

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«СЕВЕРО-КАВКАЗСКАЯ ГОСУДАРСТВЕННАЯ АКАДЕМИЯ»

«УТВЕРЖДАЮ»

Проректор по учебной работе \_\_\_\_\_ Г.Ю. Нагорная  
« 30 » 03 2023 г.



**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

Программирование на языке высокого уровня

Уровень образовательной программы \_\_\_\_\_ бакалавриат

Направление подготовки \_\_\_\_\_ 09.03.04 Программная инженерия

Направленность (профиль) \_\_\_\_\_ Программная инженерия

Форма обучения \_\_\_\_\_ очная

Срок освоения ОП \_\_\_\_\_ 4 года

Институт \_\_\_\_\_ Цифровых технологий

Кафедра разработчик РПД \_\_\_\_\_ Общая информатика

Выпускающая кафедра \_\_\_\_\_ Прикладная информатика

Начальник \_\_\_\_\_ Семенова Л.У.  
учебно-методического управления

Директор института ЦТ \_\_\_\_\_ Тебугев Д.Б.

Заведующий выпускающей кафедрой \_\_\_\_\_ Хапаева Л.Х.

г. Черкесск, 2023 г.

## СОДЕРЖАНИЕ

- 1. Цели освоения дисциплины**
  - 2. Место дисциплины в структуре образовательной программы**
  - 3. Планируемые результаты обучения по дисциплине**
  - 4. Структура и содержание дисциплины**
    - 4.1. Объем дисциплины и виды учебной работы
    - 4.2. Содержание дисциплины
      - 4.2.1. Разделы (темы) дисциплины, виды учебной деятельности и формы контроля
      - 4.2.2. Лекционный курс
      - 4.2.3. Лабораторный практикум
      - 4.2.4. Практические занятия
    - 4.3. Самостоятельная работа обучающегося
  - 5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине**
  - 6. Образовательные технологии**
  - 7. Учебно-методическое и информационное обеспечение дисциплины**
    - 7.1. Перечень основной и дополнительной учебной литературы
    - 7.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»
    - 7.3. Информационные технологии, лицензионное программное обеспечение
  - 8. Материально-техническое обеспечение дисциплины**
    - 8.1. Требования к аудиториям (помещениям, местам) для проведения занятий
    - 8.2. Требования к оборудованию рабочих мест преподавателя и обучающихся
    - 8.3. Требования к специализированному оборудованию
  - 9. Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья**
- Приложение 1. Фонд оценочных средств**

## 1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Цель освоения дисциплины «Программирование на языке высокого уровня» состоит в формировании у обучающихся теоретических знаний в области алгоритмизации и современного программирования, включающего в себя методы проектирования, анализа и создания программных продуктов, основанных на использовании структурной и объектно-ориентированной методологии и практических навыков составления алгоритмов, воплощения их на языке программирования Python, тестирования и отладки алгоритмов/

При этом *задачами* дисциплины являются:

- приобретение обучающимися знаний основных принципов структурного, модульного, объектно-ориентированного программирования, принципов составления алгоритмов, базовых конструкций изучаемых языков программирования, этапов решения задач на компьютере;
- применение синтаксиса языка программирования Python для написания программ, реализующих заданный алгоритм;
- овладение обучающимися методами построения блок-схемы алгоритмов и навыками реализации алгоритмов на конкретном языке программирования/

## 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

2.1. Дисциплина «Программирование на языке высокого уровня» к дисциплинам по выбору части, формируемой участниками образовательных отношений Блока 1 Дисциплины (модули), имеет тесную связь с другими дисциплинами.

2.2. В таблице приведены предшествующие и последующие дисциплины, направленные на формирование компетенций дисциплины в соответствии с матрицей компетенций ОП.

### Предшествующие и последующие дисциплины, направленные на формирование компетенций

№ п/п	Предшествующие дисциплины	Последующие дисциплины
1	Алгоритмы и структуры данных Основы программирования	Теория языков программирования и методы трансляции Операционные системы и сети Анализ требований к разработке программного обеспечения

### 3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ

Планируемые результаты освоения образовательной программы (ОП) – компетенции обучающихся определяются требованиями стандарта по направлению подготовки (специальности) и формируются в соответствии с матрицей компетенций ОП

№ п/п	Номер/ индекс компетенции	Наименование компетенции (или ее части)	В результате изучения дисциплины обучающиеся должны:
1	2	3	4
1.	ПК-2.	Способен обосновать принимаемые проектные решения, осуществлять постановку и выполнение экспериментов по проверке их корректности и эффективности	ПК-2.1. Разрабатывает и управляет проектами, и реализует их с использованием современного программного обеспечения. ПК-2.5. Производит анализ методологии постановки вычислительных экспериментов по проверке их эффективности ПК-2.6. Работает с современными системами программирования и проектирования
	ПК-3	Способен использовать навыки различных технологий разработки программного обеспечения, операционных систем, сетевых технологий, средств разработки программного интерфейса, применения языков и методов формальных спецификаций, систем управления базами данных	ПК-3.3. Разрабатывает компоненты информационных систем, включая модели баз данных и модели разработки программных интерфейсов ПК-3.5. Применяет стратегии проектирования и конструирования программного продукта, используя основы моделирования и анализа программных систем ПК-3.6. Использует современные технологии разработки ПО (структурное, объектно-ориентированное)

#### 4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

##### 4.1. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Вид учебной работы	Всего часов	Семестры	
		№ 4	
		часов	
1	2	3	
<b>Аудиторная контактная работа (всего)</b>	<i>108</i>	<i>108</i>	
В том числе:			
Лекции (Л)	<i>32</i>	<i>32</i>	
Лабораторные работы (ЛР)	<i>32</i>	<i>32</i>	
Практические занятия (ПЗ)	<i>16</i>	<i>16</i>	
<b>Контактная внеаудиторная работа, в том числе</b>			
индивидуальные и групповые консультации	<i>2</i>	<i>2</i>	
<b>Самостоятельная работа обучающегося (СРО) ** (всего)</b>	<i>62</i>	<i>62</i>	
<i>Работа с книжными и электронными источниками</i>	<i>12</i>	<i>12</i>	
<i>Подготовка к контрольной работе</i>	<i>12</i>	<i>12</i>	
<i>Подготовка к лабораторным занятиям</i>	<i>12</i>	<i>12</i>	
<i>Подготовка к практическим занятиям</i>	<i>10</i>	<i>10</i>	
<i>Подготовка к текущему тестовому контролю</i>	<i>8</i>	<i>8</i>	
<i>Подготовка к промежуточному контролю</i>	<i>8</i>	<i>8</i>	
<b>Промежуточная аттестация</b>	Экзамен (Э)	Э (36)	
	<b>в том числе:</b>		
	Прием экз., час.	<i>0,5</i>	<i>0,5</i>
	Консультация, час.	<i>2</i>	<i>2</i>
	СРО, час.	<i>33,5</i>	<i>33,5</i>
<b>ИТОГО:</b> <b>Общая трудоемкость</b>	<b>часов</b>	<i>180</i>	<i>180</i>
	<b>зач. ед.</b>	<i>5</i>	<i>5</i>

## 4.2. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

### 4.2.1. Разделы (темы) дисциплины, виды учебной деятельности и формы контроля

№ п/п	№ семестра	Наименование раздела (темы) дисциплины	Виды учебной деятельности, включая самостоятельную работу обучающихся (в часах)					Формы текущей и промежуточной аттестации
			Л	ЛР	ПЗ	СРО	всего	
1	2	3	4	5	6	7	8	9
1.	4	Модули в языке программирования Python	4	2	2	8	16	Контрольная работа, лабораторные работы, практические работы, текущий тестовый контроль
2.		Классы и ООП.	4	2	2	8	16	
3.		Исключения и инструменты	2	2	2	6	12	
4.		Реализация алгоритмов обработки данных на Python	2	4	2	8	16	
5.		Введение в библиотеку NumPy	6	6	2	8	22	
6.		Манипуляции над данными с помощью пакета Pandas	6	6	2	8	22	
7.		Визуализация с помощью библиотеки Matplotlib	4	6	2	8	20	
8.		Элементы графики в Python	4	4	2	8	18	
9		Контактная внеаудиторная работа					2	
10		Промежуточная аттестация					36	Экзамен
<b>Всего часов в 4 семестре:</b>			32	32	16	62	180	

### 4.2.2. Лекционный курс

№ п/п	Наименование раздела (темы) дисциплины	Наименование темы лекции	Содержание лекции	Всего часов
1	2	3	4	5
<b>Семестр 4</b>				
1	Модули в языке программирования Python	Основы программирования модулей	Модули стандартной библиотеки. Создание модуля, использование модулей, эквивалентность инструкций <code>import</code> и <code>from</code>	2
		Пакеты модулей	Основы операции импортирования пакетов.	2

			Инструкции from и import для пакетов. Соккрытие данных в модулях	
2	Классы и ООП.	Основы программирования классов	Принципы ООП: инкапсуляция, наследование и полиморфизм. Поиск унаследованных атрибутов Инструкция class. Методы. Специализация унаследованных методов	2
		Шаблоны проектирования с классами	Пространства имен: окончание истории. Конструкторы и выражения: init and sub. Общие методы перегрузки операторов Наследование, композиция, делегирование.	2
3	Исключения и инструменты	Объекты исключений и особенности их использования	Инструкция try/except/else. Инструкция try/finally. Инструкция raise. Исключения на основе классов	2
4	Реализация алгоритмов обработки данных на Python	Алгоритмы сортировки и поиска элемента на Python	Сортировка массивов на Python. Поиск минимального, максимального элемента массива, вычисления среднего значения.	2
5	Введение в библиотеку NumPy	Введение в массивы библиотеки NumPy	Целое число в языке Python. Массивы с фиксированным типом в языке Python. Индексация массива: доступ к отдельным элементам. Срезы массивов: доступ к подмассивам. Слияние и разбиение массивов.	2
		Работа с массивами библиотеки NumPy:	Обзор универсальных функций библиотеки NumPy. Агрегирование, транслирование, сравнения. Сортировка массивов. Структурированные массивы библиотеки NumPy	4
6	Манипуляции над данными с помощью пакета Pandas	Объекты библиотеки Pandas. Индексация и выборка данных	Установка и использование библиотеки Pandas. Объект Series библиотеки Pandas. Объект DataFrame библиотеки Pandas. Объект Index библиотеки Pandas.	2
		Операции над данными в библиотеке Pandas.	Универсальные функции: сохранение индекса, выравнивание индексов, выполнение операции между объектами DataFrame и Series.	4

			Обработка отсутствующих данных	
	Визуализация с помощью библиотеки Matplotlib	Импорт matplotlib, настройка интерфейса	Использование Matplotlib в сценарии; в терминале оболочки IPython; в блокноте IPython	2
		Построение графиков в библиотеке Matplotlib	Простые линейные графики. Простые диаграммы рассеяния: функции plot и scatter. Графики плотности и контурные графики. Гистограммы, разбиения по интервалам и плотность.	2
	Элементы графики в Python	Элементы графики в Python	Принципы событийного программирования. Виджеты. Графика.	2
<b>Всего часов в 4 семестре:</b>				<b>32</b>

#### 4.2.3. Лабораторный практикум

№ п/п	Наименование раздела (темы) дисциплины	Наименование лабораторной работы	Содержание лабораторной работы	Всего часов
1	2	3	4	5
<b>Семестр 4</b>				
1	Модули в языке программирования Python	Лабораторная работа №1:	Модули стандартной библиотеки. Создание модуля, использование модулей, эквивалентность инструкций import и from	2
2	Классы и ООП.	Лабораторная работа №2	Основы операции импортирования пакетов. Инструкции from и import для пакетов. Скрытие данных в модулях	2
3	Исключения и инструменты	Лабораторная работа №3	Принципы ООП: инкапсуляция, наследование и полиморфизм. Поиск унаследованных атрибутов Инструкция class. Методы. Специализация унаследованных методов	2
4	Реализация алгоритмов обработки данных на	Лабораторная работа №4	Пространства имен: окончание истории.	4



	Python		Конструкторы и выражения: init and sub. Общие методы перегрузки операторов Наследование, композиция, делегирование.	
5	Введение в библиотеку NumPy	Лабораторная работа №5	Инструкция try/except/else. Инструкция try/finally. Инструкция raise. Исключения на основе классов	6
6	Манипуляции над данными с помощью пакета Pandas	Лабораторная работа №6	Сортировка массивов на Python. Поиск минимального, максимального элемента массива, вычисления среднего значения.	6
7	Визуализация с помощью библиотеки Matplotlib	Лабораторная работа №7	Целое число в языке Python. Массивы с фиксированным типом в языке Python. Индексация массива: доступ к отдельным элементам. Срезы массивов: доступ к подмассивам. Слияние и разбиение массивов.	6
8	Элементы графики в Python	Лабораторная работа №8	Обзор универсальных функций библиотеки NumPy. Агрегирование, транслирование, сравнения. Сортировка массивов. структурированные массивы библиотеки NumPy	4
<b>Всего часов в 4 семестре:</b>				<b>32</b>

#### 4.2.4. Практические занятия

№ п/п	Наименование раздела дисциплины	Наименование практического занятия	Содержание практического занятия	Всего часов
				ОФО
1	2	3	4	5
<b>Семестр 4</b>				
1	Модули в языке	Модули в языке программирования	Модули стандартной библиотеки. Создание модуля, использование	2

	программирования Python	Python	модулей, эквивалентность инструкций <code>import</code> и <code>from</code>	
2	Классы и ООП.	Классы и ООП.	Основы операции импортирования пакетов. Инструкции <code>from</code> и <code>import</code> для пакетов. Соккрытие данных в модулях	2
3	Исключения и инструменты	Исключения и инструменты	Принципы ООП: инкапсуляция, наследование и полиморфизм. Поиск унаследованных атрибутов Инструкция <code>class</code> . Методы. Специализация унаследованных методов	2
4	Реализация алгоритмов обработки данных на Python	Реализация алгоритмов обработки данных на Python	Пространства имен: окончание истории. Конструкторы и выражения: <code>init</code> and <code>sub</code> . Общие методы перегрузки операторов Наследование, композиция, делегирование.	2
5	Введение в библиотеку NumPy	Введение в библиотеку NumPy	Инструкция <code>try/except/else</code> . Инструкция <code>try/finally</code> . Инструкция <code>raise</code> . Исключения на основе классов	2
6	Манипуляции над данными с помощью пакета Pandas	Манипуляции над данными с помощью пакета Pandas	Сортировка массивов на Python. Поиск минимального, максимального элемента массива, вычисления среднего значения.	2
7	Визуализация с помощью библиотеки Matplotlib	Визуализация с помощью библиотеки Matplotlib	Целое число в языке Python. Массивы с фиксированным типом в языке Python. Индексация массива: доступ к отдельным элементам. Срезы массивов: доступ к подмассивам. Слияние и разбиение массивов.	2
8	Элементы графики в Python	Элементы графики в Python	Обзор универсальных функций библиотеки NumPy. Агрегирование, транслирование, сравнения. Сортировка массивов. структурированные массивы библиотеки NumPy	2
<b>Всего часов в 4 семестре:</b>				<b>16</b>

#### 4.3. САМОСТОЯТЕЛЬНАЯ РАБОТА ОБУЧАЮЩЕГОСЯ

№ п/п	Наименование раздела (темы) дисциплины	Виды СРО	Всего часов
1	3	5	6
<b>Семестр 4</b>			
1	Модули в языке программирования Python	Работа с книжными источниками Работа с электронными источниками Подготовка к практическим занятиям Подготовка к лабораторным занятиям Подготовка к текущему тестовому контролю	8
2	Классы и ООП.	Работа с книжными источниками Работа с электронными источниками Подготовка к практическим занятиям Подготовка к лабораторным занятиям Подготовка к текущему тестовому контролю	8
3	Исключения и инструменты	Работа с книжными источниками Работа с электронными источниками Подготовка к лабораторным занятиям Подготовка к текущему тестовому контролю	6
4	Реализация алгоритмов обработки данных на Python	Работа с книжными источниками Работа с электронными источниками Подготовка к практическим занятиям Подготовка к лабораторным занятиям Подготовка к текущему тестовому контролю	8
5	Введение в библиотеку NumPy	Работа с книжными источниками Работа с электронными источниками Подготовка к практическим занятиям Подготовка к лабораторным занятиям Подготовка к текущему тестовому контролю	8
6	Манипуляции над данными с помощью пакета Pandas	Работа с книжными источниками Работа с электронными источниками Подготовка к лабораторным занятиям Подготовка к текущему тестовому контролю	8
7	Визуализация с помощью библиотеки Matplotlib	Работа с книжными источниками Работа с электронными источниками Подготовка к практическим занятиям Подготовка к лабораторным занятиям Подготовка к текущему тестовому контролю	8
8	Элементы графики в Python	Работа с книжными источниками Работа с электронными источниками Подготовка к лабораторным занятиям Подготовка к текущему тестовому контролю Подготовка к промежуточному контролю	8
<b>Всего часов в 4 семестре:</b>			<b>62</b>

## **5. ПЕРЕЧЕНЬ УЧЕБНО-МЕТОДИЧЕСКОГО ОБЕСПЕЧЕНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ РАБОТЫ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ**

### **5.1. Методические указания для подготовки обучающихся к лекционным занятиям**

Лекция является основной формой обучения в высшем учебном заведении. Записи лекций в конспектах должны быть избирательными, полностью следует записывать только определения. В конспекте рекомендуется применять сокращение слов, что ускоряет запись. Вопросы, возникающие в ходе лекции, рекомендуется записывать на полях и после окончания лекции обратиться за разъяснением к преподавателю.

Работа над конспектом лекции осуществляется по этапам:

- повторить изученный материал по конспекту;
- непонятные положения отметить на полях и уточнить;
- неоконченные фразы, пропущенные слова и другие недочеты в записях устранить, пользуясь материалами из учебника и других источников;
- завершить техническое оформление конспекта (подчеркивания, выделение главного, выделение разделов, подразделов и т.п.).

Самостоятельную работу следует начинать с доработки конспекта, желательно в тот же день, пока время не стерло содержание лекции из памяти. Работа над конспектом не должна заканчиваться с прослушивания лекции. После лекции, в процессе самостоятельной работы, перед тем, как открыть тетрадь с конспектом, полезно мысленно восстановить в памяти содержание лекции, вспомнив ее структуру, основные положения и выводы.

С целью доработки необходимо прочитать записи, восстановить текст в памяти, а также исправить описки, расшифровать не принятые ранее сокращения, заполнить пропущенные места, понять текст, вникнуть в его смысл. Далее прочитать материал по рекомендуемой литературе, разрешая в ходе чтения, возникшие ранее затруднения, вопросы, а также дополнения и исправляя свои записи. Записи должны быть наглядными, для чего следует применять различные способы выделений. В ходе доработки конспекта углубляются, расширяются и закрепляются знания, а также дополняется, исправляется и совершенствуется конспект. Еще лучше, если вы переработаете конспект, дадите его в новой систематизации записей. Это, несомненно, займет некоторое время, но материал вами будет хорошо проработан, а конспективная запись его приведена в удобный для запоминания вид. Введение заголовков, скобок, обобщающих знаков может значительно повысить качество записи. Этому может служить также подчеркивание отдельных мест конспекта красным карандашом, приведение на полях или на обратной стороне листа краткой схемы конспекта и др.

Подготовленный конспект и рекомендуемая литература используется при подготовке к практическому занятию. Подготовка сводится к внимательному прочтению учебного материала, к выводу с карандашом в руках всех утверждений и формул, к решению примеров, задач, к ответам на вопросы, предложенные в конце лекции преподавателем или помещенные в рекомендуемой литературе. Примеры, задачи, вопросы по теме являются средством самоконтроля.

Непременным условием глубокого усвоения учебного материала является знание основ, на которых строится изложение материала. Обычно преподаватель напоминает, какой ранее изученный материал и в какой степени требуется подготовить к очередному занятию. Эта рекомендация, как и требование систематической и серьезной работы над всем лекционным курсом, подлежит безусловному выполнению. Потери логической связи как внутри темы, так и между ними приводит к негативным последствиям: материал учебной дисциплины перестает основательно восприниматься, а творческий труд подменяется утомленным переписыванием. Обращение к ранее изученному материалу не только помогает восстановить в памяти известные положения, выводы, но и приводит

разрозненные знания в систему, углубляет и расширяет их. Каждый возврат к старому материалу позволяет найти в нем что-то новое, переосмыслить его с иных позиций, определить для него наиболее подходящее место в уже имеющейся системе знаний. Неоднократное обращение к пройденному материалу является наиболее рациональной формой приобретения и закрепления знаний. Очень полезным, но, к сожалению, еще мало используемым в практике самостоятельной работы, является предварительное ознакомление с учебным материалом. Даже краткое, беглое знакомство с материалом очередной лекции дает многое. Обучающиеся получают общее представление о ее содержании и структуре, о главных и второстепенных вопросах, о терминах и определениях. Все это облегчает работу на лекции и делает ее целеустремленной.

## **5.2. Методические указания для подготовки обучающихся к лабораторным занятиям**

Ведущей дидактической целью лабораторных занятий является систематизация и обобщение знаний по изучаемой теме, приобретение практических навыков по тому или другому разделу курса, закрепление полученных теоретических знаний. Лабораторные работы сопровождают и поддерживают лекционный курс. Подготовка к лабораторным занятиям и практикумам носит различный характер, как по содержанию, так и по сложности исполнения.

Многие лабораторные занятия требуют большой исследовательской работы, изучения дополнительной научной литературы. Прежде чем приступить к выполнению такой работы, обучающемуся необходимо ознакомиться обстоятельно с содержанием задания, уяснить его, оценить с точки зрения восприятия и запоминания все составляющие его компоненты. Это очень важно, так как при проработке соответствующего материала по конспекту лекции или по рекомендованной литературе могут встретиться определения, факты, пояснения, которые не относятся непосредственно к заданию. Обучающийся должен хорошо знать и понимать содержание задания, чтобы быстро оценить и отобрать нужное из читаемого. Далее, в соответствии со списком рекомендованной литературы, необходимо отыскать материал к данному заданию по всем пособиям.

Весь подобранный материал нужно хотя бы один раз прочитать или внимательно просмотреть полностью. По ходу чтения помечаются те места, в которых содержится ответ на вопрос, сформулированный в задании. Читая литературу по теме, обучающийся должен мысленно спрашивать себя, на какой вопрос задания отвечает тот или иной абзац прорабатываемого пособия. После того, как материал для ответов подобран, желательно хотя бы мысленно, а лучше всего устно или же письменно, ответить на все вопросы. В случае если обнаружится пробел в знаниях, необходимо вновь обратиться к литературным источникам и проработать соответствующий раздел. Только после того, как преподаватель убедится, что обучающийся хорошо знает необходимый теоретический материал, что его ответы достаточно аргументированы и доказательны, можно считать обучающегося подготовленным к выполнению лабораторных работ.

## **5.3. Методические указания для подготовки обучающихся к практическим занятиям**

В процессе подготовки к практическим занятиям обучающиеся закрепляют полученные ранее теоретические знания, приобретают навыки их практического применения, опыт рациональной организации учебной работы.

Подготовку к практическому занятию каждый обучающийся должен начать с ознакомления с планом практического занятия, который отражает содержание предложенной темы. Тщательное продумывание и изучение вопросов плана основывается на проработке текущего материала, а затем изучение обязательной и дополнительной литературы, рекомендованной к данной теме.

Все новые понятия по изучаемой теме необходимо выучить наизусть и внести в глоссарий, который целесообразно вести с самого начала изучения курса. Результат такой работы должен проявиться в способности обучающегося свободно ответить на теоретические вопросы, его выступлении и участии в коллективном обсуждении вопросов изучаемой темы, правильном выполнении практических заданий. Предлагается следующая опорная схема подготовки к практическим занятиям.

Обучающийся при подготовке к практическому занятию может консультироваться с преподавателем и получать от него наводящие разъяснения, задания для самостоятельной работы.

1. Ознакомление с темой практического занятия. Выделение главного (основной темы) и второстепенного (подразделы, частные вопросы темы).

2. Освоение теоретического материала по теме с опорой на лекционный материал, учебник и другие учебные ресурсы. Самопроверка: постановка вопросов, затрагивающих основные термины, определения и положения по теме, и ответы на них.

3. Выполнение практического задания. Обнаружение основных трудностей, их решение с помощью дополнительных интеллектуальных усилий и/или подключения дополнительных источников информации.

4. Решение типовых заданий расчетно-графической работы.

#### **5.4 Методические указания по самостоятельной работе обучающегося**

Самостоятельная работа обучающегося предполагает различные формы индивидуальной учебной деятельности: конспектирование научной литературы, сбор и анализ практического материала в СМИ, проектирование, выполнение тематических и творческих заданий и пр. Выбор форм и видов самостоятельной работы определяется индивидуально-личностным подходом к обучению совместно преподавателем и обучающимся. Формы текущего контроля успеваемости и промежуточной аттестации обучающихся.

Содержание внеаудиторной самостоятельной работы обучающегося по дисциплине «Программирование на языке высокого уровня» включает в себя следующие виды деятельности:

- Работа с электронными источниками
- Подготовка к лабораторным занятиям
- Подготовка к практическим занятиям
- Подготовка к текущему тестовому контролю
- Подготовка к промежуточному контролю
- Работа с книжными источниками
- Подготовка к контрольной работе

В процессе подготовки к практическим занятиям, обучающимся необходимо обратить особое внимание на самостоятельное изучение рекомендованной учебно-методической (а также научной и популярной) литературы.

Самостоятельная работа с учебниками, учебными пособиями, научной, справочной и популярной литературой, материалами периодических изданий и Интернета, статистическими данными является наиболее эффективным методом получения знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у обучающихся свое отношение к конкретной проблеме.

Более глубокому раскрытию вопросов способствует знакомство с дополнительной литературой, рекомендованной преподавателем по каждой теме семинарского или практического занятия, что позволяет обучающимся проявить свою индивидуальность в рамках выступления на данных занятиях, выявить широкий спектр мнений по изучаемой проблеме.

Тестирование представляет собой средство контроля усвоения учебного материала темы или раздела дисциплины. При самостоятельной подготовке к тестированию обучающимся необходимо:

а) проработать информационный материал по дисциплине, проконсультироваться с преподавателем по вопросу выбора учебной литературы;

б) выяснить все условия тестирования заранее, узнать, сколько тестов будет предложено, сколько времени отводится на тестирование, какова система оценки результатов и т.д.

в) приступая к работе с тестами, внимательно и до конца прочитать вопрос и предлагаемые варианты ответов, выбрать правильные (их может быть несколько).

г) в процессе решения желательно применять несколько подходов в решении задания, это позволяет максимально гибко оперировать методами решения, находя каждый раз оптимальный вариант.

д) если встретился чрезвычайно трудный вопрос, не тратить много времени на него, перейти к другим тестам, вернуться к трудному вопросу в конце.

е) обязательно оставить время для проверки ответов, чтобы избежать механических ошибок.

### **Контрольная работа**

Контрольная работа рассматривается как разновидность самостоятельной работы обучающихся, которая направлена на формирование умений самостоятельно работать с теоретическим материалом, анализировать важнейшие категории дисциплин, устанавливать взаимосвязь между их главными понятиями. В ходе выполнения контрольной работы обучающийся должен научиться определять практическую направленность и значимость дисциплины, ее связь с другими учебными дисциплинами и соответствующими науками.

Написание работы сводится к тщательному исследованию материала, имеющегося в теме контрольной работы, а потом к изложению результатов в каждом пункте, причем фактические данные могут предшествовать выводам или, наоборот, вытекать из них. Изложение каждого задания надо четко отделить для того, чтобы можно было ясно видеть, где начинается и где заканчивается их освещение. В основной части необходимо основательно и подробно изложить, как именно указанная проблема представлена в научной и учебной литературе. Важно показать умение использовать знание теории для объективного и всестороннего анализа исследуемой проблемы; показать свое отношение к вопросу, который изучается; выделить положительное, определить недостатки, сформулировать наиболее эффективные меры относительно их устранения.

Написание текста контрольной работы – это изложение сведений о предмете (объекте) исследования, которые являются необходимыми и достаточными для раскрытия сущности вопросов плана обозначенной работы и ее результатов. К главным требованиям, относительно содержания контрольной работы, принадлежит ее самостоятельность. Писать надо своими словами, анализируя и обобщая знания, полученные из учебной и научной литературы и законодательных (нормативно-правовых) актов. В тексте следует использовать материалы из научных источников. Изложение материала должно быть последовательным, логическим и четким, в соответствии с планом работы. Недопустимо переписывать из учебников целыми разделами! Такое цитирование не имеет ничего общего с понятием контрольной работы и является ничем другим, как плагиатом.

### **Тестирование**

Тестирование представляет собой средство контроля усвоения учебного материала темы или раздела дисциплины. При самостоятельной подготовке к тестированию обучающимся необходимо:

а) проработать информационный материал по дисциплине, проконсультироваться с

преподавателем по вопросу выбора учебной литературы;

б) выяснить все условия тестирования заранее, узнать, сколько тестов будет предложено, сколько времени отводится на тестирование, какова система оценки результатов и т.д.

в) приступая к работе с тестами, внимательно и до конца прочитать вопрос и предлагаемые варианты ответов, выбрать правильные (их может быть несколько).

г) в процессе решения желательно применять несколько подходов в решении задания, это позволяет максимально гибко оперировать методами решения, находя каждый раз оптимальный вариант.

д) если встретился чрезвычайно трудный вопрос, не тратить много времени на него, перейти к другим тестам, вернуться к трудному вопросу в конце.

е) обязательно оставить время для проверки ответов, чтобы избежать механических ошибок.

### **Работа с книжными и электронными источниками**

В процессе подготовки к практическим занятиям, студентам необходимо обратить особое внимание на самостоятельное изучение рекомендованной учебно-методической (а также научной и популярной) литературы.

Самостоятельная работа с учебниками, учебными пособиями, научной, справочной и популярной литературой, материалами периодических изданий и Интернета, статистическими данными является наиболее эффективным методом получения знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у обучающихся свое отношение к конкретной проблеме.

Более глубокому раскрытию вопросов способствует знакомство с дополнительной литературой, рекомендованной преподавателем по каждой теме семинарского или практического занятия, что позволяет студентам проявить свою индивидуальность в рамках выступления на данных занятиях, выявить широкий спектр мнений по изучаемой проблеме.

### **Методические рекомендации по подготовке к экзамену**

Экзамен – это форма итоговой отчетности студента по изученной дисциплине. По решению кафедры экзамен может проводиться в нескольких формах – устной по билетам. Главная задача проведения экзамена – проверка знаний, навыков и умений студента, по прослушанной дисциплине.

Огромную роль в успешной подготовке к экзамену играет правильная организация подготовки к нему. Рекомендуется при подготовке к экзамену опираться на следующий план:

просмотреть программу курса, с целью выявления наиболее проблемных тем, вопросов, которые могут вызвать трудности при подготовке к экзамену.

прорешать задачу, предложенные в учебно-методическом комплексе. При этом для эффективного закрепления информации первый раз без использования учебных материалов и нормативно-правовых актов, второй раз с их использованием.

При выполнении первых двух пунктов плана студент получит возможность оценить свои знания и навыки по прослушанной дисциплине и сориентироваться при планировании объема подготовки.

темы необходимо изучать последовательно, внимательно обращая внимание на описание вопросов, которые раскрывают ее содержание. Начинать необходимо с первой темы.

после работы над первой темой необходимо ответить на контрольные вопросы к теме и решить тестовые задания к ней.

после изучения всех тем студенту рекомендуется ответить на контрольные вопросы



по всему курсу.

## 6. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

№ п/п	№ семестра	Виды учебной работы	Образовательные технологии	Всего часов
1	2	3	4	
1	4	<i>Лекция «Модули в языке программирования Python»</i>	Мультимедийные и телекоммуникационные технологии	2
2		<i>Лекция «Манипуляции над данными с помощью пакета Pandas»</i>	Мультимедийные и телекоммуникационные технологии	4
3		<i>Лекция «Реализация алгоритмов обработки данных на Python»</i>	Мультимедийные и телекоммуникационные технологии	2
<b>Всего часов в 4 семестре:</b>				<b>8</b>

## 7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

### 7.1. Перечень основной и дополнительной учебной литературы

#### Основная литература

1. Дроботун, Н. В. Алгоритмизация и программирование. Язык Python : учебное пособие / Н. В. Дроботун, Е. О. Рудков, Н. А. Баев. — Санкт-Петербург : Санкт-Петербургский государственный университет промышленных технологий и дизайна, 2020. — 119 с. — ISBN 978-5-7937-1829-5. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/102400.html>
2. Сузи, Р. А. Язык программирования Python : учебное пособие / Р. А. Сузи. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 350 с. — ISBN 978-5-4497-0705-5. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/97589.html>
3. Шелудько, В. М. Основы программирования на языке высокого уровня Python : учебное пособие / В. М. Шелудько. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2017. — 146 с. — ISBN 978-5-9275-2649-9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/87461.html> . — Режим доступа: для авторизир. пользователей
4. Широков, А. И. Информатика: разработка программ на языке программирования Питон: базовые языковые конструкции : учебник / А. И. Широков, М. О. Пышняк. — Москва : Издательский Дом МИСиС, 2020. — 142 с. — ISBN 978-5-907226-76-0. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/106713.html>

#### Дополнительная литература

1. Буйначев, С. К. Основы программирования на языке Python : учебное пособие / С. К. Буйначев, Н. Ю. Боклаг ; под редакцией Ю. В. Песин. — Екатеринбург : Уральский федеральный университет, ЭБС АСВ, 2014. — 92 с. — ISBN 978-5-7996-1198-9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/66183.html>
2. Томашевский, П. Р. Привет, Python! Моя первая книга по программированию / П. Р. Томашевский. — Санкт-Петербург : Наука и Техника, 2018. — 256 с. — ISBN 978-5-94387-748-3. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/73047.html>
3. Рик, Гаско Простой Python просто с нуля / Гаско Рик. — Москва : СОЛОН-Пресс, 2019. — 256 с. — ISBN 978-5-91359-334-4. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/94940.html>

### 7.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

<http://window.edu.ru>- Единое окно доступа к образовательным ресурсам;

<http://fcior.edu.ru> - Федеральный центр информационно-образовательных ресурсов;

<http://elibrary.ru> - Научная электронная библиотека.

### 7.3. Информационные технологии, лицензионное программное обеспечение.

Лицензионное программное обеспечение	Реквизиты лицензий/ договоров
Microsoft Azure Dev Tools for Teaching 1. Windows 7, 8, 8.1, 10 2. Visual Studio 2008, 2010, 2013, 2019 5. Visio 2007, 2010, 2013 6. Project 2008, 2010, 2013 7. Access 2007, 2010, 2013 и т. д.	Идентификатор подписчика: 1203743421 Срок действия: 30.06.2022  (продление подписки)
MS Office 2003, 2007, 2010, 2013	Сведения об Open Office: 63143487, 63321452, 64026734, 6416302, 64344172, 64394739, 64468661, 64489816, 64537893, 64563149, 64990070, 65615073 Лицензия бессрочная
Антивирус Dr.Web Desktop Security Suite	Лицензионный сертификат Серийный № 8DVG-V96F-H8S7-NRBC Срок действия: с 20.10.2022 до 22.10.2023
Цифровой образовательный ресурс IPRsmart	Лицензионный договор № 10423/23П от 30.06.2023 г. Срок действия: с 01.07.2023 г. до 01.07.2024г.

#### Свободное ПО:

<https://www.python.org/downloads/release/python-3101/>  
<https://www.python.org/ftp/python/3.8.5/python-3.8.5.exe>  
[Winpython64-3.7.12.0PyPy.exe](#)

## 8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

### 8.1. Требования к аудиториям (помещениям, местам) для проведения занятий

Учебная аудитория для проведения занятий лекционного типа

Специализированная мебель:

Парты - 10шт., стулья - 29шт.; доска меловая - 1шт., кафедра настольная - 1шт., стул мягкий - 1шт., компьютерные столы-12шт., стол одностумбовый (преподавательский) -1шт., шкаф двухдверный - 1шт.

Набор демонстрационного оборудования и учебно-наглядных пособий, обеспечивающих тематические иллюстрации:

Интерактивная доска- 1шт.

Проектор - 1шт.

Ноутбук - 1шт.

ПК- 10шт.

Учебная аудитория для проведения занятий семинарского типа, курсового проектирования (выполнение курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации

Специализированная мебель:

Парты - 6шт., доска меловая - 1шт., компьютерные столы - 7шт., стол преподавательский - 3шт., стулья - 28 шт., стол лабораторный - 3 шт.

Лабораторное оборудование, технические средства обучения, служащие для предоставления учебной информации большой аудитории:

Компьютер в сборе - 7 шт.

Помещение для самостоятельной работы.

Библиотечно-издательский центр. Отдел обслуживания печатными изданиями

Специализированная мебель:

Рабочие столы на 1 место – 21 шт.

Стулья – 55 шт.

Набор демонстрационного оборудования и учебно-наглядных пособий, обеспечивающих тематические иллюстрации:

Экран настенный – 1 шт.

Проектор – 1шт.

Ноутбук – 1шт.

Информационно-библиографический отдел.

Специализированная мебель:

Рабочие столы на 1 место - 6 шт.

Стулья - 6 шт.

Компьютерная техника с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду ФГБОУ ВО «СевКавГА»:

Персональный компьютер – 1шт.

Сканер – 1 шт.

МФУ – 1 шт.

Отдел обслуживания электронными изданиями

Специализированная мебель:

Рабочие столы на 1 место – 24 шт.

Стулья – 24 шт.

Набор демонстрационного оборудования и учебно-наглядных пособий, обеспечивающих тематические иллюстрации:

Интерактивная система - 1 шт.

Монитор – 21 шт.

Сетевой терминал -18 шт.

Персональный компьютер -3 шт.

МФУ – 2 шт.

Принтер –1шт.

Помещение для хранения и профилактического обслуживания учебного оборудования

Ауд. 11 Специализированная мебель:

Стеллажи – 1 шт., шкаф – 1 шт., стул -1 шт., кресло компьютерное – 4 шт., стол – 5 шт.

Профилактическое оборудование:

Перфоратор -1 шт.

Аккумуляторная дрель-шуруповерт – 1 шт., наборы отверток -2 шт., пылесос -1 шт., клещи обжимные – 3 шт., тестер блоков питания -1 шт., мультиметр -1 шт., фен термовоздушный паяльный -1 шт., паяльник -3 шт.

Учебное пособие (персональный компьютер в комплекте) – 2 шт., пассатижи – 1 шт., бокорезы-1 шт.

Коммутатор -1 шт., внешний DVD привод -1 шт., внешний жесткий диск - 1 шт.

## **8.2. Требования к оборудованию рабочих мест преподавателя и обучающихся**

1. рабочее место преподавателя, оснащенное компьютером с доступом в Интернет,
2. рабочие места обучающихся, оснащенные компьютерами с доступом в Интернет, предназначенные для работы в электронной образовательной среде.
3. В компьютерном классе должны быть установлены средства MS Office: Word, Excel, PowerPoint.

## **8.3. Требования к специализированному оборудованию нет**

# **9. ОСОБЕННОСТИ РЕАЛИЗАЦИИ ДИСЦИПЛИНЫ ДЛЯ ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ**

Для обеспечения образования инвалидов и обучающихся с ограниченными возможностями здоровья разрабатывается (в случае необходимости) адаптированная образовательная программа, индивидуальный учебный план с учетом особенностей их психофизического развития и состояния здоровья, в частности применяется индивидуальный подход к освоению дисциплины, индивидуальные задания: рефераты, письменные работы и, наоборот, только устные ответы и диалоги, индивидуальные консультации, использование диктофона и других записывающих средств для воспроизведения лекционного и семинарского материала.

В целях обеспечения обучающихся инвалидов и лиц с ограниченными возможностями здоровья комплектуется фонд основной учебной литературой, адаптированной к ограничению электронных образовательных ресурсов, доступ к которым организован в БИЦ Академии. В библиотеке проводятся индивидуальные консультации для данной категории пользователей, оказывается помощь в регистрации и использовании сетевых и локальных электронных образовательных ресурсов, предоставляются места в читальном зале.

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**

**ПО ДИСЦИПЛИНЕ** \_\_\_\_\_ Программирование на языке высокого уровня \_\_\_\_\_

# 1. ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО ДИСЦИПЛИНЕ

## Программирование на языке высокого уровня

### 1. Компетенции, формируемые в процессе изучения дисциплины

Индекс	Формулировка компетенции
ПК-2.	Способен обосновать принимаемые проектные решения, осуществлять постановку и выполнение экспериментов по проверке их корректности и эффективности
ПК-3	Способен использовать навыки различных технологий разработки программного обеспечения, операционных систем, сетевых технологий, средств разработки программного интерфейса, применения языков и методов формальных спецификаций, систем управления базами данных

### 2. Этапы формирования компетенции в процессе освоения дисциплины

Основными этапами формирования указанных компетенций при изучении обучающимися дисциплины являются последовательное изучение содержательно связанных между собой разделов (тем) учебных занятий. Изучение каждого раздела (темы) предполагает овладение обучающимися необходимыми компетенциями. Результат аттестации обучающихся на различных этапах формирования компетенций показывает уровень освоения компетенций обучающимися.

Этапность формирования компетенций прямо связана с местом дисциплины в образовательной программе.

Разделы (темы) дисциплины	Формируемые компетенции (коды)	
	ПК-2	ПК-3.
Модули в языке программирования Python	+	+
Классы и ООП.	+	+
Исключения и инструменты	+	+
Реализация алгоритмов обработки данных на Python	+	+
Введение в библиотеку NumPy	+	+
Манипуляции над данными с помощью пакета Pandas	+	+
Визуализация с помощью библиотеки Matplotlib	+	+
Элементы графики в Python	+	+



### 3. Показатели, критерии и средства оценивания компетенций, формируемых в процессе изучения дисциплины

<b>ПК-2. Способен обосновать принимаемые проектные решения, осуществлять постановку и выполнение экспериментов по проверке их корректности и эффективности</b>						
<b>Индикаторы достижения компетенции</b>	<b>Критерии оценивания результатов обучения</b>				<b>Средства оценивания результатов обучения</b>	
	<b>неудовлетв</b>	<b>удовлетв</b>	<b>хорошо</b>	<b>отлично</b>	Текущий контроль	Промежуточная аттестация
ПК-2.1. Разрабатывает и управляет проектами, и реализует их с использованием современного программного обеспечения.	Не знает современного программного обеспечения и не применяет их в реализации проектов	Демонстрирует современного программного обеспечения и неуверенно применяет их в реализации проектов	Хорошие знания современного программного обеспечения и методов применения их в реализации проектов	Демонстрирует отличные знания современного программного обеспечения и методов применения их в реализации проектов	Контрольная работа, проверка лабораторных работ, текущий контроль проверки практических работ	Экзамен
ПК-2.5. Производит анализ методологии постановки вычислительных экспериментов по проверке их эффективности	Не умеет и не готов проводить анализ методологии постановки вычислительных экспериментов по проверке их эффективности	Не уверенно анализирует методологии постановки вычислительных экспериментов по проверке их эффективности	Уверенно использует методологии постановки вычислительных экспериментов по проверке их эффективности	Готов и умеет использовать методологии постановки вычислительных экспериментов по проверке их эффективности		
ПК-2.6. Работает с современными системами программирования и проектирования	Не владеет основными навыками работы с современными системами программирования и проектирования.	Частично владеет основными навыками работы с современными системами программирования и проектирования.	Владеет основными навыками работы с современными системами программирования и проектирования.	Отличное владение основными навыками работы с современными системами программирования и проектирования.		

**ПК-3 Способен использовать навыки различных технологий разработки программного обеспечения, операционных систем, сетевых технологий, средств разработки программного интерфейса, применения языков и методов формальных спецификаций, систем управления базами данных**

Индикаторы достижения компетенции	Критерии оценивания результатов обучения				Средства оценивания результатов обучения	
	неудовлетв	удовлетв	хорошо	отлично	Текущий контроль	Промежуточная аттестация
ПК-3.3. Разрабатывает компоненты информационных систем, включая модели баз данных и модели разработки программных интерфейсов	Не знает компоненты информационных систем, включая модели баз данных и модели разработки программных интерфейсов	Демонстрирует частичные знания методов разработки компонент информационных систем, включая модели баз данных и модели разработки программных интерфейсов	Хорошие знания методов разработки компонент информационных систем, включая модели баз данных и модели разработки программных интерфейсов	Демонстрирует отличные знания методов разработки компонент информационных систем, включая модели баз данных и модели разработки программных интерфейсов	Контрольная работа, проверка лабораторных работ, текущий тестовый контроль проверка практических работ	Экзамен
ПК-3.5. Применяет стратегии проектирования и конструирования программного продукта, используя основы моделирования и анализа программных систем	Не умеет и не готов применять стратегии проектирования и конструирования программного продукта, используя основы моделирования и анализа программных систем	Не уверено применяет стратегии проектирования и конструирования программного продукта, используя основы моделирования и анализа программных систем	Уверено применяет стратегии проектирования и конструирования программного продукта, используя основы моделирования и анализа программных систем	Готов и умеет применять стратегии проектирования и конструирования программного продукта, используя основы моделирования и анализа программных систем		
ПК-3.6. Использует современные технологии разработки ПО (структурное, объектно-ориентированное)	Не владеет современными технологиями разработки ПО (структурное, объектно-ориентированное)	Частично владеет современными технологиями разработки ПО (структурное, объектно-ориентированное)	Владеет современными технологиями разработки ПО (структурное, объектно-ориентированное)	Отличное владение современными технологиями разработки ПО (структурное, объектно-ориентированное)		

## 4. Комплект контрольно-оценочных средств по дисциплине Основы и программирования

### Вопросы к экзамену

По дисциплине Программирование на ЯВУ

1. Типы данных.
2. Переменные.
3. Числовые типы данных.
4. Операции над числовыми типами данных.
5. Строки.
6. Строки `unicod`.
7. Вывод данных.
8. Ввод данных.
9. Форматированный ввод/вывод.
10. Списки.
11. Выражения в списках.
12. Оператор `del`.
13. Использование списков, как стеков.
14. Использование списков, как очередей.
15. Операции сравнения для списков.
16. Диапазоны.
17. Кортежи. Отличие кортежей от словарей
18. Словари.
19. Оператор `if`. Особенности операторов сравнения.
20. Операторы цикла. Оператор `for`. Оператор `while`. Завершение цикла.
21. Продолжение цикла. Оператор `pass`.
22. Определение функции.
23. Пространство имен функции.
24. Передача параметров. Ключи.
25. Передача в функцию переменного числа аргументов.
26. Элементы функционального программирования.
27. Использование лямбда функций.
28. Функции работы со структурами данных.
29. Функция `map()`. Примеры применения
30. Функция `filter()`. Примеры применения
31. Функция `reduce()`. Примеры применения
32. Документирование функций.

### Типовые задачи

по дисциплине Программирование на ЯВУ

1. Малое предприятие в первый день работы выпустило  $P$  единиц товарной продукции. Каждый последующий день оно выпускало продукции на  $Q$  единиц больше, чем в предыдущий. Сколько дней потребуется предприятию, чтобы общее количество выпущенной продукции за все время работы впервые превысило запланированный объем  $T$ ?
2. Найти номер минимального по модулю элемента массива.

3. Числа Фибоначчи определяются как  $a(0)=1$ ,  $a(1)=1$ ,  $a(i)=a(i-1)+a(i-2)$ . Найти: а)  $n$ -ое число Фибоначчи. б) сумму первых  $n$  чисел Фибоначчи.
4. Найти в массиве те элементы, значение которых меньше среднего арифметического, взятого от всех элементов массива
5. Найти наименьшее общее кратное (НОК) двух натуральных чисел.
6. Даны два натуральных числа. Проверить, являются ли они дружественными.  
Примечание: дружественными числами называются два различных натуральных числа, для которых сумма всех собственных делителей первого числа равна второму числу и сумма всех собственных делителей второго числа равна первому числу. Например, 220 и 284 – пара дружественных чисел, потому что: Сумма собственных делителей 220:  $1 + 2 + 4 + 5 + 10 + 11 + 20 + 22 + 44 + 55 + 110 = 284$  Сумма собственных делителей 284:  $1 + 2 + 4 + 71 + 142 = 220$
7. Даны натуральные числа  $n$  и  $k$  ( $k$  не превышает  $n$ ). Вычислить число сочетаний из  $n$  по  $k$ .

$$C_n^k = \frac{n!}{k!(n-k)!}$$

8. Дано натуральное число  $n$  заранее неизвестной разрядности. Сформировать и вывести на экран число, представляющее собой реверсную запись  $n$ .
9. Даны два натуральных числа одинаковой десятичной разрядности. Сформировать из них третье число так, чтобы цифры первого числа стояли на нечетных местах третьего, а цифры второго – на четных. При этом порядки следования цифр сохраняются. Например, при вводе 1234 и 5678 программа должна выдать ответ 15263748
10. Дано натуральное число  $n$ . Проверить, является ли оно счастливым билетом.  
Примечание: вообще, в математике обычно рассматриваются счастливые билеты с четным количеством цифр, потому что у них можно явно выделить левую и правую половины одинаковой длины, суммы цифр которых и сравниваются. Однако мы несколько расширим это определение, полагая, что если число имеет нечетную длину, его центральную цифру можно отбросить, так как ее логично было бы прибавить к накапливаемым суммам обеих половин, что, собственно, не изменит отношения между ними. Например, число 14350 – счастливый билет, так как  $1 + 4 = 5 + 0$ , а центральную цифру мы отбросили.
11. Дано действительное число  $x$ . Вычислить значение экспоненциальной функции (то есть, показательной функции  $e^x$ , где  $e$  – математическая константа,  $e \approx 2,718281828459045$ ) в точке  $x$  с заданной точностью  $\epsilon$  с помощью ряда Тейлора:

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots$$

12. Дана последовательность символов длины  $n$  ( $n \geq 1$ ). Проверить баланс круглых скобок в этом выражении. Например, при вводе выражения  $(( ))( )$  программа должна сообщить о правильности расстановки скобок, а при вводе выражения  $(( ( ))$  – о неправильности.  
Примечание: сбалансированной скобочной записью называется символьное выражение, в котором каждой открывающей скобке соответствует закрывающая скобка правее и наоборот, каждой закрывающей скобке соответствует открывающая скобка левее.

13. Сгенерировать 20 случайных целых чисел в диапазоне от -5 до 4, записать их в ячейки массива. Посчитать сколько среди них положительных, отрицательных и нулевых значений. Вывести на экран элементы массива и посчитанные количества.
14. Дана последовательность натуральных чисел, ограниченная вводом нуля. Вывести на количество простых членов этой последовательности
15. Дано натуральное число  $n$ , а затем последовательность натуральных чисел, ограниченная вводом нуля. Вывести на экран произведение двузначных элементов этой последовательности, которые делятся на  $n$ .
16. В одномерном массиве хранится информация о количестве осадков, выпавших за каждый день сентября. Определить: а) в какие дни осадков не было; б) дни, когда количество осадков превысило среднее значение; в) общее количество осадков за месяц.
17. В одномерном массиве хранится информация о коммунальных платежах каждой из семей 20-квартирного дома за месяц. Определить: а) общую сумму платежей; б) номера квартир, которые не оплатили коммунальные услуги; в) номера квартир, платежи которых превысили заданное значение.
18. Вывести на экран матрицу  $5 \times 4$ , элементами которой являются целые случайные числа из интервала  $[30,70]$ . Определить номер минимального элемента в каждом четном столбце и номер максимального элемента в каждом нечетном столбце.
19. . Найти положительные элементы главной диагонали квадратной матрицы..
20. Преобразовать три введенные строки, чтобы после каждой цифры следовал символ `_'!`
21. Изменить введенную строку цифр, разместив сначала цифры занимающие нечетные места, потом четные.
22. Вывести на экран таблицу умножения (от 1 до 9).

## СЕВЕРО-КАВКАЗСКАЯ ГОСУДАРСТВЕННАЯ АКАДЕМИЯ

Кафедра Общей информатики

20\_\_\_\_ – 20\_\_\_\_ учебный год

Экзаменационный билет № 1

по дисциплине: Программирование на ЯВУ

для обучающихся направления подготовки 09.03.04 - Программная инженерия

1. Понятие алгоритма и программы. Способы записи алгоритмов.
2. Одномерные массивы. Понятия: массив, структуры данных, одномерный, двумерный массив. Объявление. Пример описания одномерного массива.
3. Задача: Задан одномерный массив  $A[1..20]$ . Найти сумму максимального и минимального элементов.

Зав. кафедрой

Эльканова Л.М.

## Контрольная работа

по дисциплине Программирование на ЯВУ

### Вариант 1

1. Основные элементы программирования
2. Программные средства
3. Задание № 1

Составить блок-схему и программу для вычисления значения функции по формуле (все переменные принимают действительные значения)

$Z =$

### Вариант 2

1. Требования, предъявляемые к программе
2. Нисходящее проектирование
3. Задание № 2

Вычислить количество и сумму отрицательных элементов массива  $X(5,4)$ .

### Вариант 3

1. Анализ и проектирование — разработка комплекса алгоритмов. Кодирование и компиляция. Тестирование и отладка.
2. Восходящее проектирование
3. Задание № 3

В целочисленном массиве  $A(30)$  найти наименьший из элементов с нечетными номерами.

### Вариант 4

1. Назначение алгоритмического языка Python.
2. Функциональное программирование
3. Задание № 4

Найти произведение элементов массива, индексы которых кратны 3.

### Вариант 5

1. Язык программирования Python
2. Переменные типы данных.
3. Задание № 5

Составить программу и блок-схему для вычисления значений функции  $F(x) = 2 \sin^2 x + 1$ , где  $x$  изменяется на отрезке  $[a, b]$  с шагом  $h$ .

### Вариант 6

1. Игра в планирование
2. Хакер. Защита информации. Информационная безопасность. Блок-схема.
3. Задание № 6

Составить блок-схему и программу для вычисления значения многочлена в точке  $x=0.5$ .

### Вариант 7

1. Планирование итераций
2. Логика в информатике.
3. Задание № 7

Разработайте приложение для определения общей стоимости покупки (в рублях и долларах). Название продуктов выбираются из списка, цены, количество, курс доллара вводит пользователь.

### Вариант 8

1. Организация программ разветвляющейся структуры.
2. Характеристики экспертной системы.
3. Задание № 8

Введите число от 1 до 7. Напечатайте соответствующий номеру цвет из цветов радуги.

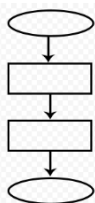
## Тестовые вопросы и задания

по дисциплине Программирование на ЯВУ

1. Язык программирования Python подходит для разработки:
  - a. Компьютерных и мобильных приложений
  - b. Аналитика и машинное обучение
  - c. Игр
  - d. Ничего из этого.

Формируемая компетенция ПК-2

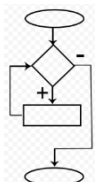
2. Назовите тип алгоритма:



- a. Разветвляющийся
- b. Линейный
- c. Циклический
- d. Смешанный

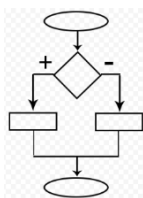
Формируемая компетенция ПК-2

3. Тип алгоритма \_\_\_\_\_



Формируемая компетенция ПК-2

4. Назовите тип алгоритма:



- a. Разветвляющийся
- b. Линейный
- c. Смешанный
- d. Циклический

Формируемая компетенция ПК-3

5. Переменная хранит в себе

- a. Имя
- b. Значение
- c. Тип
- d. Длину своего значения

Формируемая компетенция ПК-3

6. `int` обозначает \_\_\_\_\_ тип данных ?

Формируемая компетенция ПК-2

7. Выберите правильную запись оператора присваивания:

- a. `10 = x`
- b. `y = 7,8`
- c. `a = 5`
- d. `a == b + x`

Формируемая компетенция ПК-2

8. Укажите оператор ввода:

- a. `input()`
- b. `print()`
- c. `int()`
- d. `random()`

9. У переменной типа `bool` \_\_\_\_\_ возможных значений

Формируемая компетенция ПК-3

10. Укажите оператор, который здесь используется

If `n < 100`:

`B = N + A`



- a. Условный оператор
- b. Оператор присваивания
- c. Оператор сложения
- d. Оператор умножения

Формируемая компетенция ПК-3

11. Что лучше использовать для множественного ветвления?
- a. if – elif – else
  - b. Много if
  - c. if – else – elif
  - d. while

Формируемая компетенция ПК-2

12. Оператор цикла в языке Python:
- a. while
  - b. for
  - c. if
  - d. print

Формируемая компетенция ПК-2

13. Итерация цикла произойдет \_\_\_\_\_ раз (Итерация – единичное выполнение тела цикла)

```
TOTAL = 100
I = 0
WHILE I < 0:
    N = INT(INPUT()) TOTAL =
    TOTAL - NI = I + 1
PRINT("ОСТАЛОСЬ", TOTAL)
```

Формируемая компетенция ПК-3

14. Оператор break нужен для \_\_\_\_\_

Формируемая компетенция ПК-3

15. Где находятся параметры, а где аргументы функции?
- a. Параметры пишутся при объявлении функции, аргументы при вызове
  - b. Аргументы пишутся при объявлении функции, параметры при вызове
  - c. Это одно и то же!
  - d. У функции есть только параметры

Формируемая компетенция ПК-2

16. Функция `len()` возвращает \_\_\_\_\_

Формируемая компетенция ПК-2

17. Чтобы добавить модуль в программу используют
- `import math`
  - `import math()`
  - `import (math)`
  - `import.math`

Формируемая компетенция ПК-3

18. Python может работать на операционной системе...
- Windows
  - Linux
  - macOS
  - Ничего из этого

Формируемая компетенция ПК-3

19. От чего язык программирования называется «Питон»?
- В честь змеи
  - В честь ТВ-шоу
  - В честь игры
  - В честь блюда

Формируемая компетенция ПК-2

20. Создатель языка программирования Python
- Гвидо Ван Россум
  - Дэвид Паттерсон
  - Эрвин Дональд Кнут
  - Джеймс Артур Гослинг

Формируемая компетенция ПК-2

21. `a = 345`. Команда `print(a//100)` выведет значение \_\_\_\_\_

Формируемая компетенция ПК-3

22. Выберите циклический алгоритм
- `k = 0`  
`WHILE k < 10:`

```

    PRINT ("ПРИВЕТ")
    К += 1
b. a = int(input())
    b = int(input())
    c = int(input())
    s = a+b+c
    print(c)
c. a = int(input())
    if a > 0:
        PRINT(A)
    ELSE:
        PRINT(A)

```

ФОРМИРУЕМАЯ КОМПЕТЕНЦИЯ ПК-3

23. Правильно записан ввод числа с клавиатуры в строке под номером

<ol style="list-style-type: none"> <li>1. A = INT(INPUT))</li> <li>2. B=INPUT(INT())</li> <li>3. C=INT(INPUT())</li> <li>4. S=A+B+C</li> <li>5. PRINT(S)</li> </ol>
---

- a. 1
- b. 2
- c. 3
- d. 4
- e. 5

Формируемая компетенция ПК-2

24. Ошибка допущена в строке под номером

<ol style="list-style-type: none"> <li>1. A = INT(INPUT())</li> <li>2. IF A&gt;0:</li> <li>3. PRINT(A):</li> <li>4. ELSE</li> <li>5. PRINT(A)</li> </ol>
--

- a. 1
- b. 2
- c. 5
- d. 4

Формируемая компетенция ПК-2

25. Значение «else»?

- a. Так как
- b. Иначе
- c. Если
- d. Потому что

Формируемая компетенция ПК-3

26. Программа напишет слово «Пока» \_\_\_\_\_ раз

```
K=0
WHILE K<10:
    PRINT(«ПРИВЕТ»)
    K += 1
```

Формируемая компетенция ПК-3

27. Имеется кортеж вида T = (4, 2, 3). Какая из операций приведёт к тому, что имя T будет ссылаться на кортеж (1, 2, 3)?

- a. T[0] = 1
- b. T = (1) + T[1:]
- c. T = (1,) + T[1:]
- d. T.startswith(1)

Формируемая компетенция ПК-2

28. Встроенная функция enumerate() в Python используется для одновременного \_\_\_\_\_ по самим элементам и их индексам.

Формируемая компетенция ПК-2

29. Необходимо собрать и вывести все уникальные слова из строки рекламного текста. Какой из перечисленных типов данных Python подходит лучше всего?

- a. кортеж (tuple)
- b. список (list)
- c. множество (set)
- d. словарь (dict)

Формируемая компетенция ПК-3

30. В файлах `__init__.py` списки `__all__` используются для определения, что \_\_\_\_\_, когда каталог импортируется с помощью формы оператора `from*`. Для \_\_\_\_\_ переменных их имена начинаются с одиночного подчеркивания.

Формируемая компетенция ПК-2

## **Перечень лабораторных работ** по дисциплине Программирование на ЯВУ

### **Лабораторная работа №1 «Знакомство с Python»**

**Цель работы:** познакомиться со средой разработки Python. Изучить основные типы данных, команды ввода и вывода данных.

#### **Постановка задачи**

Установить Python <https://www.python.org/ftp/python/3.8.5/python-3.8.5.exe>

Написать программу, которая запрашивала бы у пользователя: Имя, Фамилия, Возраст, Место жительства - фамилия, имя ( "Ваши фамилия, имя?") - возраст ("Сколько Вам лет?") - место жительства ("Где вы живете?") После этого выводила бы три строки: "Ваши фамилия, имя" "Ваш возраст" "Вы живете в

### **Лабораторная работа №2: «Линейные программы»**

#### **Цель работы:**

Дать обучающимся практический навык в подготовке простой программы и в записи математических выражений на языке программирования Python.

#### **Постановка задачи**

Напишите программу для расчета по заданным формулам. Предварительно подготовьте тестовые примеры с помощью калькулятора или электронной таблицы Excel.

### **Лабораторная работа №3: «Разветвляющиеся вычислительные процессы»,**

#### **Цель работы:**

Дать обучающимся практический навык в использовании условных операторов ветвления на языке программирования Python. Работа состоит из двух заданий.

Следующий материал представлен в виде оформленной лабораторной работы.

#### **Задание 1**

##### **Постановка задачи**

Написать программу, которая по введенному значению аргумента вычисляет значение функции, заданной в виде графика. Параметры, необходимые для решения задания следует получить из графика и определить в программе.

#### **Задание 2**

##### **Постановка задачи**

Написать программу, которая определяет, попадает ли точка с заданными координатами в заштрихованную область. Точки на границе принадлежат области. Необходимые параметры получить из рисунка. Результат работы программы вывести в виде текстового сообщения: Попадает, Не попадает.

### **Лабораторная работа №4: «Организация циклов».**

#### **Цель работы:**

Дать обучающимся практический навык в использовании базовых конструкций структурного программирования - операторов цикла. Работа составлена из трёх заданий.

#### **Задание 1**

##### **Постановка задачи**

Вычислить и вывести на экран в виде таблицы значения функции, заданной графически (см. лабораторная работа № 2, задание 1), на интервале от Хнач до Хкон с шагом dx. Интервал и шаг задать таким образом, чтобы проверить все ветви программы. Таблицу снабдить заголовком и шапкой.

#### **Задание 2**

##### **Постановка задачи**

Для десяти выстрелов, координаты которых задаются генератором случайных чисел, вывести

текстовые сообщения о попадании в мишень (см. лабораторная работа № 2, задание 2).

### **Задание 3**

#### **Постановка задачи**

Вычислить и вывести на экран в виде таблицы значения функции интегрального синуса, заданной с помощью степенного ряда, на интервале от  $X_{нач}$  до  $X_{кон}$  с шагом  $dx$  с точностью  $\varepsilon$  – эпсилон.

Таблицу снабдить заголовком и шапкой. Каждая строка таблицы должна содержать значение аргумента, значение функции и количество просуммированных членов ряда.

### **Лабораторная работа №5: «Одномерные массивы»**

#### **Цель работы:**

Дать обучающимся практический навык в написании программ обработки одномерных массивов: поиск максимумов и минимумов, сортировка.

*Замечание:* Для организации массивов в Python могут использоваться разные варианты, например, списки или решения, предлагаемые в модулях `array`, `numpy`. В этих решениях могут быть готовые методы и функции, упрощающие программирование, например поиск максимума, минимума или сортировка. В этом руководстве предлагается использовать алгоритмы поиска решений без использования готовых методов и функций.

#### **Постановка задачи**

Сформировать одномерный список, состоящий из  $N$  вещественных чисел, полученных генератором случайных чисел. Количество элементов списка ( $N$ ) запрашивается у пользователя, но не превышает 30. Диапазон значений элементов от -5.0 до 5.0. Вычислить:

1. Первый и второй максимальные по модулю элементы списка.
2. Сумму элементов, модуль которых меньше единицы.
3. Все элементы, модуль которых превышает  $A_{max}$  обнулить.
3. Отсортировать список, сохраняя порядок ненулевых элементов. Равные нулю элементы разместить в конце списка.

### **Лабораторная работа №6: «Двумерные массивы и функции»**

#### **Цель работы:**

Дать обучающимся практический навык в написании программ обработки двумерных массивов с использованием функций.

#### **Постановка задачи**

Дан двумерный массив вещественных чисел.

1. Инициализировать элементы случайными числами в диапазоне (-10 -: 10).
2. Вычислить среднее и дисперсию ( $D$ ) по всем элементам массива.
3. Заменить, на среднее значение, те элементы массива, отклонение которых от среднего превышает  $\sigma$ , где  $\sigma = \sqrt{D}$ .
4. Пункты задания оформить в виде функций.

## **Практические работы**

по дисциплине Программирование на ЯВУ

### **Практическая работа: «Линейные программы»**

#### **Цель работы:**

Дать обучающимся практический навык в подготовке простой программы и в записи математических выражений на языке программирования Python.

#### **Постановка задачи**

Напишите алгоритм программы для расчета по заданным формулам. Предварительно

подготовьте тестовые примеры с помощью калькулятора или электронной таблицы Excel.

### **Практическая работа: «Разветвляющиеся вычислительные процессы»,**

#### **Цель работы:**

Дать обучающимся практический навык в использовании условных операторов ветвления на языке программирования Python. Работа состоит из двух заданий.

Следующий материал представлен в виде оформленной лабораторной работы.

#### **Задание 1**

##### **Постановка задачи**

Напишите алгоритм программы, которая по введенному значению аргумента вычисляет значение функции, заданной в виде графика. Параметры, необходимые для решения задания следует получить из графика и определить в программе.

#### **Задание 2**

##### **Постановка задачи**

Написать программу, которая определяет, попадает ли точка с заданными координатами в заштрихованную область. Точки на границе принадлежат области. Необходимые параметры получить из рисунка. Результат работы программы вывести в виде текстового сообщения: Попадает, Не попадает.

### **Лабораторная работа: «Организация циклов».**

#### **Цель работы:**

Дать обучающимся практический навык в использовании базовых конструкций структурного программирования - операторов цикла. Работа составлена из трёх заданий.

#### **Задание 1**

##### **Постановка задачи**

Вычислить и вывести на экран в виде таблицы значения функции, заданной графически (см. лабораторная работа № 2, задание 1), на интервале от  $X_{нач}$  до  $X_{кон}$  с шагом  $dx$ . Интервал и шаг задать таким образом, чтобы проверить все ветви программы. Таблицу снабдить заголовком и шапкой.

#### **Задание 2**

##### **Постановка задачи**

Для десяти выстрелов, координаты которых задаются генератором случайных чисел, вывести текстовые сообщения о попадании в мишень (см. лабораторная работа № 2, задание 2).

#### **Задание 3**

##### **Постановка задачи**

Вычислить и вывести на экран в виде таблицы значения функции интегрального синуса, заданной с помощью степенного ряда, на интервале от  $X_{нач}$  до  $X_{кон}$  с шагом  $dx$  с точностью  $\varepsilon$  – эпсилон.

Таблицу снабдить заголовком и шапкой. Каждая строка таблицы должна содержать значение аргумента, значение функции и количество просуммированных членов ряда.

### **Практическая работа: «Одномерные массивы»**

#### **Цель работы:**

Дать обучающимся практический навык в написании программ обработки одномерных массивов: поиск максимумов и минимумов, сортировка.

*Замечание:* Для организации массивов в Python могут использоваться разные варианты, например, списки или решения, предлагаемые в модулях `array`, `numpy`. В этих решениях могут быть готовые методы и функции, упрощающие программирование, например поиск максимума, минимума или сортировка. В этом руководстве предлагается использовать алгоритмы поиска решений без использования готовых методов и функций.

##### **Постановка задачи**

Сформировать одномерный список, состоящий из  $N$  вещественных чисел, полученных

генератором случайных чисел. Количество элементов списка ( $N$ ) запрашивается у пользователя, но не превышает 30. Диапазон значений элементов от -5.0 до 5.0. Вычислить:

1. Первый и второй максимальные по модулю элементы списка.
2. Сумму элементов, модуль которых меньше единицы.
3. Все элементы, модуль которых превышает  $A_{max}$  обнулить.
3. Отсортировать список, сохраняя порядок ненулевых элементов. Равные нулю элементы разместить в конце списка.

### **Практическая работа: «Двумерные массивы и функции»**

#### **Цель работы:**

Дать обучающимся практический навык в написании программ обработки двумерных массивов с использованием функций.

#### **Постановка задачи**

Дан двумерный массив вещественных чисел.

1. Инициировать элементы случайными числами в диапазоне (-10 -: 10).
2. Вычислить среднее и дисперсию ( $D$ ) по всем элементам массива.
3. Заменить, на среднее значение, те элементы массива, отклонение которых от среднего превышает  $\sigma$ , где  $\sigma = \sqrt{D}$ .
4. Пункты задания оформить в виде функций.



## **5. Методические материалы, определяющие процедуры оценивания компетенции**

### **5.1 Критерии оценивания результатов освоения дисциплины на экзамене:**

- оценка «зачтено» выставляется обучающемуся, если он демонстрирует знания основных теоретических положений в области алгоритмизации и современного программирования, включающего в себя методы проектирования, анализа и создания программных продуктов, основанных на использовании структурной и объектно-ориентированной методологии.

Готов и умеет составлять простые блок-схемы алгоритмов, решающих конкретную задачу.

Демонстрирует владение методами реализации алгоритмов в виде программ на конкретном языке программирования.

- оценка «не зачтено», если обучающийся, не знает основные теоретические положения в области алгоритмизации и современного программирования, включающего в себя методы проектирования, анализа и создания программных продуктов, основанных на использовании структурной и объектно-ориентированной методологии.

Не умеет и не готов составлять простые блок-схемы алгоритмов, решающих конкретную задачу.

Не владеет методами реализации алгоритмов в виде программ на конкретном языке программирования.

### **5.2 Критерии оценивания тестирования**

При тестировании все верные ответы берутся за 100%.

90%-100% отлично

70%-90% хорошо

50%-70% удовлетворительно

менее 50% неудовлетворительно

### **5.3 Критерии оценивания лабораторных работ**

Оценка «зачтено» выставляется обучающемуся, если лабораторная работа выполнена правильно и обучающийся ответил на все вопросы, поставленные преподавателем на защите.

Оценка «не зачтено» выставляется обучающемуся, если лабораторная работа выполнена не правильно или обучающийся не проявил глубоких теоретических знаний при защите работы

### **5.4 Критерии оценивания практических работ**

Критерии выполнения отчета на «отлично». Практическая работа выполнена полностью, без погрешностей и замечаний.

Критерии выполнения отчета на «неудовлетворительно» Практическая работа полностью выполнена.

Критерии оценки принятого отчета

- программный код не оптимален;
- использованы глобальные переменные;
- не на все вопросы получены верные ответы при защите работы;

Критерии дополнительных баллов за личностные качества

- работа выполнена верно с первого раза, на занятии по расписанию;
- соблюдение рекомендуемого стиля программирования;
- наличие, отсутствие или неполнота смысловых комментариев в программе.

### **5.5 Критерии оценки контрольных заданий (текущий контроль):**

- оценка **«отлично»** выставляется обучающемуся, если даны исчерпывающие и обоснованные ответы на все поставленные вопросы, правильно решены практические задания, при ответах выделялось главное, все теоретические положения умело увязывались с требованиями руководящих документов, ответы были четкими и краткими, а мысли излагались в логической последовательности, показано умение самостоятельно анализировать факты, события явления, процессы в их взаимосвязи и диалектическом развитии.
- оценка **«хорошо»** выставляется обучающемуся, если даны полные, достаточно обоснованные ответы на поставленные вопросы, правильно решены практические задания; при ответах не всегда выделялось главное, отдельные положения недостаточно увязывались с требованиями руководящих документов; ответы в основном были краткими, но не всегда четкими и по существу;
- оценка **«удовлетворительно»** выставляется обучающемуся, если даны в основном правильные ответы на все поставленные вопросы, но без должной глубины и обоснования; на уточняющие вопросы даны правильные ответы; при ответах не выделялось главное; ответы были многословными, нечеткими и без должной логической последовательности; на отдельные дополнительные вопросы не даны положительные ответы;
- оценка **«неудовлетворительно»** выставляется обучающемуся, если даны неправильные ответы на большинство вопросов; обучающийся путается в определениях и понятиях; не владеет практическими навыками решения задач.

### **5.6. Критерии оценивания качества контрольной работы**

- оценка **«зачтено»** выставляется обучающемуся, если работа выполнена без ошибок, и обучающемуся удалось защитить контрольную работу без наводящих вопросов;
- оценка **«не зачтено»** выставляется когда работа не выполнена по требованиям, или обучающийся не может защитить контрольную работу.