

## АННОТАЦИЯ К РАБОЧЕЙ ПРОГРАММЕ ДИСЦИПЛИНЫ

09.03.01 Информатика и вычислительная техника

Профиль: Вычислительные машины, комплексы, системы и сети

### Б1.О.08 «Программирование»

Индекс	Наименование	Семестр 1										Семестр 2										Итого за курс									
		Контроль	Академических часов									з.е.	Контроль	Академических часов									з.е.	Контроль	Академических часов						з.е.
			Всего	Контакт.	Лек	Лаб	Пр	КРП	СР	Контроль	Всего			Контакт.	Лек	Лаб	Пр	КРП	СР	Контроль	Всего	Контакт.			Лек	Лаб	Пр	КРП	СР	Контроль	
Б1.О.08	Программирование	Экз	252	68	34	34			148	36	7	Экз	180	50	34	16			94	36	5	Экз(2)	432	118	68	50			242	72	12

Формируемые компетенции: ОПК-7, ОПК-8, ОПК-9.

### Содержание дисциплины

Лекционные занятия 17 шт. по 2 часа:

- 1.1. Введение в язык. Основные понятия и история языка. Структура программы. Объявления (фундаментальные типы, идентификаторы, литералы, неявные конверсии типов, разница между объявлением и определением, области видимости). Выражения (базовые операторы, операторы присваивания, операторы сравнения, тернарный оператор, инкремент и декремент, оператор sizeof, оператор запятая).
- 1.2. Управляющие конструкции. Виды ошибок и UB (compile-time error (ошибки компиляции), runtime error (ошибки выполнения), undefined behaviour (UB)). Модификаторы типов и связанные с ними вопросы. Указатели. Функции (указатели на функции, функции с переменным числом аргументов, аргументы по умолчанию, перегрузка функций).
- 1.3. Операторы new и delete. Ссылки. Константы. Операторы явного преобразования типов (cast). Операторы «.», «->», this. Оператор присваивания, правило трех. Списки инициализации в конструкторах. Ключевое слово explicit. Константные методы.
- 1.4. Ключевое слово mutable. Ключевое слово friend. Делегирующие конструкторы. Ключевое слово static. Указатели на члены.
- 1.5. Перегрузка операторов. Арифметические операторы. Операторы сравнения. Инкремент и декремент. Квадратные скобки. Круглые скобки. Операторы «&&», «||» и «,». Операторы «\*», «&» и «- > ». Перегрузка C-style cast.
- 1.6. Введение в ООП. Классы и структуры. Конструктор, деструктор и оператор присваивания (конструктор, деструктор, конструктор копирования).
- 1.7. Наследование. Объявление. Соккрытие имен при наследовании. Поиск имен при наследовании (соккрытие имен наследником, явный вы-

зов методов предка, почему компилятор может не увидеть публичный метод или атрибут класса, различие классов и структур при наследовании, тонкости запретов при public-наследовании от класса с private-наследованием, доступность друзьям protected методов/атрибутов предка).

1.8. Порядок вызова конструкторов и деструкторов при наследовании. Множественное наследование (проблема ромбовидного наследования, проблема «сын - прямой наследник бабушки»). Виртуальное наследование.

1.9. Приведение типов между наследниками (dynamic\_cast). Виртуальные функции. Виртуальный деструктор. Абстрактные классы. Override и final. Оператор typeid.

1.10. Шаблоны. Объявление шаблонов. Специализации шаблонов. Использование typedef, typename, remove\_const, remove\_reference, etc. Правила вывода шаблонных типов. non-type template parameters.

1.11. Variadic templates (шаблоны с переменным количеством шаблонных аргументов). Использование функтора. CRTP (Curious Recurring Template Pattern).

1.12. Исключения (exceptions). Особенности исключений. Ключевые слова throw, try, catch. Отличие между исключениями и ошибками. Последовательность и правила обработки исключений.

1.13. Копирование при исключениях. Проблемы исключений (спецификации исключений «старая версия» и «новая версия»). Проблема исключений в конструкторах. Проблема исключений в деструкторах. Гарантия безопасности при исключениях.

1.14. Аллокаторы. Оператор placement-new. Перегрузка operator new и operator delete. nothrow new, new\_handler, allocator, allocator\_traits.

1.15. Контейнеры. Последовательные контейнеры (vector, deque, array, list, forward\_list) Методы insert, splice, unique, sort, erase, size.

1.16. Ассоциативные контейнеры (set, multiset, map, multimap). Методы find, count, lower\_bound, upper\_bound, equal\_range, unordered\_map, unordered\_set, unordered\_multimap. Адаптеры (stack, queue, priority\_queue).

1.17. Шаблонные параметры. hash. Специальные функции (для хеш-таблиц). Сравнение времени работы операций для контейнеров.

Лабораторные работы 11 шт. по 2 часа:

2.1. Линейные алгоритмы.

2.2. Разветвляющиеся алгоритмы.

2.3. Циклические алгоритмы.

2.4. Циклические алгоритмы, практическое применение.

2.5. Циклические алгоритмы, обработка последовательностей.

2.6. Двухмерные массивы. Обработка части массива.

2.7. Функции.

2.8. Функции. Массивы, как параметры.

2.9. Рекурсия.

2.10. Файлы.

2.11. Классы.

Лабораторные работы 3 шт. по 4 часа:

2.12. Одномерные массивы.

- 2.13. Двухмерные массивы
- 2.14. Строки.

Лекционные занятия 17 шт. по 2 часа:

- 1.1. Система контроля версий git. Основные операции, ветки, коммиты. Язык программирования python: описание и философия. Нотификация типов.
- 1.2. Стил написания: yagni, kiss, dry, solid. Кодировка, физические и логические строки. Операторные скобки. Юникод и эмодзи.
- 1.3. Строки, строковые литералы, операции со строками. Операции форматирования. Идентификаторы, пространства имен, область видимости.
- 1.4. Встроенные типы в языке программирования Python: целочисленный, вещественный, комплексный, логический. Управляющие конструкции: операторы ветвлений и циклов.
- 1.5. Функции в языке Python, lambda функции. Функции первого рода, чистые функции. Виды параметров функции. Параметры по умолчанию: \*args, \*\*kwargs.
- 1.6. Функции преобразования типов. Функции ввода-вывода.
- 1.7. Последовательности. Кортежи. (,) и tuple(). Последовательности. Списки. Срезы. [] и list(). Последовательности. Словари. {} и dict().
- 1.8. Множества и операции над ними. set(). Обработка исключений.
- 1.9. Декораторы. Генераторы. Генераторные выражения.
- 1.10. Итераторы. Функции: iter, enumerate, sorted. Модуль itertools. Функции itertools.chain, itertools.repeat, itertools.count. Функции для обработки последовательностей: range, map, filter, sum, reduce, zip.
- 1.11. Объектно-ориентированное программирование (ООП) на языке программирования Python. Описание классов. ООП в python: атрибуты, свойства, сокрытие данных.
- 1.12. Типизация и полиморфизм. Имитация типов. Перегрузка операторов. Магические методы. ООП в Python. Особенности наследования. Статические методы и методы класса.
- 1.13. Утиная типизация в Python. Менеджер контекста. Работа с файлами в Python. Чтение бесконечных файлов.
- 1.14. Асинхронное программирование. Мотивация, проблемы и решения. Asyncio.async/await. Корутины. GIL и многопоточные программы. Мьютексы и семафоры. Класс Thread. Поток, процесс.
- 1.15. Модули и пакеты. GUI библиотеки. Tkinter, PyQt, Библиотеки numpy, matplotlib, pandas, sympy.
- 1.16. Фреймворки Django, aiohttp. Особенности применения. Тестирование. Юнит и интеграционное тестирование. Форматы CSV, json, xml. Чтение и запись средствами Python.
- 1.17. Регулярные выражения. Определение, шаблоны, основные методы. Ускорение Python. Numba, Cython, интеграция с C/C++.

Лабораторные работы 8 шт. по 2 часа:

- 2.1. Система контроля версий git.
- 2.2. Структура проекта. Обсуждение технического задания.
- 2.3. Разработка библиотеки.

- 2.4. Выгрузка библиотек в открытый доступ.
- 2.5. Тестирование.
- 2.6. Код ревью.
- 2.7. Оформление пояснительной записки.
- 2.8. Представление проекта.

*Год начала подготовки (по учебному плану)*  
*Образовательный стандарт (ФГОС)*

2022  
№ 929 от 19.09.2017