

Домашнее задание по теме «Понятия ООП: наследование, инкапсуляция, полиморфизм»

Формулировка задания

Решить несколько задач на python с применением принципов ООП. Каждая задача представляет собой написание функции или класса. Результирующий код должен быть читаемым, с единой системой отступов и адекватными названиями переменных.

Описание плана работы

Решить несколько задач на python с применением принципов ООП. Желательна реализация в файлах `.py`. Сохранить задачи (сделать коммиты для каждой) в локальном `git` и опубликовать в удаленном репозитории.

Для отчета по работе выполнить задание в файле `.py` или `.ipynb`. Сделать снимок корректного выполнения программы в IDE.

Задача 1

Опишите класс комплексных чисел. У пользователя должна быть возможность создать его объект на основе числа и в алгебраической форме, и в полярной. Класс должен поддерживать основные математические операции (+, -, \cdot , /) за счет перегрузки соответствующих магических методов. Также он должен поддерживать возможность получить число в алгебраической и полярной форме. Допускается использование модуля `math`.

Задача 2

Опишите класс для векторов в N-мерном пространстве. В качестве основы используйте список значений координат вектора, задаваемый `list`. Обеспечьте поддержку следующих операций: сложение, вычитание (с созданием нового вектора-результата), скалярное произведение, косинус угла, евклидова норма. Все операции, которые можно перегрузить с помощью магических методов, должны быть реализованы именно через них. Класс должен производить проверку консистентности аргументов для каждой операции и в случаях ошибок выбрасывать исключение `ValueError` с исчерпывающим объяснением ошибки.

Перечень необходимых инструментов

- Python
- math
- venv
- Jupiter Notebook
- IDE VS Code
- GigaIDE

Форма предоставления результата

1. В поле ссылки загрузить ссылку на удаленный репозиторий с доступом для наставника.
2. В поле файла загрузить архив с папкой, в которой разместить отчет со скриншотами по заданиям и решение задач части 1 и 2. Решения должны быть представлены в формате .ipynb или .py.

Шкала оценивания

- 1.0 – отлично
- 0.7–0.9 – хорошо
- 0.5–0.6 – удовлетворительно
- Менее 0.5 – задание не выполнено