

## Программная инженерия

### Лабораторная работа №5

Срок сдачи: 23.12.23

Кроме методов, перечисленных в заданиях, для всех классов необходимо реализовать метод-инициализатор `__init__` и методы для получения строкового представления `__str__` и `__repr__`. Также при инициализации объекта необходимо проверять допустимость входных аргументов.

1. Создать класс для описания даты. Объект должен содержать атрибуты: день, месяц, год (три целых числа). В классе необходимо определить следующие методы:

- 1) метод сравнения `__eq__(self, other)` (т.е. оператор `==`);
- 2) метод, который проверяет, является ли год високосным;
- 3) метод, который вычисляет разницу между двумя датами в днях.

2. Написать класс для описания многочлена. Объект должен содержать в качестве атрибутов степень многочлена и его коэффициенты. В классе необходимо определить следующие методы:

- 1) метод сравнения `__eq__(self, other)` (т.е. оператор `==`);
- 2) методы для сложения, вычитания и умножения: `__add__(self, other)`, `__sub__(self, other)`, `__mul__(self, other)`;
- 3) метод `__call__(self, x)`, который вычисляет значение многочлена в точке  $x$ .

3. Написать класс для описания многочлена. Объект должен содержать в качестве атрибутов степень многочлена и его коэффициенты. В классе необходимо определить следующие методы:

- 1) метод, который возвращает многочлен, являющийся производной данного многочлена;
- 2) метод, который возвращает многочлен, являющийся первообразной данного многочлена и принимающий заданное значение в точке  $x = 0$ .

4. Написать класс для описания несократимой дроби. Объект должен содержать в качестве атрибутов числитель и знаменатель (два целых числа). В классе необходимо определить следующие методы:

- 1) методы сравнения `__eq__(self, other)` и `__lt__(self, other)` (т.е. операторы `==` и `<`);
- 2) методы для сложения, вычитания и умножения: `__add__(self, other)`, `__sub__(self, other)`, `__mul__(self, other)`;
- 3) метод, который проверяет является ли дробь правильной.

**Указание.** Для нахождения наибольшего общего делителя двух целых чисел можно использовать функцию `math.gcd`.

5. Написать класс для описания работника предприятия. Объект должен содержать в качестве атрибутов фамилию, имя, отчество, оклад и дату поступления на работу. В классе необходимо определить следующие методы:

- 1) метод, который вычисляет стаж работника на данном предприятии;
- 2) метод, который изменяет оклад на заданное число процентов;
- 3) метод сравнения `__eq__(self, other)` (т.е. оператор `==`).

**Указание.** Текущую дату можно получить при помощи метода `date.today()` класса `date`, который определен в модуле `datetime`.

6. Написать класс для описания комплексного числа. Объект должен содержать в качестве атрибутов вещественную и мнимую части. В классе необходимо определить следующие методы:

- 1) метод сравнения `__eq__(self, other)` (т.е. оператор `==`);
- 2) методы для сложения, вычитания, умножения и деления: `__add__(self, other)`, `__sub__(self, other)`, `__mul__(self, other)`, `__truediv__(self, other)`;
- 3) метод, который вычисляет модуль комплексного числа;

7. Написать класс для описания трехмерного вектора в прямоугольной декартовой системе координат. Объект должен содержать в качестве атрибутов координаты вектора (три числа). В классе необходимо определить следующие методы:

- 1) метод сравнения `__eq__(self, other)` (т.е. оператор `==`);
- 2) методы для сложения векторов, вычитания векторов и умножения на число: `__add__(self, other)`, `__sub__(self, other)`, `__rmul__(self, other)`;
- 3) метод, который вычисляет модуль вектора.

8. Написать класс для описания трехмерного вектора в прямоугольной декартовой системе координат. Объект должен содержать в качестве атрибутов координаты вектора (три числа). В классе необходимо определить следующие методы:

- 1) метод, вычисляющий скалярное произведение данного вектора с другим;
- 2) метод, вычисляющий векторное произведение данного вектора с другим;
- 3) метод, который возвращает единичный вектор, имеющий то же направление, что и данный.
- 4) метод, который вычисляет смешанное произведение данного вектора с двумя другими.

9. Написать класс, который поддерживает интерфейс итератора (методы `__iter__` и `__next__`) и удовлетворяет условиям:

- 1) метод `__init__` принимает целое число  $N \geq 2$ ;
- 2) последовательные вызовы функции `next` для экземпляра возвращают все простые числа, не превосходящие  $N$ , начиная с числа 2.

10. Написать класс для описания прямой проходящей через две точки на плоскости. Метод `__init__` должен принимать два кортежа, каждый из

которых содержит два числа – координаты двух точек. В классе необходимо определить следующие методы:

- 1) метод, который вычисляет единичный направляющий вектор прямой – кортеж из двух чисел;
- 2) метод, который вычисляет единичный нормальный вектор прямой – кортеж из двух чисел;
- 3) метод сравнения `__eq__(self, other)` (т.е. оператор `==`);
- 4) метод, который проверяет параллельна ли данная прямая другой прямой.

11. Написать класс, реализующий изменяемую строку. Метод `__init__` должен принимать строку, а символы хранить в виде списка односимвольных строк. В классе реализовать следующие методы:

- 1) методы `__getitem__(self, idx)`, `__setitem__(self, idx)`, `__delitem__(self, idx)` для получения, изменения и удаления символа по его позиции;
- 2) метод `__len__(self)` для получения длины строки;
- 3) метод `__int__(self)` для перевода строки в целое число, если строка состоит только из цифр.

### Дополнительное задание.

Написать декоратор, который создает на основе заданной функции функцию с проверкой типов аргументов при помощи аннотаций. Например, если декорировать функцию с заголовком `f(x: int)`, то вызов `f(x=1.0)` должен выбрасывать исключение **`TypeError`**.

Считать, что декоратор применим для функций, которые удовлетворяют следующим условиям:

- 1) функция не принимает универсальных аргументов `*args` и `**kwargs`;
- 2) аннотациями являются только обычные классы (`int`, `str`, `list` и т.д.), но не псевдонимы классов или специальные классы для аннотаций (`list[int]`, `TypeVar`, `Final`, `Literal` и т.д.);
- 3) не обязательно все аргументы функции снабжены аннотациями.

### Список вариантов

2, 9	Садыков Тимур Наилевич
3, 7	Исмагилов Матвей Георгиевич
4, 8	Капланов Заур Джамалович
6, 8	Иванчук Александра Дмитриевна
1, 5	Дедяева Дарья Александровна
7, 9	Степанов Алексей Игоревич
5, 10	Янченко Ксения Сергеевна

4, 5	Кузьмин Илья Евгеньевич
1, 3	Усманов Ильнур Ленарович
5, 10	Островский Артём Витальевич
6, 11	Руденко Кирилл Вячеславович
1, 9	Сулейманова Саида Сулеймановна
7, 11	Денисова Наталия Николаевна
2, 6	Пачковский Никита Владимирович