МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Санкт-Петербургский государственный экономический университет»

УТВЕРЖДАЮ
Проректор по учебной и методической работе
_____/Шубаева В.Г./
«20» июня 2018 г.

ЯЗЫКИ И МЕТОДЫ ПРОГРАММИРОВАНИЯ

Рабочая программа дисциплины

Направление подготовки	01.03.02 Прикладная математика и информатика
Направленность (профиль) программы	Прикладная математика и информатика в экономике и управлении
Уровень высшего образования	бакалавриат
Форма обучения	очная

Составитель:
_____/ ст. преподаватель Федоров Д.Ю.
_____/к.э.н., доцент Гниденко И.Г.

Санкт-Петербург
2018

СОДЕРЖАНИЕ

1. ЦЕЛЬ И ЗАДАЧИ ДИСЦИПЛИНЫ 3
2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ 3
3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ 3
4. ОБЪЕМ И СТРУКТУРА ДИСЦИПЛИНЫ 4
5. СОДЕРЖАНИЕ РАЗДЕЛОВ И ТЕМ ДИСЦИПЛИНЫ 5
6. ЗАНЯТИЯ СЕМИНАРСКОГО ТИПА 8
7. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ 9
7.1. Методические указания для обучающегося по освоению дисциплины 9
7.2. Организация самостоятельной работы 10
8. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ 11
9. РЕСУРСНОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ 11
9.1 Учебно-методическое и информационное обеспечение дисциплины 11
9.2 Материально-техническое обеспечение учебного процесса 12
10. ОСОБЕННОСТИ ОСВОЕНИЯ ДИСЦИПЛИНЫ ДЛЯ ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ 13
11. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ 13

1. ЦЕЛЬ И ЗАДАЧИ ДИСЦИПЛИНЫ

Цель (цели) освоения дисциплины: дать студентам необходимые знания в области языков и методов программирования, умения и навыки использования современных средств разработки приложений.

Задачи:

- знакомство с тенденциями развития языков и методов программирования;
- формирование умения использовать современные средства разработки приложений;
- выработка практических навыков работы с инструментарием в области разработки приложений.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Дисциплина Б1.Б.16 «Языки и методы программирования» относится к базовой части дисциплин Блока 1 и является обязательной для освоения обучающимся вне зависимости от направленности (профиля) программы.

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы представлены в таблице 3.1.

Таблица 3.1 – Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы

Код и наименование компетенции выпускника	Этапы формирования компетенций	Планируемые результаты обучения/индикаторы достижения компетенций (показатели освоения компетенции)
ОПК-3 способностью к разработке алгоритмических и программных решений в области системного и прикладного программирования, математических, информационных и имитационных моделей, созданию информационных ресурсов глобальных сетей, образовательного контента, прикладных баз данных, тестов и средств тестирования систем и средств на соответствие стандартам и исходным	Первый уровень (пороговый) (ОПК-3)-1	Знать: принципы программирования на языках высокого уровня, основные конструкции и типы данных современных языков программирования З1(II) (ОПК-3) Уметь: разрабатывать программы и реализовывать типовые алгоритмы на языках программирования высокого уровня, выполнять отладку и тестирование программ У1(II) (ОПК-3) Владеть: навыками программирования в современных средах разработки приложений В1(II) (ОПК-3)

Код и наименование компетенции выпускника	Этапы формирования компетенций	Планируемые результаты обучения/индикаторы достижения компетенций (показатели освоения компетенции)
требованиям		
ОПК-4 способностью решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	Первый уровень (пороговый) (ОПК-4)-1	Знать: современные подходы, основные этапы и требования к разработке приложений З1(I) (ОПК-4) Уметь: применять на практике современные технологии разработки программ У1(I) (ОПК-4) Владеть: навыками применения языков и методов программирования для решения стандартных задач профессиональной деятельности В1(I) (ОПК-4)

4. ОБЪЕМ И СТРУКТУРА ДИСЦИПЛИНЫ

Трудоемкость дисциплины составляет 11 зачетных единиц, 396 часов, из которых 72 часа самостоятельной работы обучающегося отводится на подготовку и защиту экзаменов.

Форма промежуточной аттестации: зачет - 2 семестр, экзамен - 3 семестр, экзамен – 4 семестр, курсовая работа - 3 семестр.

Распределение фонда времени по темам по очной форме обучения представлено в таблице 4.1

Таблица 4.1 – Распределение фонда времени по темам дисциплины (очная форма обучения)

Номер и наименование тем	Объем дисциплины (ак. часы)			
	Контактная работа			СРО
	ЗЛТ	ПЗ	ЛР	
1	*2*	*3*	*4*	*5*
РАЗДЕЛ 1. Основы программирования на языке Python				
Тема 1.1. История создания языков программирования.	2	4	-	6
Тема 1.2. Этапы разработки программ.	2	4	-	6
Тема 1.3. Выполнение программ на языке Python и среда разработки Jupyter Lab.	2	4	-	10
Тема 1.4. Вычисление математических выражений на языке Python (простой калькулятор).	2	4	-	6
Тема 1.5. Строки.	4	4	-	6
Тема 1.6. Операторы сравнения и инструкция if.	4	4	-	10
Тема 1.7. Подключение модулей.	4	4	-	6
Тема 1.8. Строковые методы.	2	4	-	4
Всего за семестр:	**22**	**32**	**-**	**54**

Номер и наименование тем	Объем дисциплины (ак. часы)			
	Контактная работа			СРО
	ЗЛТ	ПЗ	ЛР	
РАЗДЕЛ 2. Объектно-ориентированный подход в программировании на языке Python				
Тема 2.1. Списки.	4	4	-	8
Тема 2.2. Выполнение итераций на языке Python.	4	4	-	8
Тема 2.3. Дополнительные встроенные типы данных в Python.	4	8	-	8
Тема 2.4. Обработка исключений.	4	4	-	12
Тема 2.5. Работа с текстовыми файлами.	4	4	-	8
Тема 2.6. Работа с открытыми данными на языке Python.	4	4	-	12
Тема 2.7. Работа с JSON.	4	4	-	8
Тема 2.8. Создание собственных типов данных.	4	8	-	10
Тема 2.9. Иерархия наследования в Python.	4	4	-	8
Тема 2.10. Документирование и тестирование функций на языке Python.	6	4	-	8
Всего за семестр:	**42**	**48**	**-**	**90**
РАЗДЕЛ 3. Применение возможностей языка Python для решения задач				
Тема 3.1. Сравнение времени работы алгоритмов поиска.	2	3	-	2
Тема 3.2. Построение графиков с помощью модуля matplotlib.	2	3	-	2
Тема 3.3. Создание интерактивных виджетов в среде Jupyter Lab.	1	3	-	10
Тема 3.4. Создание графического интерфейса с помощью модуля tkinter.	2	3	-	2
Тема 3.5. Клиент-серверное программирование на языке Python.	2	3	-	2
Тема 3.6. Использование возможностей языка Python для обработки естественного языка.	2	3	-	10
Тема 3.7. Использование возможностей Python для обработки изображений.	2	3	-	2
Тема 3.8. Использование возможностей языка Python для решения задач анализа данных.	2	1	-	2
Тема 3.9. Применение языка Python в области искусственного интеллекта.	2	3	-	4
Тема 3.10. Интеграция языков программирования на примере написания расширений для языка Python.	2	1	-	10
Тема 3.11. Создание модуля Python на языке C.	2	3	-	4
Тема 3.12. Использование Cython для создания расширений языка Python.	1	3	-	4
Всего за семестр:	**22**	**32**	**-**	**54**
Всего по дисциплине:	**86**	**112**	**-**	**198**

*ЗЛТ – занятия лекционного типа, ПЗ – все виды занятий семинарского типа, кроме лабораторных работ, ЛР – лабораторные работы, СРО – самостоятельная работа обучающегося

5. СОДЕРЖАНИЕ РАЗДЕЛОВ И ТЕМ ДИСЦИПЛИНЫ

РАЗДЕЛ 1. Основы программирования на языке Python

Тема 1.1. История создания языков программирования

Назначение языков программирования, первые языки. Виды языков программирования. Связь с операционными системами. Этапы развития языков программирования. Перспективы развития языков программирования.

Тема 1.2. Этапы разработки программ

Представление программы как набора типов данных и операций над ними. Шаги при разработке программы. Подход при изучении языков программирования. Алгоритмы и типы данных как основа программы. Графический (веб) интерфейс как интерфейс доступа к возможностям программы.

Тема 1.3. Выполнение программ на языке Python и среда разработки Jupyter Lab

История создания среды Jupyter Lab, ее основные возможности. Интерактивные возможности IPython. Расширения среды разработки. Магические команды и связь с операционной системой. Перспективы развития среды разработки Jupyter Lab.

Тема 1.4. Вычисление математических выражений на языке Python (простой калькулятор)

Числовой тип данных. Операции над числами в Python. Целые и вещественные типы данных. Неизменяемый тип данных и особенности работы с переменными в Python. Модель памяти при работе с переменными. Неизменяемость типа данных как механизм оптимизации при работе с памятью.

Тема 1.5. Строки

Строковый тип данных в Python. Операции над строками. Срезы. Конкатенация строк. Строки как неизменяемый тип данных. Подходы к созданию новых строк.

Тема 1.6. Операторы сравнения и инструкция if

Логические операторы (и, или, не). Особенности реализации логических операторов в Python. Вычисление по короткой схеме. Булевый тип данных. Операции над булевым типом данных.

Тема 1.7. Подключение модулей

Использование модулей в Python для расширения возможностей языка. Создание собственных модулей. Варианты импортирования модулей. Примеры встроенных модулей Python. Обзор внешних модулей и процесс их установки.

Тема 1.8. Строковые методы

Отличие методов от функций. Строковые методы. Примеры популярных строковых методов. Особенности работы строковых методов.

РАЗДЕЛ 2. Объектно-ориентированный подход в программировании на языке Python

Тема 2.1. Списки

Списки как изменяемый тип данных. Примеры использования списков. Отличие списков от массивов в других языках программирования. Методы списков. Примеры популярных методов списков. Особенности работы методов списка.

Тема 2.2. Выполнение итераций на языке Python

Циклы в языке Python. Использование цикла for, использование цикла while. Пример бесконечного цикла. Обработка ввода пользователя. Особенности применения циклов при решении задач.

Тема 2.3. Дополнительные встроенные типы данных в Python

Кортежи, множества и словари в Python. Методы кортежей, множеств и словарей. Создание словаря на примере адресной книги. Обращение к несуществующему ключу словаря. Использование словаря для подсчета элементов в последовательности.

Тема 2.4. Обработка исключений

Генерация исключительных ситуаций. Встроенные типы ошибок в Python. Перехват исключений с помощью try except. Иерархия исключений. Особенности работы с исключениями в языке Python.

Тема 2.5. Работа с текстовыми файлами

Функции для работы с текстовыми файлами. Режимы открытия файла. Построчное чтение из файла. Обработка прочитанных данных. Запись в файл. Добавление в файл. Работа с несколькими файлами.

Тема 2.6. Работа с открытыми данными на языке Python

Понятие «открытые данные». Сайты, где размещаются открытые данные. Форматы файлов. Принцип работы с открытыми данными. Пример работы с открытыми данными «Сбербанка».

Тема 2.7. Работа с JSON

Преимущества формата JSON. Модуль json в Python. Запись и чтение данных в формате JSON. Примеры данных в формате JSON.

Тема 2.8. Создание собственных типов данных

Объектно-ориентированное программирование на языке Python. Создание классов и объектов. Методы объектов. Конструктор. Инициализация переменных в объекте. Специальные методы.

Тема 2.9. Иерархия наследования в Python

Принцип наследования в языке Python. Наследование методов. Вызов методов из базового класса. Особенности реализации наследования в языке Python.

Тема 2.10. Документирование и тестирование функций на языке Python

Принцип написание программ через предварительное написание тестов. Модуль doctest. Написание тестов для проверки правильности написания функций. Обзор модулей для тестирования кода на языке Python.

РАЗДЕЛ 3. Применение возможностей языка Python для решения задач

Тема 3.1. Сравнение времени работы алгоритмов поиска

Реализация алгоритмов поиска на языке Python. Нотация О-большое. Время выполнения алгоритмов. Использование возможностей Jupyter Lab для сравнения времени выполнения программ.

Тема 3.2. Построение графиков с помощью модуля matplotlib

Модуль matplotlib. Построение графиков функций. Определение осей для построения графика. Особенности работы с модулем matplotlib. Интерфейс MATLAB при работе с модулем matplotlib.

Тема 3.3. Создание интерактивных виджетов в среде Jupyter Lab

Исследование возможностей Jupyter Lab для создание интерактивных виджетов. Пример решения задачи вычисления факториала. Обзор виджетов Jupyter Lab.

Тема 3.4. Создание графического интерфейса с помощью модуля tkinter

Основные возможности модуля tkinter. Создание главного окна. Виджеты. Виджет Lable. Фреймы и виджеты. Встроенные переменные модуля tkinter. Виджет Entry. Разработка оконных приложений.

Тема 3.5. Клиент-серверное программирование на языке Python

История создания сети Интернет. Стек протоколов TCP/IP. Интерфейс сокетов в Python. Разработка клиент-серверного приложения на языке Python.

Тема 3.6. Использование возможностей языка Python для обработки естественного языка

Модуль nltk. Нормализация текста. Стемминг. Разбиение текста на отдельные элементы с учетом знаков препинания. Использование возможностей классификации и кластеризации при анализе текстовой информации.

Тема 3.7. Использование возможностей Python для обработки изображений

Модуль pillow. Возможности модуля для обработки изображений. Обзор модулей для модификации изображений на языке Python.

Тема 3.8. Использование возможностей языка Python для решения задач анализа данных

Модуль numpy, pandas. Основные возможности модуля для анализа данных. Представление данных в виде таблиц (Data Frames). Работа с таблицами, фильтрация. Пример решения задачи анализа данных (кластеризация, классификация).

Тема 3.9. Применение языка Python в области искусственного интеллекта

Распознавание образов на языке Python, нейронные сети. Обзор модулей языка Python для решения задач в области ИИ.

Тема 3.10. Интеграция языков программирования на примере написания расширений для языка Python

Обзор возможности для расширения языка Python. Основные подходы для разработки расширений. Преимущества и недостатки существующих подходов.

Тема 3.11. Создание модуля Python на языке С

История языка программирования Си. Основные конструкции языка Си. Преимущества языка Си. API-C для разработки расширений Python.

Тема 3.12. Использование Cython для создания расширений языка Python

Возможности языка Cython. Пример создания расширения для языка Python. Преимущества языка Cython для создания расширений Python.

6. ЗАНЯТИЯ СЕМИНАРСКОГО ТИПА

Таблица 6.1 – Практические занятия/Семинарские занятия / Лабораторные работы

№ темы	Тема занятия	Вид занятия / Оценочное средство
1	2	3
1.1	История создания языков программирования	ПЗ/Решение практических задач
1.2	Этапы разработки программ	ПЗ/Решение практических задач
1.3	Выполнение программ на языке Python и среда разработки Jupyter Lab	ПЗ/Решение практических задач/Индивидуальное задание 1
1.4	Вычисление математических выражений на языке Python (простой калькулятор)	ПЗ/Решение практических задач
1.5	Строки	ПЗ/Решение практических задач
1.6	Операторы сравнения и инструкция if	ПЗ/Решение практических задач/Индивидуальное задание 2
1.7	Подключение модулей	ПЗ/Решение практических задач
1.8	Строковые методы	ПЗ/Решение практических задач
2.1	Списки	ПЗ/Решение практических задач
2.2	Выполнение итераций на языке Python	ПЗ/Решение практических задач
2.3	Дополнительные встроенные типы данных в Python	ПЗ/Решение практических задач
2.4	Обработка исключений	ПЗ/Решение практических задач/Индивидуальное задание 3
2.5	Работа с текстовыми файлами	ПЗ/Решение практических задач

№ темы	Тема занятия	Вид занятия / Оценочное средство
2.6	Работа с открытыми данными на языке Python	ПЗ/Решение практических задач/Индивидуальное задание 4
2.7	Работа с JSON	ПЗ/Решение практических задач
2.8	Создание собственных типов данных	ПЗ/Решение практических задач/Индивидуальное задание 5
2.9	Иерархия наследования в Python	ПЗ/Решение практических задач
2.10	Документирование и тестирование функций на языке Python	ПЗ/Решение практических задач
3.1	Сравнение времени работы алгоритмов поиска	ПЗ/Решение практических задач
3.2	Построение графиков с помощью модуля matplotlib	ПЗ/Решение практических задач
3.3	Создание интерактивных виджетов в среде Jupyter Lab	ПЗ/Решение практических задач/Индивидуальное задание 6
3.4	Создание графического интерфейса с помощью модуля tkinter	ПЗ/Решение практических задач
3.5	Клиент-серверное программирование на языке Python	ПЗ/Решение практических задач
3.6	Использование возможностей языка Python для обработки естественного языка	ПЗ/Решение практических задач/Индивидуальное задание 7
3.7	Использование возможностей Python для обработки изображений	ПЗ/Решение практических задач
3.8	Использование возможностей языка Python для решения задач анализа данных	ПЗ/Решение практических задач
3.9	Применение языка Python в области искусственного интеллекта	ПЗ/Решение практических задач
3.10	Интеграция языков программирования на примере написания расширений для языка Python	ПЗ/Решение практических задач/Индивидуальное задание 8
3.11	Создание модуля Python на языке C	ПЗ/Решение практических задач
3.12	Использование Cython для создания расширений языка Python	ПЗ/Решение практических задач

*ПЗ – практические занятия, СЗ – семинарские занятия, ЛР – лабораторные работы

7. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ

7.1. Методические указания для обучающегося по освоению дисциплины

Для формирования четкого представления об объеме и характере знаний и умений, которыми надо будет овладеть по дисциплине в самом начале учебного курса обучающийся должен ознакомиться с учебно-методической документацией:

– рабочей программой дисциплины: с целями и задачами дисциплины, ее связями с другими дисциплинами образовательной программы, перечнем знаний и умений, которыми в процессе освоения дисциплины должен владеть обучающийся,

– порядком проведения текущего контроля успеваемости и

промежуточной аттестации;

– графиком консультаций преподавателей кафедры.

Систематическое выполнение учебной работы на занятиях лекционных и семинарских типов, а также выполнение самостоятельной работы позволит успешно освоить дисциплину.

В процессе освоения дисциплины обучающимся следует:

– слушать, конспектировать излагаемый преподавателем материал;

– ставить, обсуждать актуальные проблемы курса, быть активным на занятиях;

– задавать преподавателю уточняющие вопросы с целью уяснения теоретических положений;

– выполнять задания практических занятий полностью и установленные сроки.

При затруднениях в восприятии материала следует обратиться к основным литературным источникам. Если разобраться в материале не удалось, то обратится к лектору (по графику его консультаций) или к преподавателю на занятиях семинарского типа.

Обучающимся, пропустившим занятия (независимо от причин), не имеющим письменного решения задач или не подготовившимся к данному занятию, рекомендуется не позже, чем в 2-недельный срок явиться на консультацию к преподавателю и отчитаться по теме.

7.2. Организация самостоятельной работы

Под самостоятельной работой обучающихся понимается планируемая работа обучающихся, направленная на формирование указанных компетенций, выполняемая во внеаудиторное время по заданию и при методическом руководстве преподавателя, без его непосредственного участия.

Методическое обеспечение самостоятельной работы при наличии обучающихся лиц с ограниченными возможностями представляется в формах, адаптированных к ограничениям их здоровья.

Виды самостоятельной работы по дисциплине представлены в таблице 7.2.1.

Таблица 7.2.1 – Организация самостоятельной работы обучающегося

№ темы	Вид самостоятельной работы
1	*2*
1.1-1.8	Подготовка к проверке знаний по теме. Подготовка к зачету.
1.3	Подготовка к выполнению и защите контрольной точки 1. Подготовка к проверке знаний по теме. Подготовка к зачету.
1.6	Подготовка к выполнению и защите контрольной точки 2. Подготовка к проверке знаний по теме. Подготовка к зачету.
2.1-2.10	Подготовка к проверке знаний по теме. Подготовка к экзамену.
2.3	Сбор материала для выполнения курсовой работы.
2.4	Подготовка к выполнению и защите контрольной точки 1. Подготовка к проверке знаний по теме. Подготовка к экзамену.
2.6	Подготовка к выполнению и защите контрольной точки 2. Подготовка к проверке знаний по теме. Подготовка к экзамену.
2.8	Подготовка к выполнению и защите контрольной точки 3. Подготовка к проверке знаний по теме. Подготовка к экзамену.
2.9	Подготовка к экзамену. Подготовка к защите курсовой работы.
3.1-3.12	Подготовка к проверке знаний по теме. Подготовка к экзамену.

№ темы	Вид самостоятельной работы
3.3	Подготовка к выполнению и защите контрольной точки 1. Подготовка к проверке знаний по теме. Подготовка к экзамену.
3.6	Подготовка к выполнению и защите контрольной точки 2. Подготовка к проверке знаний по теме. Подготовка к экзамену.
3.10	Подготовка к выполнению и защите контрольной точки 3. Подготовка к проверке знаний по теме. Подготовка к экзамену.

Каждый вид СРО, указанный в таблице 7.2.1 обеспечен методическими материалами.

8. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

В преподавании дисциплины «*Языки и методы программирования*» используются разнообразные образовательные технологии как традиционные, так и с применением активных и интерактивных методов обучения.

Активные и интерактивные методы обучения:

К интерактивным методам обучения относится выполнение индивидуальных заданий на ПК в ходе практических занятий и самостоятельной работы по темам 1.3, 1.6, 2.4, 2.6, 2.8, 3.3, 3.6, 3.10.

Выполнение индивидуальных заданий на ПК предполагает выполнение индивидуальных заданий по изучению пакетов прикладных программ, языков программирования, информационно-коммуникационных технологий (как правило, организационной формой таких занятий являются лабораторные работы).

9. РЕСУРСНОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

9.1 Учебно-методическое и информационное обеспечение дисциплины

Таблица 9.1.1 – Учебно-методическое обеспечение дисциплины

Библиографическое описание издания (автор, заглавие, вид, место и год издания, кол. стр.)	**Основная/ дополнительная литература**	**Книгообеспеченность**	
		Кол-во. экз. в библ. СПбГЭУ	**Электронные ресурсы**
Федоров, Д. Ю. Языки и методы программирования : учебное пособие / Д.Ю. Федоров .— Санкт-Петербург : Изд-во СПбГЭУ, 2019 .— 74 с. — Сведения доступны также по Интернету: opac.unecon.ru .	Основная	35	ЭБ OPAC.UNECON.RU
Федоров Д.Ю. Программирование на языке высокого уровня Python : учебное пособие . — 2-е изд., перераб. и доп. — Электрон. дан. — М. : Издательство Юрайт, 2019. — 161 с.	Основная	–	ЭБС Юрайт
Гниденко И.Г Технологии и методы программирования : учебное пособие / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров.. — Электрон. дан. — Москва : Издательство Юрайт, 2019 .— 235 с.	Основная	–	ЭБС Юрайт

Гниденко И.Г. Технологии и методы программирования : учебное пособие / И.Г. Гниденко, Д.Ю. Федоров . – Санкт-Петербург : Изд-во СПбГЭУ, 2017 . – 58 с. – Сведения доступны также по Интернету: opac.unecon.ru.	Дополнительная	25	ЭБ OPAC.UNECON.RU
Федоров Д. Ю. Языки программирования : учебное пособие . – Санкт-Петербург: Изд-во СПбГЭУ, 2018. – 84 с. – Сведения доступны также по Интернету: opac.unecon.ru.	Дополнительная	35	ЭБ OPAC.UNECON.RU

Таблица 9.1.2 – Перечень современных профессиональных баз данных (СПБД)

№	Наименование СПБД
1	Электронная библиотека Grebennikon.ru – www.grebennikon.ru
2	Научная электронная библиотека eLIBRARRY – www.elibrary.ru
3	Научная электронная библиотека КиберЛеника – www.cyberleninka.ru
4	База данных ПОЛПРЕД Справочники – www.polpred.com
5	База данных OECD Books, Papers & Statistics на платформе OECD iLibrary – www.oecd-ilibrary.org

Таблица 9.1.3 – Перечень информационных справочных систем (ИСС)

№	Наименование ИСС
1	Справочная правовая система КонсультантПлюс (инсталлированный ресурс СПбГЭУ или www.consultant.ru)
2	Справочная правовая система «ГАРАНТ» (инсталлированный ресурс СПбГЭУ или www.garant.ru)
3	Информационно-справочная система «Кодекс» (инсталлированный ресурс СПбГЭУ или www.kodeks.ru)
4	Электронная библиотечная система BOOK.ru - www.book.ru
5	Электронная библиотечная система ЭБС ЮРАЙТ – www.urait.ru
6	Электронно-библиотечная система ЗНАНИУМ (ZNANIUM) – www.znanium.com
7	Электронная библиотека СПбГЭУ– opac.unecon.ru

9.2 Материально-техническое обеспечение учебного процесса

Для реализации данной дисциплины имеются специальные помещения для проведения занятий лекционного типа, занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы.

Таблица 9.2.1 – Перечень программного обеспечения (ПО)

№ п/п	Наименование ПО
1	Microsoft Windows Professional (КОНТРАКТ № 244/20 «26» июня 2020 г)
2	Microsoft Office Professional (КОНТРАКТ № 244/20 «26» июня 2020 г)
3	7-Zip (freeware)
4	Wolfram Mathematica 12 (Сублицензионный договор № 328/20 от 20.08.2020г.)
5	Python 3.7.4 (freeware)

10. ОСОБЕННОСТИ ОСВОЕНИЯ ДИСЦИПЛИНЫ ДЛЯ ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

Обучение обучающихся с ограниченными возможностями здоровья при необходимости осуществляется на основе адаптированной рабочей программы с использованием специальных методов обучения и дидактических материалов, составленных с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся (обучающегося).

В целях освоения учебной программы дисциплины инвалидами и лицами с ограниченными возможностями здоровья Университет обеспечивает:

– для инвалидов и лиц с ограниченными возможностями здоровья по зрению: размещение в доступных для обучающихся, являющихся слепыми или слабовидящими, местах и в адаптированной форме справочной информации о расписании учебных занятий; присутствие ассистента, оказывающего обучающемуся необходимую помощь; выпуск альтернативных форматов методических материалов (крупный шрифт или аудиофайлы);

– для инвалидов и лиц с ограниченными возможностями здоровья по слуху: надлежащими звуковыми средствами воспроизведение информации;

– для инвалидов и лиц с ограниченными возможностями здоровья, имеющих нарушения опорно-двигательного аппарата: возможность беспрепятственного доступа обучающихся в учебные помещения, туалетные комнаты и другие помещения кафедры, а также пребывание в указанных помещениях.

Образование обучающихся с ограниченными возможностями здоровья может быть организовано как совместно с другими обучающимися, так и в отдельных группах или в отдельных организациях.

11. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ОБУЧАЮЩИХСЯ ПО ДИСЦИПЛИНЕ

Фонд оценочных средств для проведения аттестации уровня сформированности компетенций обучающихся по дисциплине оформляется отдельным документом и является приложением к рабочей программе дисциплины (модуля).