МИНОБРНАУКИ РОССИИ

федеральное государственное бюджетное образовательное учреждение высшего образования

«Санкт-Петербургский государственный экономический университет»

|  |  |
| --- | --- |
|  | УТВЕРЖДАЮ  Проректор по образовательной деятельности  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.Г. Шубаева  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_\_г. |

***Технологии программирования***

**Рабочая программа дисциплины**

|  |  |
| --- | --- |
| Направление подготовки/ *Специальность* | *38.03.05 Бизнес-информатика* |
| Направленность (профиль) программы/  *Специализация* | *Деловая аналитика* |
| Уровень высшего образования | *Бакалавриат* |
| Форма обучения | *очная* |
| Год набора | *2025* |

Составитель*(и)*:

|  |
| --- |
| к.т.н, Никифоров Сергей Геннадиевич |

|  |  |  |  |
| --- | --- | --- | --- |
| Часов по учебному плану | 216 | **Виды контроля в семестрах:**   |  | | --- | | Экзамен: семестр 3 | |
| в том числе: |  |
| контактная работа | 112 |
| самостоятельная работа | 68 |
| практическая подготовка | 0 |
| часов на контроль | 36 |

**Распределение часов дисциплины:**

|  |  |
| --- | --- |
| Семестр: | 3 |
| Вид занятий | Часы |
| Лекционные занятия | 42 |
| Практические занятия | 70 |
| Лабораторные работы |  |
| **Итого аудиторных часов** | **112** |
| Самостоятельная работа | 68 |
| Часы на контроль | 36 |
| **Итого академических часов** | **216** |
| **Общая трудоемкость в зачетных единицах** | **6** |

Санкт-Петербург

2025

**СОДЕРЖАНИЕ**

[**1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ** 3](#_Toc83656871)

[**2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ** 3](#_Toc83656872)

[**3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ** 3](#_Toc83656873)

[**4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ\*** 4](#_Toc83656874)

[**5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ** 7](#_Toc83656875)

[**5.1 Рекомендуемая литература** 7](#_Toc83656876)

[**5.2 Перечень лицензионного и свободно распространяемого программного обеспечения, в т.ч. отечественного производства** 7](#_Toc83656877)

[**5.3 Перечень информационных справочных систем (ИСС) и современных профессиональных баз данных (СПБД)** 8](#_Toc83656878)

[**6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ** 8](#_Toc83656879)

[**7. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩЕГОСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ** 9](#_Toc83656880)

[**8. ОСОБЕННОСТИ ОСВОЕНИЯ ДИСЦИПЛИНЫ ДЛЯ ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ** 10](#_Toc83656881)

[**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ** 12](#_Toc83656882)

[**1.1 Контрольные вопросы и задания к промежуточной аттестации** 12](#_Toc83656883)

[**1.2 Темы письменных работ** 16](#_Toc83656884)

[**1.3 Контрольные точки** 16](#_Toc83656885)

[**1.4 Другие объекты оценивания** 16](#_Toc83656886)

[**1.5 Самостоятельная работа обучающегося** 16](#_Toc83656887)

[**1.6 Шкала оценивания результата** 16](#_Toc83656888)

# **1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ**

|  |  |
| --- | --- |
| **Цель:** | Освоение современных технологий разработки программного обеспечения, изучение языка программирования Python и интегрированных сред программирования. |

# **2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ**

Дисциплина Б1.О Технологии программирования относится к обязательной части Блока 1.

# **3.** **ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ**

| **Код и наименование компетенции выпускника** | **Код и наименование индикатора достижения компетенций** | **Планируемые результаты обучения по дисциплине** |
| --- | --- | --- |
| ОПК-3 - Способен управлять процессами создания и использования продуктов и услуг в сфере информационно-коммуникационных технологий, в том числе разрабатывать алгоритмы и программы для их практической реализации; | ОПК-3.2 - Способен управлять процессом разработки алгоритмов и программ | Знать: модели жизненного цикла, технологии и процессы разработки программного обеспечения (ПО). Особенности формулировки функциональных требований к ПО.  Уметь: выявлять, анализировать, документировать требования к программным средствам.  Владеть: инструментами визуального моделирования и прототипирования. |
| ОПК-6 - Способен выполнять отдельные задачи в рамках коллективной научно-исследовательской, проектной и учебно-профессиональной деятельности для поиска, выработки и применения новых решений в области информационно-коммуникационных технологий. | ОПК-6.3 - Использует современные инструменты для поиска, выработки и применения новых решений в области информационно-коммуникационных технологий | Знать: особенности формулировки пользовательских, функциональных и нефункциональных требований к ПО. Технологии визуального моделирования и прототипирования.  Уметь: разрабатывать требования к программному обеспечению.  Владеть: инструментами визуального моделирования и прототипирования. |
| ПК-3 - Способен применять методы, технологии и инструменты проектирования систем малого, среднего и крупного масштаба и сложности | ПК-3.1 - Способен использовать методы и технологии программирования | Знать: синтаксис, основные типы данных, базовые конструкции и библиотеки языка программирования Python  Уметь: - разрабатывать алгоритмы решения различных задач, - использовать интегрированные среды разработки и облачные платформы для разработки и отладки программного обеспечения.  Владеть: навыками чтения, написания, отладки и тестирования программ на языке программирования Python. |

# **4.** **СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ\***

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Номер и наименование тем и/или разделов/тем** | **Содержание дисциплины** | | **Объем дисциплины**  **(академические часы)** | | | | |
| **Контактная работа** | | | | **СРО** |
| **ЗЛТ** | | **ПЗ** | **ЛР** |
| Тема 1. Язык программирования Python. Общие сведения. Установка. Основы работы в среде Colab. | Python. Общие сведения. Установка. Основы работы в среде Colab. | | 2 | | 2 |  | 1 |
| Тема 2. Python. Синтаксис. Переменные. Основные типы данных. Ошибки и исключения. | Python. Синтаксис. Переменные. Основные типы данных. Ошибки и исключения. | | 2 | | 2 |  | 1 |
| Тема 3. Python. Числа. Арифметические операции. Перевод чисел в различные системы счисления. | Python. Числа. Арифметические операции. Перевод чисел в различные системы счисления. | | 2 | | 2 |  | 2 |
| Тема 4. Python. Булевы переменные. Логические выражения, Условные операторы. | Python. Булевы переменные. Логические выражения, Условные операторы. | | 2 | | 4 |  | 2 |
| Тема 5. Python. Функции. | Python. Функции. Лямбда-функции. | | 2 | | 4 |  | 4 |
| Тема 6. Python. Строки. Операции над строками. Методы строк. | Python. Строки. Операции над строками. Индексы, срезы. Методы строк. | | 2 | | 4 |  | 4 |
| Тема 7. Python. Циклы | Python. Циклы for и while | | 2 | | 4 |  | 4 |
| Тема 8. Python. Массивы: Списки. Кортежи. Матрицы. | Python. Массивы: Списки. Кортежи. Матрицы. | | 2 | | 4 |  | 4 |
| Тема 9. Python. Генерация случайных чисел. Модуль Random. Модуль timeit, измерение времени работы программы. Модуль array. Массивы в python. | Python. Генерация случайных чисел. Модуль Random. Модуль timeit, измерение времени работы программы. Модуль array. Массивы в python. | | 2 | | 4 |  | 4 |
| Тема 10. Pyton. Модуль matplotlib для построения и отображения графиков. Построение гистограмм случайных чисел. | Pyton. Модуль matplotlib для построения и отображения графиков. Построение гистограмм случайных чисел. | | 2 | | 2 |  | 4 |
| Тема 11. Python. Работа с файлами. Плоские файлы формата SCV. Сериализация. | Python. Работа с файлами. Плоские файлы формата SCV. Сериализация. | | 2 | | 4 |  | 4 |
| Тема 12. Python. Библиотека Numpy, установка, основные типы данных. Создание массивов, свойства массивов, операции над массивами. | Библиотека Numpy, установка, основные типы данных. Создание массивов, свойства массивов, операции над массивами. | | 2 | | 4 |  | 4 |
| Тема 13. Python. Модули. пакеты. Оценка производительности. О-нотация. | Python. Модули. пакеты. Оценка производительности. О-нотация. | | 2 | | 2 |  | 2 |
| Тема 14. Python. Алгоритмы поиска и сортировки. Линейный и бинарный поиск. Ограничения бинарного поиска. Модуль Bisect. Пузырьковая сортировка. Оптимизация пузырьковой сортировки. Сортировки слиянием и вставками. | Python. Алгоритмы поиска и сортировки. Линейный и бинарный поиск. Ограничения бинарного поиска. Модуль Bisect. Пузырьковая сортировка. Оптимизация пузырьковой сортировки. Сортировки слиянием и вставками. | | 2 | | 6 |  | 6 |
| Тема 15. Python. Словари. Множества, неизменяемые множества. | Python. Словари. Множества, неизменяемые множества. | | 2 | | 4 |  | 4 |
| Тема 16. Программирование Python. Работа с файлами в формате JSON. Работа с геоданными в формате GEOJSON. | Программирование Python. Работа с файлами в формате JSON. Работа с геоданными в формате GEOJSON. | | 2 | | 4 |  | 4 |
| Тема 17. Программирование Python. Обработка исключений. \* распаковка, \*args, \*\*kwargs Множественные аргументы функций. | Программирование Python. Обработка исключений. \* распаковка, \*args, \*\*kwargs Множественные аргументы функций. | | 2 | | 4 |  | 4 |
| Тема 18. Программирование Python. Классы и объектно-ориентированное программирование(ООП). | Программирование Python. Классы и объектно-ориентированное программирование(ООП). | | 2 | | 4 |  | 4 |
| Тема 19. Python. Модуль requests. | Python. Модуль requests. | | 2 | | 6 |  | 4 |
| Тема 20. Основные модели жизненного цикла и методологии разработки ПО. Каскадная и гибкая методологии. | Процессы разработки ПО. Жизненный цикл и процессы разработки ПО. Осн модели жизненного цикла и методологии разработки ПО. Каскадная и гибкая методологии. | | 2 | |  |  | 1 |
| Тема 21. Требования. Классификация. Приемы сбора требований. | Требования. Классификация. Приемы сбора требований. Техническое задание и спецификация. Анализ требований и моделирование системы. Графические нотации. Концептуальная схема и DFD-диаграммы, нотации UML, BPMN. | | 2 | |  |  | 1 |
| **Контроль:** | | | | | | | **36** |
| **Всего по дисциплине:** | | **42** | | **70** | |  | **68** |

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

# **5. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

## **5.1 Рекомендуемая литература**

|  |  |
| --- | --- |
| **Библиографическое описание издания (автор, заглавие, вид, место и год издания, кол. стр.)** | **Электронные ресурсы** |
| Федоров, Д. Ю. Программирование на языке высокого уровня Python : учебное пособие для вузов / Д. Ю. Федоров. — 5-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 227 с. — (Высшее образование). — ISBN 978-5-534-17323-9. — URL : https://urait.ru/bcode/539651 | <https://urait.ru/bcode/539651> |
| Чернышев, С. А. Основы программирования на Python : учебное пособие для вузов / С. А. Чернышев. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 349 с. — (Высшее образование). — ISBN 978-5-534-17139-6. — URL : https://urait.ru/bcode/544190 | [https://urait.ru/bcode/544190](%20https://urait.ru/bcode/544190) |
| Северанс, Ч. Python для всех : практическое руководство / Ч. Северанс ; пер. с англ. А. В. Снастина. - Москва : ДМК Пресс, 2022. - 262 с. - ISBN 978-5-93700-104-7. - Текст : электронный. - URL: https://znanium.ru/catalog/product/2155875 | [https://znanium.ru/catalog/product/2155875](https://znanium.ru/catalog/product/2155875%20) |
| Маккинни, У. Python и анализ данных. Первичная обработка данных с применением pandas, NumPy и Jupiter / У. Маккинни ; пер. А. А. Слинкина. - 3-е изд. -Москва : ДМК Пресс, 2023. - 536 с. - ISBN 978-5-93700-174-0. - Текст : электронный. - URL: https://znanium.ru/catalog/product/2150530 | <https://znanium.ru/catalog/product/2150530> |
| Титов, А. Н. Визуализация данных в Python. Работа с библиотекой Matplotlib : учебно-методическое пособие / А. Н. Титов, Р. Ф. Тазиева ; Минобрна-уки России, Казан. нац. исслед. технол. ун-т. - Казань : Изд-во КНИТУ, 2022. - 92 с. - ISBN 978-5-7882-3176-1. - Текст : электронный. - URL: https://znanium.com/catalog/product/2069267 | <https://znanium.com/catalog/product/2069267> |

## **5.2 Перечень лицензионного и свободно распространяемого программного обеспечения, в т.ч. отечественного производства**

|  |
| --- |
| - 7-Zip |
| - ОС Альт образование 10 |
| - Python |
| - pandas |
| - LibreOffice Base |
| - LibreOffice Calc |
| - LibreOffice Writer |

## **5.3 Перечень информационных справочных систем (ИСС) и современных профессиональных баз данных (СПБД)**

|  |  |
| --- | --- |
| **№** | **Наименование СПБД/ ИСС** |
| 1. | Электронная библиотека Grebennikon.ru – [www.grebennikon.ru](http://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](http://www.polpred.com) |
| 5. | База данных OECD Books, Papers & Statistics на платформе OECD iLibrary  [www.oecd-ilibrary.org](http://www.oecd-ilibrary.org) |
| 6. | Справочная правовая система КонсультантПлюс (инсталлированный ресурс  СПбГЭУ или www.consultant.ru) |
| 7. | Справочная правовая система «ГАРАНТ» (инсталлированный ресурс СПбГЭУ или www.garant.ru) |
| 8. | Информационно-справочная система «Кодекс» (инсталлированный ресурс  СПбГЭУ или www.kodeks.ru) |
| 9. | Электронная библиотечная система BOOK.ru - www.book.ru |
| 10. | Электронная библиотечная система ЭБС ЮРАЙТ – www.urait.ru |
| 11. | Электронно-библиотечная система ЗНАНИУМ (ZNANIUM) – [www.znanium.com](http://www.znanium.com) |
| 12. | Электронная библиотека СПбГЭУ– opac.unecon.ru |

# **6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ**

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.

Помещения оснащены оборудованием и техническими средствами обучения.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду университета.

|  |  |
| --- | --- |
| **Наименование учебных аудиторий, перечень** | **Адрес (местоположение) учебных аудиторий** |
| Ауд. 0003 Компьютерный класс (для проведения практических занятий, курсового проектирования (выполнения курсовых работ) с применением вычислительной техники). Оборудован мультимедийным комплексом. Оборудован мультимедийным комплексом.Специализированная мебель и оборудование: Учебная мебель на 25 посадочных мест, рабочее место преподавателя, стол - 1 шт., доска маленькая меловая - 1 шт., доска маркерная на колесиках - 1 шт., кафедра - 1 шт., вешалка стойка - 3 шт., жалюзи - 2 шт., Компьютер I5-7400/8Gb/1Tb/DELL S2218H - 24 шт., Интерактивная доска ScreenMedia OP78 с мобильной стойкой и крепеж для проектора - 1 шт. Наборы демонстрационного оборудования и учебно-наглядных пособий: мультимедийные приложения к лекционным курсам и практическим занятиям, интерактивные учебно-наглядные пособия. | 191023, г. Санкт-Петербург, ул. Канал Грибоедова, 30/32, литер «А», «Б», «Р» |
| Ауд. 0007 Компьютерный класс (для проведения практических занятий, курсового проектирования (выполнения курсовых работ) с применением вычислительной техники). Оборудован мультимедийным комплексом. Специализированная мебель и оборудование: Учебная мебель на 33 посадочных места, рабочее место преподавателя, доска меловая - 1 шт., доска маркерная на колесиках - 1 шт., вешалка стойка - 3 шт., жалюзи - 3 шт., Компьютер Intel Core i3 6100/ MSI H110M PRO-D/ ОЗУ DDR4 8GB 2400MHz/SSD SATA III 240Gb/Aerocool Qs-180 400W/Клавиатура + мышь Microsoft400 for Business/монитор Asus VS228DE - 24 шт., Мультимедийный проектор Тип 1 Optoma x 400 - 1 шт., Ноутбук HP 250 G6 1WY58EA - 2 шт. Наборы демонстрационного оборудования и учебно-наглядных пособий: мультимедийные приложения к лекционным курсам и практическим занятиям, интерактивные учебно-наглядные пособия. | 191023, г. Санкт-Петербург, ул. Канал Грибоедова, 30/32, литер «А», «Б», «Р» |
| Ауд. 2068 Учебная аудитория (для проведения занятий лекционного типа и занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации), оборудована мультимедийным комплексом.Специализированная мебель и оборудование: Учебная мебель на 94 посадочных места, рабочее место преподавателя, доска маркерная - 1 шт., кафедра - 1 шт., стол - 1 шт., стул - 2 шт., Компьютер Intel i3-2100 2.4 Ghz/500/4/Acer V193 19" - 1 шт., Акустическая система JBL CONTROL 25 WH - 2 шт., Экран с электропривод. DRAPER 96 - 1 шт., Акустическая система - 1 шт., Доска магнитно-маркерная 100\*200 - 1 шт., Мультимедийный проектор Panasonic PT-VX610E - 1 шт. Наборы демонстрационного оборудования и учебно-наглядных пособий: мультимедийные приложения к лекционным курсам и практическим занятиям, интерактивные учебно-наглядные пособия. | 191023, г. Санкт-Петербург, ул. Канал Грибоедова, 30/32, литер «А», «Б», «Р» |

# **7. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩЕГОСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ**

Приступая к изучению дисциплины, обучающемуся необходимо ознакомиться со следующими документами:

* учебно-методической документацией;
* локальными нормативными актами, регламентирующими основные вопросы организации и осуществления образовательной деятельности, в том числе регламентирующие порядок проведения текущего контроля успеваемости и промежуточной аттестации обучающихся;
* графиком консультаций сотрудников профессорско-преподавательского состава.

Уровень и глубина освоения дисциплины определяются активной и систематической работой обучающихся на лекционных занятиях, занятиях семинарского типа, выполнением самостоятельной работы, в том числе в части выделения наиболее значимых и актуальных проблем для дальнейшего изучения. Особым условием качественного освоения дисциплины является эффективная организация труда, позволяющая распределить учебную нагрузку равномерно в соответствии с графиком учебного процесса.

При подготовке к учебным занятиям обучающимся предоставляется возможность посещения консультаций сотрудников профессорско-преподавательского состава СПбГЭУ согласно расписанию, установленному в графике консультаций.

Аудиторная и внеаудиторная работа обучающихся должна быть направлена на формирование:

* фундаментальных основ мировоззрения обучающихся и естественнонаучного познания;
* базисных знаний, соответствующих направлению подготовки и заявленной профессиональной области, формирующих целевую и профессиональную основу для подготовки кадров;
* профессиональных компетенций ориентированных на удовлетворение потребностей рынка труда;
* индивидуальной траектории посредством освоения уникального набора профессиональных компетенций дополняющих компетентностную модель обучающегося, за счет ориентации на конкретные профессиональные специализированные области знаний, определяемые представителями рынка труда;
* метанавыков обучающихся, таких как: командная работа и лидерство, анализ данных, цифровые навыки, разработка и реализация проектов, межкультурное взаимодействие.

# **8. 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.

Обучающиеся из числа инвалидов и лиц с ОВЗ обеспечены печатными и (или) электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья. Образование обучающихся с ограниченными возможностями здоровья может быть организовано как совместно с другими обучающимися, так и в отдельных группах или в отдельных организациях.

# **ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**

## **1.1 Контрольные вопросы и задания к промежуточной аттестации**

**Вопросы:**

1. Стандартные типы данных в Python, проверка типа данных, явное и неявное преобразование типов.
2. Числовые типы данных в Python, операции над числами.
3. Вещественные числа в Python, форматирование чисел с использованием % и f-строки, функция round().
4. Модуль math Python: округление с помощью ceil() и floor(), математические функции.
5. Булев тип данных в Python, преобразование к типу bool.
6. Операции сравнения в Python, сравнение объектов различных типов.
7. Операции над булевыми переменными в Python.
8. Логические операции в Python, сложные логические выражения, приоритет операторов.
9. Условный оператор if в Python, вложенные условные операторы.
10. Функции в Python: объявление, аргументы, результат и выход из функции.
11. Области видимости переменных в функциях Python.
12. Анонимные функции (лямбда-выражения) в Python.
13. Строки в Python: создание, индексы, срезы, служебные символы и экранирование символов, операции над строками.
14. Методы строки в Python.
15. Форматирование строк в Python: метод .format и f-строка.
16. Функция print в Python.
17. Циклы for и while в Python.
18. Списки в Python, создание и изменение, индексы и срезы, операции над списками.
19. Методы списков в Python.
20. Генераторы списков и функция map() в Python.
21. Кортежи в Python: создание, основные операции над кортежами и методы кортежей.
22. Модули в Python: классификация, создание собственного модуля, подключение модуля, инcтрукции import и from, использование псевдонимов.
23. Оценка алгоритмов по времени и памяти, О-нотация. Оценка производительности кода с использованием модулей time, timeit.
24. Генерация псевдослучайных чисел в Python. Модуль random. Методы модуля random для целых чисел, вещественных чисел и списков.
25. Словари в Python: способы задания, операции над словарями.
26. Методы словаря в Python.
27. Множества в Python: способы задания, основные операции с множествами.
28. Методы множеств в Python.
29. Исключения и их обработка в Python, оператор try.
30. Классы в Python, создание класса и экземпляра класса, атрибуты и методы класса и экземпляра класса.

**Задания:**

|  |  |
| --- | --- |
| № | Текст задачи |
| 1 | Создайте класс Cat, у которого есть:   * метод класса **init**, принимающий аргументы: имя (name), цвет (color) и записывающий их в соответствующие защищенные атрибуты; * метод voice, который выводит на экран сообщение "Meeeoooww!"   Объявите экземпляр класса Cat и сохраните его в переменную cat.  Затем создайте класс Dog, у которого есть:   * инициализатор класса **init**, принимающий аргументы: имя (name), цвет (color) и записывающий их в соответствующие защищенные атрибуты; * метод voice, который выводит на экран сообщение "Bark! Woof!"   Объявите экземпляр класса Dog и сохраните его в переменную dog.  Вызовите метод voice сначала у cat, а затем у dog. |
| 2 | Создайте:Родительский класс Pet с инициализатором класса init, который принимает аргумент имя (name).  * Дочерний класс Cat наследуется от Pet и имеет метод print\_info(), который печатает информацию об экземпляре класса в следующей форме: Имя: {name}, Вид: Кошка * Дочерний класс Dog наследуется от Pet и имеет метод print\_info(), который печатает информацию об экземпляре класса в следующей форме: Имя: {name, Вид: Собака * Дочерний класс Hamster наследуется от Pet и имеет метод print\_info(), который печатает информацию об экземпляре класса в следующей форме: Имя: {name}, Вид: Хомяк   Напишите реализацию классов. |
| 3 | Реализуйте функцию print\_products(). Функция должна распечатать список продуктов. На вход функции поступает произвольное количество значений. Продуктом считается любая непустая строка. Любые другие типы данных (списки, числа и так далее) нужно проигнорировать. Напечатать список продуктов необходимо в следующем виде:  Пример. Код:  print\_products('печенье', 'хлеб', 'молоко')  Вывод:   1. печенье 2. хлеб 3. молоко   Если в переданных значениях не встретится ни одного продукта, то распечатайте: "Нет продуктов". |
| 4 | Напишите функцию create\_character(), которая принимает произвольное количество именованных аргументов и возвращает словарь с характеристиками персонажа. Если функции create\_character() не передавать никаких аргументов, то она должна возвращать базовый словарь с ключами name, surname, age. Вот так он выглядит.  Пример. Если передавать именованные параметры, которые отсутствуют в базовом словаре, они дополняются к этому словарю.  Код:  create\_character(height=180, author='Артур Конан Дойл')  Вывод:  {'name': 'Шерлок', 'surname': 'Холмс', 'age': 42, 'height': 180, 'author': 'Артур Конан Дойл'} |
| 5 | Обработайте исключения, которые могут возникнуть при выполнении приведенного кода. Если возникло исключение — выведите фразу 'Что-то пошло не так, проверьте введенные данные'. Код:  a = float(input())  b = float(input())  c = a / b  print(c) |
| 6 | Программа принимает на вход строковую переменную — имя файла и пробует его открыть. Выясните, какой тип исключения вызывается в программе. Обработайте данное исключение и выведите сообщение 'Нам не прочесть этот файл'. Код:  file\_name = input()  file = open(file\_name, 'r')  print(file.read()) |
| 7 | На вход подается строка из 10 случайных неуникальных чисел через пробел.  Для каждого числа выведите в отдельной строке слово "YES", если это число ранее встречалось в последовательности, или "NO", если не встречалось. |
| 8 | Напишите программу, которая удаляет из строки повторяющиеся символы. Необходимо учитывать регистр букв.  Входные данные:  На вход подается строка, состоящая из букв, цифр, символов и знаков препинания.  Выходные данные:  Программа должна вывести строку, из которой удалены все дублирующие символы. |
| 9 | Программа должна генерировать два списка случайной длины в диапазоне [10, 20]. Каждый список должен состоять из случайных неуникальных чисел в диапазоне [0, 100[. Найдите разность между количеством уникальных элементов (то есть элементов, которые встречаются лишь в одном из списков) в этих двух списках и выведите ее на экран. Разность возьмите по модулю, воспользовавшись функцией abs(). |
| 10 | В программе задан массив, состоящий из некоторого количества целочисленных элементов, отсортированных по возрастанию. На вход программы передается целое число x. Напишите реализацию алгоритма правого бинарного поиска, выдающую индекс последнего вхождения элемента x в заданный массив или -1, если элемент отсутствует в списке. |
| 11 | На вход подается массив, состоящий из целочисленных элементов, упорядоченных по возрастанию. Также подается целое число х, известно, что в массиве нет элементов, равных по значению х. С помощью бинарного поиска найдите позицию (индекс), на которую нужно вставить элемент х (подвинув элементы, стоящие справа), чтобы сохранить упорядоченность массива. |
| 12 | Дан упорядоченный целочисленный массив длины N, содержащий уникальные значения в диапазоне от 0 до N включительно. С помощью бинарного поиска найдите единственное пропущенное число из диапазона 0-N. |
| 13 | Напишите функцию генерации списка уникальных целых случайных чисел.Аргументы функции: размер списка, нижняя и верхняя границы диапазона чисел включительно, могут быть отрицательными. Функция должна возвращать список заданной длины, состоящий из случайных чисел, лежащих в заданном диапазоне. Числа не должны повторяться. Продемонстрируйте работу функции на тестовых примерах. |
| 14 | Напишите функцию генерации списка случайных символов русского или английского алфавита (по выбору) в нижнем регистре. Аргументы функции: размер списка, язык (eng - английский по умолчанию, ru - русский). Функция должна возвращать список заданной длины, состоящий из случайных символов. Символы могут повторяться. Продемонстрируйте работу функции на тестовых примерах. |
| 15 | Напишите функцию генерации списка уникальных случайных символов русского и английского алфавита (по выбору) в нижнем регистре. Размер списка должен быть ограничен алфавитом языка. Символы не должны повторяться. Аргументы функции: размер списка, язык символов (eng - английский по умолчанию, ru - русский). Функция должна возвращать список заданной длины, состоящий из случайных символов. Продемонстрируйте работу функции на тестовых примерах. |
| 16 | Напишите функцию генерации списка строк (слов), состоящих из случайно выбранных букв английского алфавита. Аргументы функции: размер списка, длина строки (от 0 до 9, по умолчанию 3). Слова могут повторяться. |
| 17 | Напишите функцию генерации списка уникальных строк (слов), состоящих из случайно выбранных букв английского алфавита. Аргументы функции: размер списка, длина строки (от 0 до 9, по умолчанию 3). Слова не должны повторяться.Продемонстрируйте работу функции на тестовых примерах. |
| 18 | Напишите программу, моделирующую бросание 2-х игральных кубиков. Определите эмпирические вероятности каждой комбинации при числе бросков равном 10, 100, 1000, 10000. |
| 19 | В программе задана числовая матрица произвольного размера. Передайте на вход программы две целочисленные переменные n и m, а затем замените на m те элементы матрицы, которые больше n. Выведите обновленную матрицу на экран. |
| 20 | Вам дается целое число n. Выведите матрицу размера nxn, в которой все элементы 0, кроме главной диагонали, на которой стоят 1. (Главная диагональ квадратной матрицы состоит из элементов с индексами i и j, в которых i=j. То есть это элементы с индексами 00, 11, 22 и т.д.) |
| 21 | Необходимо раскрасить шахматную доску размера nхn в черный и белый цвета. Элемент с индексом [0][0] раскрасьте в белый цвет, а далее в шахматном порядке. Для обозначения черного цвета используйте символ "b", для обозначения белого цвета — символ "w". |
| 22 | Напишите программу, выводящую на экран пирамиду заданной высоты, состоящую из символов \*. Значение высоты вводится пользователем. Важно соблюдать увеличение количества звезд в зависимости от высоты, а также правильно центрировать звездочки. Требуется именно центрированная пирамида-елочка в привычном понимании. |
| 23 | Напишите программу, которая принимает на вход строку и определяет, является ли она палиндромом (читается одинаково слева направо и справа налево). Программа должна выводить на экран True, если строка - палиндром. В противном случае программа должна выводить False. Учтите, что строка может содержать пробелы и буквы в разном регистре. |
| 24 | Напишите программу, содержащую функции проверки паспорта. Номера паспортов в России состоят из 6 цифр. Допустим, что паспорта нового образца начинаются с номеров 75 и 76. Функция должна принимать на вход номер паспорта и возвращать "new", если передан номер паспорта нового образца, и "old", если передан номер паспорта старого образца. Если передана строка, не соответствующая правилам формирования номера паспорта функция должна вернуть "wrong number".Продемонстрируйте работу функции на тестовых примерах. |
| 25 | Программа принимает на вход целочисленную переменную n (n>1) и формирует последовательность целых чисел от 0 до n включительно, в которой первые 3 элемента равны 1, 1 и 2 соответственно, а каждый следующий равен сумме трех предыдущих. Программа должна выводить на экран значение n-ого элемента последовательности. |
| 26 | Числа Фибоначчи — известная числовая последовательность. Она задается следующим образом: первые два элемента: 0 и 1, а далее каждый элемент является суммой двух предыдущих. То есть последовательность Фибоначчи имеет вид: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, … Напишите программу, которая по введенному n (2 < n < 30) выводит n-е число Фибоначчи. |
| 27 | Напишите программу, которая генерирует массив из 30 случайных целых чисел в диапазоне от 0 до 100, и содержит функцию сортировки выбором по возрастанию, которая принимает в качестве аргумента сгенерированный массив, а затем возвращает отсортированный массив. В качестве проверки программа сверяет результаты сортировки Вашей функцией и стандартной функцией сортировки. |
| 28 | Анаграмма строки S1 — это новая строка S2, которая получена из исходной строки S1 путем перестановки ее символов. Напишите программу, которая на вход получает две строки S1 и S2 и проверяет является ли строка S2 анаграммой строки S1. Выведите "YES", если S2 — анаграмма S1, и "NO" в противном случае. |
| 29 | Напишите программу, которая принимает на вход 3 строки произвольной длины. Каждая из строк состоит из слов, разделенных пробелами. 1-е слово строки является названием страны, последующие слова – названиями городов этой страны. Из введенных строк программа должна формировать словарь, в котором ключ – название страны, а значение – список городов этой страны. Если во введенной строке уже существующая в словаре страна, программа должна проверить список городов этой страны и дополнить его, если в строке есть новые города. |
| 30 | Напишите программу, которая находит количество уникальных слов во введенном предложении на английском языке, длина которых больше 7. Регистр букв учитывать не нужно. |

## **1.2 Темы письменных работ**

|  |  |
| --- | --- |
|  | Рабочей программой дисциплины не предусмотрено. |

## **1.3 Контрольные точки**

|  |  |  |  |
| --- | --- | --- | --- |
| **Номер контрольной точки** | **Тип контрольной точки** | **Способ проведения** | **Номера тем** |
| 1 | Решение задач | с помощью технических средств и информационных систем | 1-8 |
| 2 | Решение задач | с помощью технических средств и информационных систем | 9-15 |
| 3 | Текущий контроль | с помощью технических средств и информационных систем | 16-19 |

## **1.4 Другие объекты оценивания**

|  |  |
| --- | --- |
|  | Рабочей программой дисциплины не предусмотрено. |

## **1.5 Самостоятельная работа обучающегося**

|  |  |
| --- | --- |
| **Наименования самостоятельной работы** | **Номера тем** |
| Подготовка к лекционным и практическим занятиям | 1-21 |
| Подготовка сообщений, докладов | 9,12,19,20,21 |
| Подготовка к экзамену | 1-21 |

## **1.6** **Шкала оценивания результата**

Шкалы оценивания и процедуры оценивания результатов обучения **по дисциплине** регламентируются Положением о текущем контроле успеваемости и промежуточной аттестации обучающихся по программам высшего образования и Положением о балльно-рейтинговой системе.

Для оценки сформированности результатов обучения по дисциплине используется **балльно-рейтинговая система успеваемости обучающихся**:

Формой итогового контроля по дисциплине является экзамен (или дифференцированный зачет), итоговая оценка формируется в соответствии со шкалой, приведенной ниже в таблице:

|  |  |
| --- | --- |
| Баллы | Оценка |
| <=54 | неудовлетворительно |
| 55-69 | удовлетворительно |
| 70-84 | хорошо |
| >=85 | отлично |

**Шкала оценивания результата**

|  |  |
| --- | --- |
| 2 (балл до 54) | Демонстрирует непонимание проблемы. Многие требования, предъявляемые к заданию не выполнены.  Демонстрируется первичное восприятие материала. Работа незакончена и /или это плагиат. |
| 3 (балл 55-69) | Демонстрирует частичное понимание проблемы. Большинство требований, предъявляемых, к заданию выполнены.  Владение элементами заданного материала. В основном выполненный материал понятен и носит целостный характер. |
| 4 (балл 70-84) | Демонстрирует значительное понимание проблемы обозначенной дисциплиной. Все требования, предъявляемые к заданию выполнены.  Содержание выполненных заданий раскрыто и рассмотрено с разных точек зрения. |
| 5 (балл 85-100) | Демонстрирует полное понимание проблемы. Все требования, предъявляемые к заданию выполнены.  Продемонстрировано уверенное владение материалом дисциплины. Выполненные задания носят целостных характер, выполнены в полном объеме, структурированы, представлены различные точки зрения, продемонстрирован творческий подход. |