**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ РЕСПУБЛИКИ**

**ОУ “КЫРГЫЗСКО-КОРЕЙСКИЙ КОЛЛЕДЖ”**

«Утверждаю»

Директор ККК

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Бейшеналиева Г.Б.

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2024г.

**УЧЕБНО – МЕТОДИЧЕСКИЙ КОМПЛЕКС**

По дисциплине: «Python programming»

Для студентов направления: Computer Science

дневной формы обучения

Учебно-методический комплекс разработан преподавателем \_\_\_\_\_\_Нармухамедовым Р.Т.

**Бишкек 2024г.**

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ РЕСПУБЛИКИ**

**ОУ “КЫРГЫЗСКО-КОРЕЙСКИЙ КОЛЛЕДЖ”**

«Утверждаю»

Директор ККК

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Бейшеналиева Г.Б

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2024г.

**РАБОЧАЯ ПРОГРАММА**

По дисциплине: «Python Programming»

Для студентов направления: Computer Science

Дневной формы обучения

Всего часов по учебному плану 90

из них:

Лекций………………...............................................16

Практических занятий..............................................32

СРС………………….................................................42

Итоговый контроль – экзамен

Рабочая программа разработана преподавателем \_\_\_\_\_\_\_\_\_\_\_ Нармухамедовым Р.Т.

**Бишкек 2024г.**

**1.Введение (пояснительная записка)**

Основу содержания дисциплины "Python Programming" составляют основные принципы построения алгоритмов и их реализации. Студенты изучают различные структуры данных и методы их обработки, что позволяет эффективно решать задачи. Важное внимание уделяется написанием кода, которые помогают визуализировать алгоритмы. Также рассматриваются основные операторы и конструкции программирования, что способствует формированию развития логики python в примерах.

Основными критериями отбора тем и методов для изучения являются:

 Актуальность: Темы должны соответствовать современным требованиям и трендам в области программирования.

 Практическая значимость: Изучаемые методы должны быть применимы в реальных проектах и задачах, что способствует развитию практических навыков.

 Постепенность сложности: Темы должны быть организованы так, чтобы последовательно увеличивать уровень сложности и углублять знания студентов.

 Междисциплинарность: Темы должны интегрироваться с другими дисциплинами, что позволяет студентам видеть связь между различными областями знаний.

**2. Цели освоения дисциплины**

Рабочая программа учебной дисциплины "Операционные системы" ориентирована на достижение следующих целей:

1. **Формирование теоретических знаний** о принципах работы логики программирования, их архитектуре и основных функциях, что позволит студентам глубже понять, как функционируют современные компьютерные системы.
2. **Развитие практических навыков** в установке, настройке и администрировании различных логических систем, что обеспечит студентов необходимыми компетенциями для решения реальных задач в сфере информационных технологий.
3. **Изучение методов управления ресурсами** python, включает процессы, логики написания кода, что поможет студентам эффективно использовать ресурсы библиотек и прочих модулей.
4. **Подготовка к исследовательской деятельности** в области разработки и улучшения логики что способствует развитию аналитического мышления и способности к решению сложных задач в области python программирования.

Применение знаний об операционных системах для хранения и обработки данных включает манипуляции с файловыми системами, управление доступом к данным и оптимизацию дискового пространства. Это позволяет эффективно организовывать данные, обеспечивать целостность информации и разрабатывать приложения для быстрого доступа к данным.

В результате освоения дисциплины студент должен знать:

 Овладеть методологией анализа и проектирования классических типов алгоритмов (линейных, условных, циклических, рекурсивных и эвристических).

 Знать теоретические и практические аспекты перехода от содержательной постановки задачи к разработке алгоритма её решения.

 Алгоритмы планирования процессов и механизмы многозадачности.

 Уметь применять полученные знания к решению вопросов составления и документирования алгоритмов, разработки программ и их сопровождения, использования современных инструментальных средств и технологий.

В результате освоения дисциплины студент должен уметь:

 Устанавливать и настраивать различные операционные системы.

 Управлять процессами и памятью в операционной системе.

 Осуществлять манипуляции с файловыми системами и организовывать данные.

 Решать практические задачи, связанные с безопасностью и защитой информации в ОС.

**3. Методы изучения дисциплины.**

Изучение данной дисциплины предполагает проведение:

Лекций – 16 часов,

Практических занятий - 32 часа,

СРС – 42 часа.

Промежуточная аттестация производится согласно учебному плану, рабочему учебному плану и графику учебного процесса.

Проведение итогового контроля по завершении курса-экзамен.

**Содержание дисциплины *Fundamentals of Algorithms***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Наименование темы** | **Лекционные занятия** | **Практические занятия** | **СРС** |
| **1 модуль** | | | | |
| **1.** | Введение в программирование на языке Python. | 2 | 2 | 4 |
| **2.** | Переменные и типы данных в Python. | 1 | 2 | 2 |
| **3.** | Операторы Python. Арифметический, логический, реляционный, битовый, оператор сдвига. | 1 | 2 | 4 |
| **4.** | Основные стандартные модули Python. Библиотеки. | 1 | 2 | 2 |
| **5.** | Структура ветвления, поток управления if/elif/else. | 1 | 2 | 4 |
| **6.** | Работа с циклами. Операторы for, while, break, continue. Функции отладки, Функция обработки исключений. | 1 | 2 | 2 |
| **7.** | Цикличная структура. Теория и практика цикличной структуры. | 1 | 2 | 4 |
| **8.** | Рубежный контроль: Тест | 2 | | |
| **2 модуль** | | | | |
| **9.** | Функции и методы работы со строками в Python. Одномерные массивы в Python. | 2 | 2 | 2 |
| **10.** | Структура данных. Список, методы списков, кортеж, словарь, набор. | 1 | 2 | 4 |
| **11.** | Работа с двумерными массивами. Создание списка. | 1 | 2 | 2 |
| **12.** | Структура программы в Python. Ввод и вывод данных. | 1 | 2 | 4 |
| **13.** | . Файловый ввод/вывод. Пользовательские подпрограммы и моделирование. Модуль Tkinter. | 1 | 2 | 2 |
| **14.** | Объектно-ориентированное программирование (ООП). Класс. Инкапсуляция. | 1 | 2 | 4 |
| **15.** | Полиморфизм и наследования | 1 | 2 | 2 |
| **16.** | Рубежный контроль: Тест | 2 | | |

**График проведения рейтингового контроля**

|  |  |
| --- | --- |
| **Наименование тем** | **Сроки сдачи** |
| Практическая работа №1  Практическая работа №2  Практическая работа №3  Практическая работа №4  Практическая работа №5  Практическая работа №6 | До 8 недели |
| **Модуль 1** | 8-я неделя |
| Практическая работа №7  Практическая работа №8  Практическая работа №9  Практическая работа №10  Практическая работа №11  Практическая работа №12 | До 16 недели |
| **Модуль 2** | 16-я неделя |
| **Экзамен** | 17-я неделя |

**Оценка знаний, умений и навыков**

В рабочем учебном плане предусмотрено проведение СРС, организационного контроля, рубежного контроля и итогового экзамена. Организационный контроль – оценка посещаемости студентов за период учебного года. Рассматривается по окончании 1 и 2 модулей за семестр, одновременно для всех студентов.

Рубежный контроль – оценка знаний по освоению пройденного материала. Проводится по окончании 1 и 2 модулей, одновременно для всех студентов.

Итоговый экзамен проводится по окончанию семестра в период экзаменационной недели.

Сумма общего балла – СРС, организационного (ОК), промежуточного (РК) и итогового контроля (ИК) исчисляется следующим путем:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 модуль | | | 2 модуль | | | Текущий контроль | ИК | Общий балл |
| ОК 1 | СРС 1 | РК 1 | ОК 2 | СРС 2 | РК 2 |
| 10 | 10 | 10 | 10 | 10 | 10 | 60 | 40 | 100 |

Баллы распределяются следующим образом:

**«Отлично»**

- А+ (95-100)

- А (90-94)

- А- (85-89);

**«Хорошо»**

- В+(80-84)

- В (75-79)

- В (70-74);

**«Удовлетворительно»**

- С+ (67-69)

- С (64-66)

- С- (60-63);

**«Неудовлетворительно»:**

- D (40-59)

- F (<40)

Сумма баллов всех дисциплинарных модулей и итогового экзамена должна составлять минимальных - 60, а максимальных – 100.

**Учебно-методические материалы**

 **"Изучаем Python"** — Марк Лутц  
Полное руководство по языку Python, подходящее как для новичков, так и для более опытных программистов.

 **"Python для анализа данных"** — Уэс МакКинни  
Книга, посвященная использованию Python для анализа и обработки данных с использованием библиотек, таких как Pandas и NumPy.

 **"Python. Книга для профессионалов"** — Дэвид Бимонт  
Подробный справочник по языку Python с практическими примерами и углубленным изучением.

 **"Изучаем Python 3"** — Эрик Мэтиз  
Учебник для начинающих, который охватывает все основные аспекты Python.

 **"Алгоритмы на Python"** — Магомед Касимов  
Книга, посвященная алгоритмам и структурам данных с использованием Python.

 **"Fluent Python"** — Лучано Рамальо  
Изучение Python на более глубоком уровне с акцентом на его возможности и идиомы.

 **"Python Crash Course"** — Эрик Мэтиз  
Быстрый и практичный вводный курс по Python, который охватывает основы языка.

 **"Automate the Boring Stuff with Python"** — Аль Свейгарт  
Книга, ориентированная на автоматизацию рутинных задач с помощью Python.

 **"Learning Python"** — Марк Лутц  
Обширное руководство, охватывающее основные концепции Python с примерами и упражнениями.

 **"Effective Python"** — Бретт Слотт  
Книга с 59 способами улучшить свои навыки программирования на Python и писать более эффективный код.