, пока не будет достигнуто заданное условие.

Задания для лабораторной работы могут включать в себя:

Написать программу для вывода чисел от 1 до 100 с помощью цикла while.

Составить программу для вывода чисел от 100 до 1 с шагом -1 с помощью цикла while.

Написать программу для вывода таблицы умножения от 1 до 10 с помощью цикла while.

Лабораторная работа № 5.«Создание списков и кортежей».

Цель: научиться создавать списки и кортежи, а также работать с ними в программе на языке Python.

Задания для лабораторной работы могут включать в себя:

Написать программу для создания списка из 10 случайных чисел и вывода его на экран.

Составить программу для создания кортежа из заданного количества элементов и вывода его на экран.

Написать программу для создания списка из 10 первых чисел Фибоначчи и вывода его на экран.

Лабораторная работа № 6.«Словари, кортежи, множества».

Цель: научиться работать со встроенными коллекциями данных в программе на языке Python: словарями, кортежами и множествами.

Задания для лабораторной работы могут включать в себя:

Написать программу для создания словаря из 10 случайных чисел и их квадратов и вывода его на экран.

Составить программу для создания кортежа из заданного количества элементов и поиска вхождения заданного элемента в нем.

Написать программу для создания множества из первых 10 простых чисел и вывода его на экран.

Лабораторная работа № 7.«Создание собственных модулей и пакетов».

Цель: научиться создавать собственные модули и пакеты в программе на языке Python и использовать их в своих программах.

Задания для лабораторной работы могут включать в себя:

Написать модуль для генерации случайного пароля заданной длины и спакуть его в файл.

Составить модуль для вычисления среднего арифметического массива чисел и упаковать его в файл.

Написать модуль для нахождения всех простых чисел от 1 до заданного числа и упаковать его в файл.

7.1.3. Типовые тестовые задания

Вопросы для тестирования по темам

3 семестр

Тема 2. Структуры данных

1. Что такое список в Python?

a) Неизменяемая последовательность

b) Изменяемая последовательность

c) Коллекция уникальных элементов

d) Пара "ключ-значение"

2. Что такое кортеж в Python?

a) Изменяемая последовательность
b) Неизменяемая последовательность
c) Коллекция пар "ключ-значение"
d) Упорядоченное множество
3. Что такое стек?
a) Структура данных, работающая по принципу FIFO
b) Структура данных, работающая по принципу LIFO
c) Упорядоченная коллекция уникальных элементов
d) Неизменяемая последовательность
4. Что такое структура данных?
a) Алгоритм для обработки данных
b) Способ организации и хранения данных
c) Язык программирования
d) Программа для управления базами данных
5. Какой из следующих типов данных является изменяемым?
a) Кортеж
b) Строка
c) Список
d) Множество
6. Какой из следующих типов данных является неизменяемым?
a) Словарь
b) Список
c) Кортеж
d) Множество
7. Какой из следующих типов данных не допускает дубликатов?
a) Список
b) Кортеж
c) Множество
d) Словарь
8. Какой принцип работы стека?
a) FIFO (первый пришёл — первый вышел)
b) LIFO (последний пришёл — первый вышел)
c) Упорядоченный
d) Неупорядоченный
9. Какой из следующих типов данных позволяет хранить упорядоченные элементы?
a) Множество
b) Словарь
c) Кортеж
d) Все вышеперечисленные
10. Чем отличаются списки от кортежей?

Тема 7. Условный оператор
1. Какой оператор используется для проверки равенства в Python?
a) =
b) ==
c) ===
d) !=
2. Какой оператор используется для проверки неравенства в Python?
a) <>
b) !=
c) !==
d) != и ≠
3. Какой из следующих операторов используется для логического И?
a) ||

b) &&
c) and
d) &
4. Какой из следующих операторов используется для логического ИЛИ?
a) ||
b) or
c) &&
d) |
5. Какой оператор используется для проверки, является ли значение истинным?
a) if
b) while
c) for
d) elif
6. Какой оператор используется для проверки равенства в Python?
7. Как выглядит базовая структура условного оператора if в Python?
8. Какой оператор используется для логического И в условных выражениях?
9. Что произойдет, если условие в операторе if ложно и нет блока else?
10. Как проверить, находится ли переменная x в диапазоне от 1 до 10?

4 семестр

Тема 12. Строки

1. Как создать строку в Python?
2. Как получить длину строки s?
3. Как объединить две строки s1 и s2?
4. Как получить первый символ строки s?
5. Как преобразовать строку s в верхний регистр?
6. Как создать строку в Python?
a) string = "Hello"
b) string = Hello
c) string = 'Hello"
7. Как получить длину строки s?
a) length(s)
b) len(s)
c) s.length()
8. Как объединить две строки s1 и s2?
a) s1.add(s2)
b) s1 + s2
c) s1.append(s2)
9. Как получить последний символ строки s?
a) s[-1]
b) s[len(s)]
c) s.last()
10. Как преобразовать строку s в нижний регистр?
a) s.lowercase()
b) s.toLower()
c) s.lower()

Тема 14. Работа с файлами

1. Как открыть файл для чтения в Python?
a) open('file.txt', 'r')
b) open('file.txt', 'w')

c) open('file.txt', 'a')
2. Как записать строку в файл?
a) file.write('Hello')
b) file.add('Hello')
c) file.append('Hello')
3. Как прочитать все строки из файла?
a) file.read()
b) file.readlines()
c) file.all()
4. Как закрыть файл после работы с ним?
a) file.end()
b) file.close()
c) file.stop()
5. Как открыть файл для добавления данных?
a) open('file.txt', 'r')
b) open('file.txt', 'w')
c) open('file.txt', 'a')
6. Как открыть файл для чтения в Python?
7. Как записать строку в файл?
8. Как прочитать все строки из файла?
9. Как закрыть файл после работы с ним?
10. Как открыть файл для добавления данных?

7.2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта в ходе промежуточной аттестации

Форма проведения промежуточной аттестации по дисциплине): контрольная работа (3 семестр) и дифференцированный зачет(4 семестр) (по результатам накопительного рейтинга или в форме компьютерного тестирования).

Устно-письменная форма предполагается, как правило, для сдачи академической задолженности.

Вопросы для подготовки к промежуточной аттестации (контрольная работа) (3 семестр)

Контрольная работа проводится для закрепления полученных знаний и направлена на расширение, углубление и систематизацию знаний, полученных при изучении ОП.04 Основы алгоритмизации и программирования обучающимися по специальности 09.02.07 Информационные системы и программирование для промежуточной аттестации.

Содержание контрольной работы соответствует требованиям рабочей программы дисциплины и требованиям, изложенным в ФГОС СПО по специальности.

Цель контрольной работы состоит в установлении уровня подготовки студента к выполнению профессиональных задач и соответствия его подготовки требованиям ФГОС СПО по специальности 09.02.07 Информационные системы и программирование.

Контрольная работа состоит из ответов на вопросы открытого типа по темам дисциплины. Время на проведение контрольной работы – 40 минут.

№	Содержание вопроса
3 семестр	
ОК 02 Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности	
1.	Что такое алгоритм?

№	Содержание вопроса
	A) Это последовательность шагов для решения задачи B) Это набор правил программирования C) Это план действий для выполнения работы D) Все вышеперечисленное верно
2.	Какие основные свойства алгоритма? A) Дискретность, результативность, массовость B) Конечность, определенность, массовость C) Детерминированность, конечность, массовость D) Массовость, конечность, универсальность
3.	Какой тип алгоритмической структуры используется для последовательного выполнения операций? A) Линейный алгоритм B) Ветвление C) Цикл D) Рекурсия
4.	Какой из перечисленных ниже алгоритмов сортировки является наиболее эффективным по времени исполнения для почти отсортированных данных? A) Пузырьковая сортировка B) Быстрая сортировка C) Сортировка слиянием D) Сортировка вставками
5.	Какой из указанных алгоритмов сортировки обладает наилучшим временем работы в худшем случае? A) Быстрая сортировка B) Сортировка слиянием C) Сортировка выбора D) Пирамидальная сортировка
6.	Какой из алгоритмов сортировки основан на принципе "разделяй и властвуй"? A) Быстрая сортировка B) Сортировка кучей C) Сортировка слиянием D) Сортировка выбором
7.	Какой алгоритм поиска минимизирует количество сравнений при поиске в отсортированном массиве? A) Линейный поиск B) Бинарный поиск C) Поиск методом деления пополам D) Интерполяционный поиск
8.	Какой из алгоритмов поиска особенно эффективен при работе с большими объемами данных? A) Линейный поиск B) Бинарный поиск C) Jump search D) Depth-first search
9.	Какой алгоритм поиска находит элемент за минимальное время в массиве, отсортированном по возрастанию? A) Линейный поиск B) Бинарный поиск C) Interpolation search D) Fibonacci search
10.	Какие типы алгоритмических структур существуют?
11.	Назовите основные свойства алгоритма.
12.	Что такое алгоритм?

№	Содержание вопроса
13.	Какой алгоритм сортировки является устойчивым?
14.	Какой алгоритм сортировки использует принцип "разделяй и властвуй"?
15.	Какой алгоритм сортировки самый эффективный для больших объемов данных?
16.	Каково основное свойство двоичного дерева поиска (BST)?
17.	Какой из алгоритмов поиска использует метод дихотомии (разбиения на части)?
18.	Какой из алгоритмов поиска относится к классу жадных алгоритмов?
ПК 3.1 Осуществлять ревьюирование программного кода в соответствии с технической документацией	
19.	Как в Python обозначаются комментарии? A) // B) # C) <!-- --> D) /* */
20.	Какой оператор используется для вывода текста на экран в Python? A) print() B) echo() C) output() D) console.log()
21.	Как объявляется переменная в Python? A) С использованием ключевого слова var B) Просто присваивается значение C) С использованием символа $ D) С использованием ключевого слова let
22.	Как называется основной файл программы на Python? A) .exe B) .py C) .java D) .cpp
23.	Как выполняется сложение чисел в Python? A) add(5, 3) B) sum(5, 3) C) 5 + 3 D) concatenate(5, 3)
24.	Какой тип данных в Python используется для целых чисел? A) int B) float C) str D) bool
25.	Какая операция в Python возвращает остаток от деления? A) % B) / C) ** D) //
26.	Какой тип данных в Python используется для дробных чисел?
27.	Как называется динамическая типизация в Python?
28.	Какие основные типы данных есть в Python?
29.	Может ли одна переменная хранить различные типы данных в разное время?
30.	Как объявляются переменные в Python?
31.	Как запустить интерпретатор Python в интерактивном режиме?
32.	Как вывести сообщение "Привет, мир!" в Python?
33.	Как объявить переменную в Python?
34.	Как в Python создать комментарий?

№	Содержание вопроса
ПК 3.3 Производить исследование созданного программного кода с использованием специализированных программных средств с целью выявления ошибок и отклонения от алгоритма	
35.	Как в Python записывается условный оператор if? A) if (условие): B) if условие: C) if {условие}: D) if [условие]:
36.	Как правильно записать условие "если число равно 5"? A) if x = 5: B) if x == 5: C) if x is 5: D) if x === 5:
37.	Как в Python добавить блок else, если условие не выполнено? A) else if: B) elif: C) else: D) otherwise:
38.	Как записать условие "если число больше 0 и меньше 10"? A) if 0 < x and x < 10: B) if 0 > x or x > 10: C) if 0 <= x >= 10: D) if 0 && x || x > 10:
39.	Как выглядит запись условия "если число не равно 0"? A) if not x == 0: B) if x != 0: C) if !x == 0: D) if x <> 0:
40.	Как в Python записать условие "если число больше нуля"?
41.	Как указать действие, которое выполняется, если ни одно из условий не выполнено?
42.	Как добавить альтернативное условие в Python?
43.	Как соединить несколько условий в Python?
44.	Как в Python реализовать тернарный оператор?
Вопросы для подготовки к промежуточной аттестации (дифференцированный зачет) 4 семестр	
ПК 3.1 Осуществлять ревьюирование программного кода в соответствии с технической документацией	
45.	Как в Python записать цикл for, который будет перебирать элементы списка? A) for i in range(len(list)): B) for element in list: C) foreach element in list: D) while element in list:
46.	Как в Python организовать бесконечный цикл? A) while True: B) loop forever: C) repeat until False: D) infinite loop:
47.	Как в Python определяется функция? A) def function_name(): B) function function_name(): C) func function_name(): D) define function_name():
48.	Что такое массив?

№	Содержание вопроса
	A) Набор объектов, хранимых последовательно в памяти B) Неупорядоченная коллекция уникальных элементов C) Древовидная структура данных D) Списки элементов, объединенные связями
49.	Можно ли изменить элемент кортежа после его создания? A) Да B) Нет C) Только если кортеж пустой D) Только если кортеж содержит один элемент
50.	Как добавить новый элемент в словарь в Python? A) dict[key] = value B) dict.add(key, value) C) dict.insert(key, value) D) dict.put(key, value)
51.	Как получить длину строки в Python? A) len(string) B) string.length() C) string.size() D) length(string)
52.	Как пропустить текущую итерацию цикла и перейти к следующей в Python?
53.	Как в Python записать цикл while, который будет выполняться до тех пор, пока условие истинно?
54.	Как вернуть значение из функции в Python?
55.	Как в Python объявляется функция?
56.	Как создать кортеж в Python?
57.	Как создать пустой список в Python?
58.	Как создать пустой словарь в Python?
59.	Как создать пустое множество в Python?
60.	Как извлечь подстроку из строки в Python?
61.	Как объединить две строки в Python?
62.	Как открыть файл для чтения в Python?
63.	Как записать данные в файл в Python?
ПК 3.3 Производить исследование созданного программного кода с использованием специализированных программных средств с целью выявления ошибок и отклонения от алгоритма	
64.	Как импортировать модуль в Python? A) import module B) load module C) require module D) module.load()
65.	Как импортировать конкретную функцию из модуля в Python? A) from module import function B) import module.function C) use module.function D) module.import(function)
66.	Как создать пакет в Python? A) Создать папку с файлом __init__.py внутри B) Создать файл с расширением .pkg C) Использовать команду python create_package D) Создать папку с файлом package.py внутри
67.	Как установить сторонний пакет в Python? A) pip install package_name B) python install package_name

№	Содержание вопроса
	C) install package_name D) download package_name
68.	Как записать данные в файл в Python? A) file.write(data) B) write(file, data) C) save(file, data) D) put(file, data)
69.	Как прочитать содержимое файла построчно в Python? A) for line in file.readlines(): B) for line in file.read_lines(): C) for line in file.lines(): D) for line in file.line_by_line():
70.	Как организовать бесконечный цикл в Python?
71.	Как прервать выполнение цикла в Python?
72.	Как вызвать функцию в Python?
73.	Можно ли изменить элемент кортежа после его создания?
74.	Можно ли изменить элемент списка после его создания?
75.	Чем отличается словарь от множества в Python?
76.	Как проверить, содержится ли подстрока в строке в Python?