

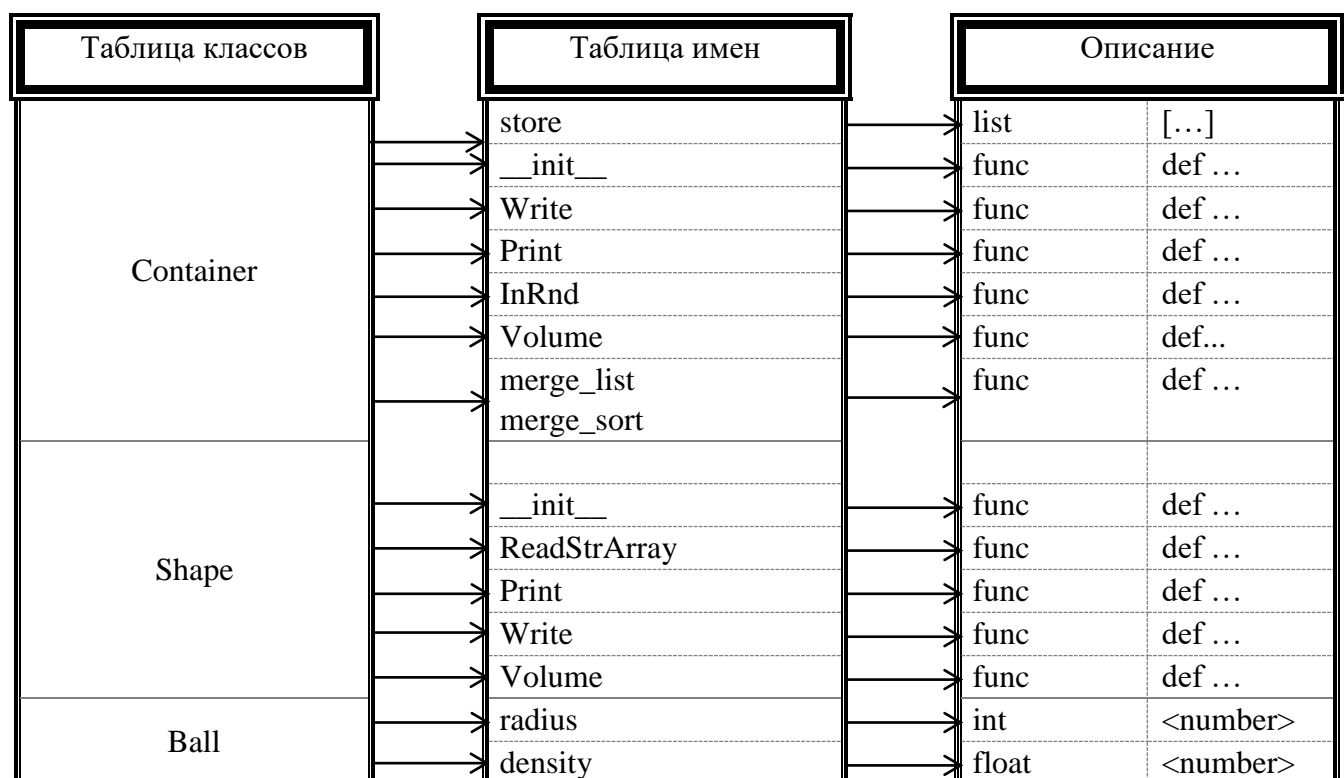
ОПИСАНИЕ ЗАДАНИЯ

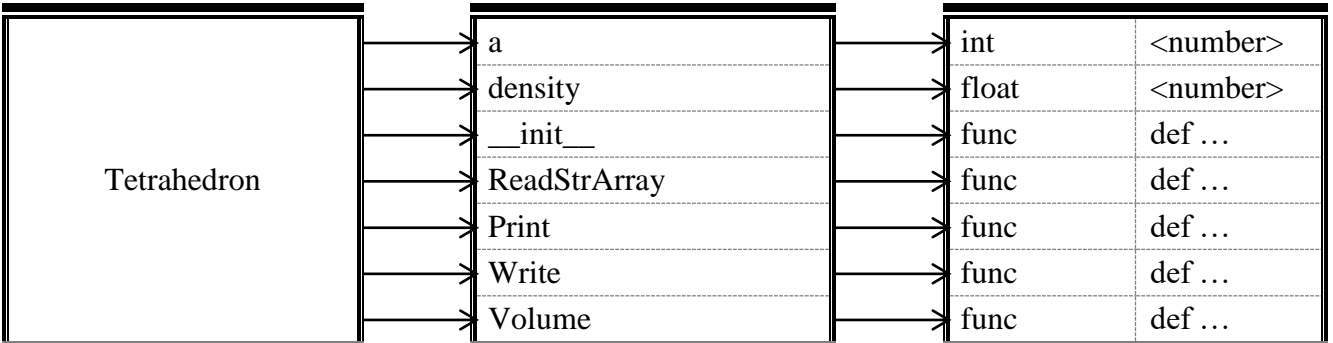
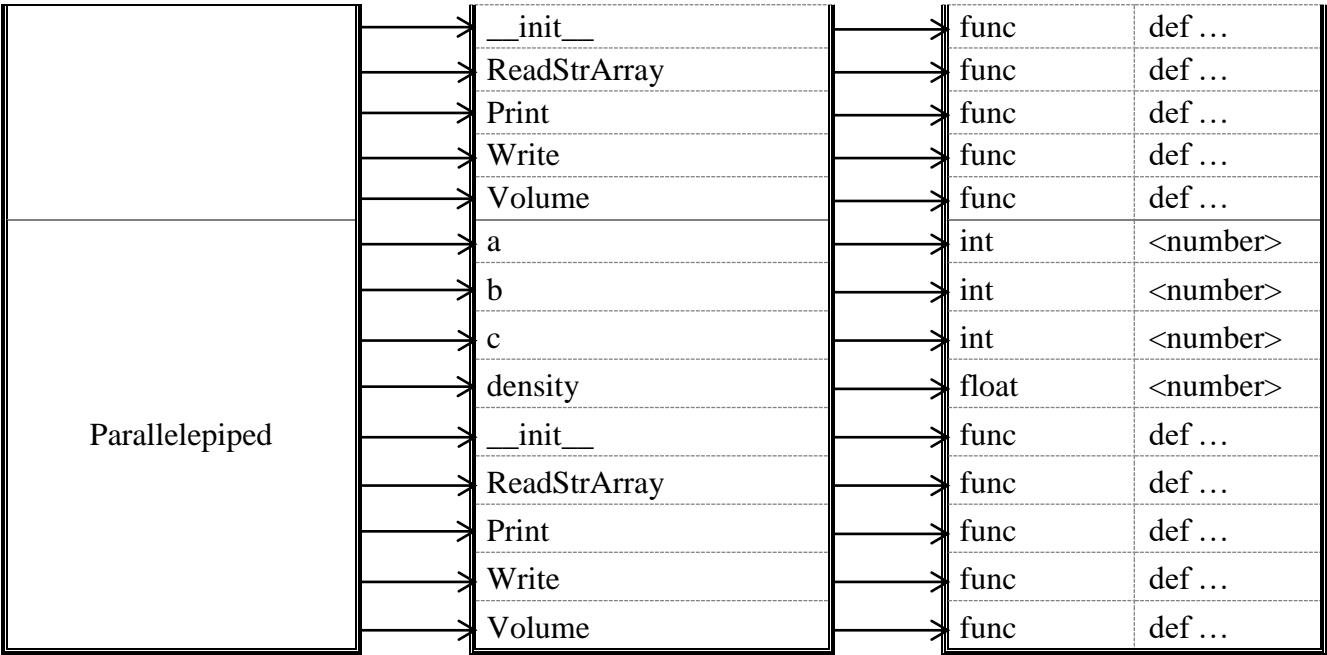
Разработать программный продукт с использованием динамической проверки типов во время выполнения программы. Программа должна содержать следующие структуры и функции:

Обобщенный артефакт, используемый в задании	Базовые альтернативы (уникальные параметры, задающие отличительные признаки альтернатив)	Общие для всех альтернатив переменные	Общие для всех альтернатив функции
Объемная (трехмерная) геометрическая фигура.	1. Шар (целочисленный радиус) 2. Параллелепипед (три целочисленных ребра) 3. Правильный тетраэдр (длина ребра – целое)	Плотность материала фигуры (действительное число)	Вычисление объема (действительное число)

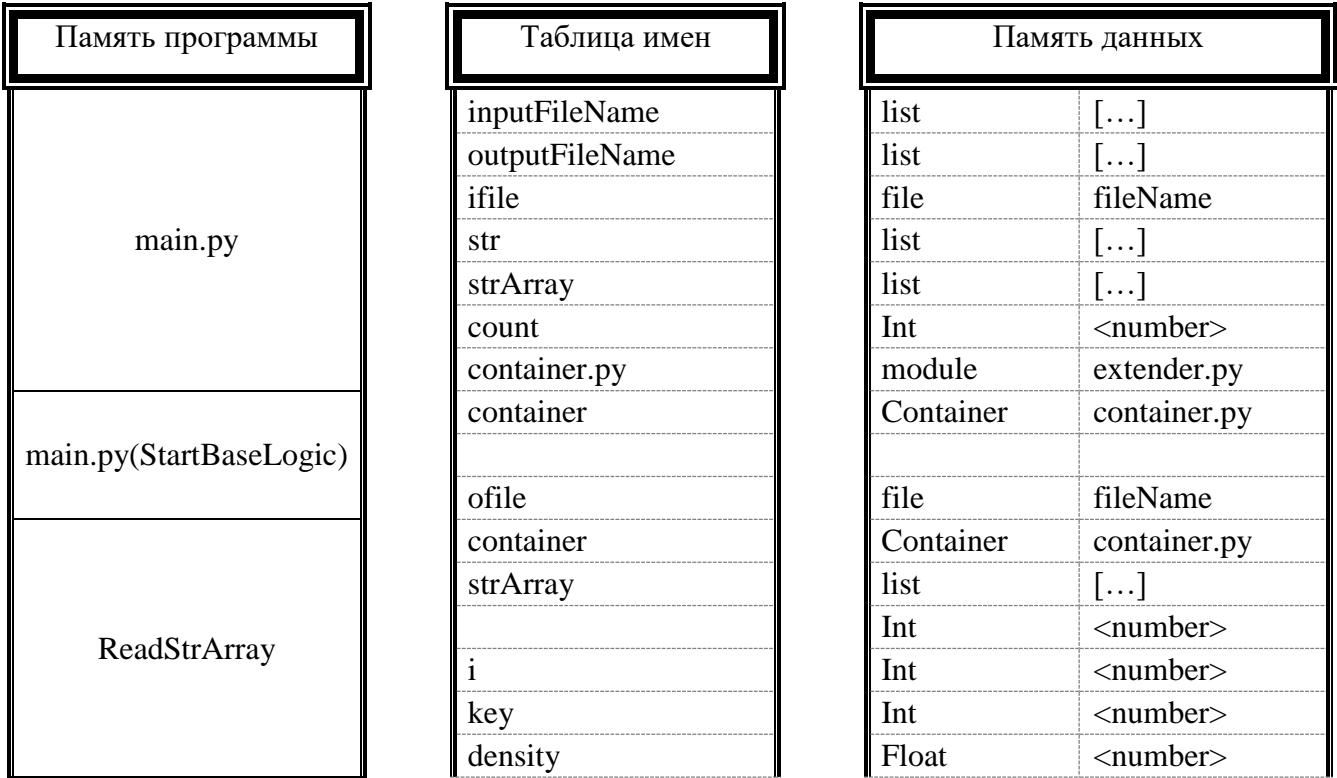
Дополнительная функция:

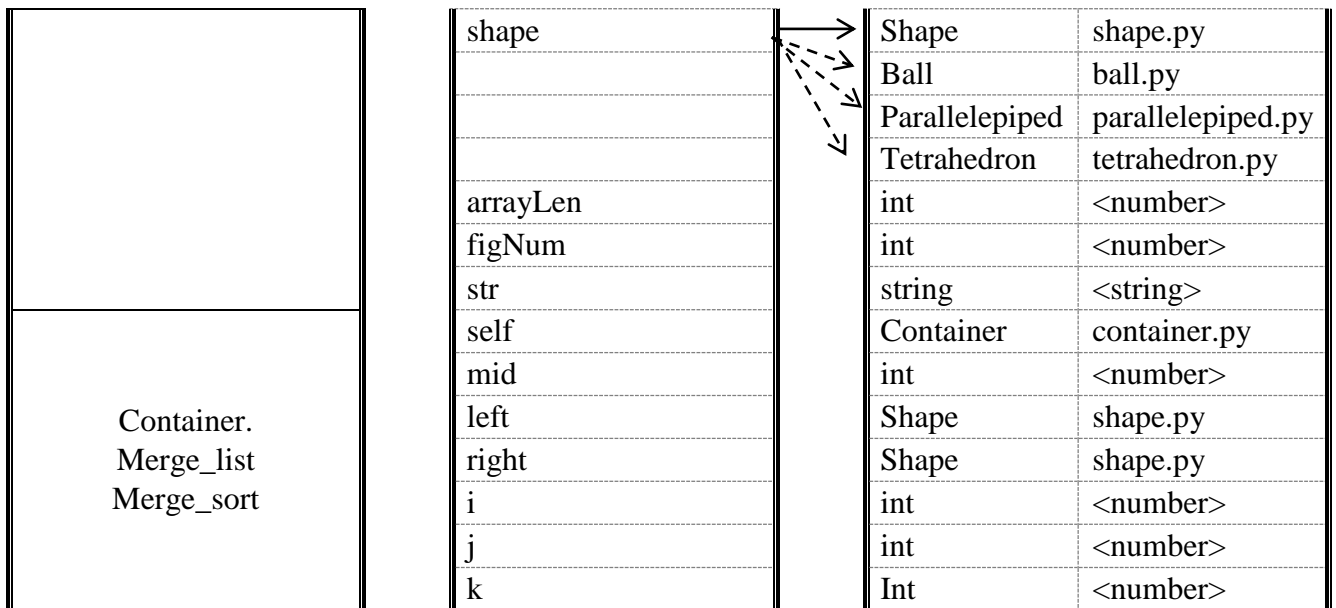
Упорядочить элементы контейнера по убыванию используя сортировку с помощью прямого слияния (Straight Merge). В качестве ключей для сортировки и других действий используются результаты функции, общей для всех альтернатив (объем).

СТРУКТУРНАЯ СХЕМА АРХИТЕКТУРЫ ВС С ПРОГРАММОЙ:



ОТРАЖЕНИЕ НА ПАМЯТЬ МЕТОДОВ КЛАССОВ:





ОСНОВНЫЕ ХАРАКТЕРИСТИКИ ПРОГРАММЫ:

- Число заголовочных файлов – 1 (extender.py)
- Число модулей реализации – 7
- Общий размер исходных текстов – 326 строк кода вместе с комментариями
- Размер исполняемого файла – 0 (файл не генерируется, так как питон – скриптовый язык).
- Время выполнения программы для различных тестовых прогонов:

Номер теста	Время выполнения в секундах
Тест1	0.000998
Тест2	0.0079464
Тест3	0.012875
Тест4	0.008526
Тест5	0.020942

СРАВНЕНИЕ С ПРЕДЫДУЩИМИ ВЕРСИЯМИ ПРОГРАММЫ:

Программа работает быстрее, чем версия, написанная на С и примерно с той же скоростью, что и программа, написанная на С++:

Таблица тестов программы на С:

Номер теста	Время выполнения, сек
1	0.011
2	0.016
3	0.045
4	0.047
5	0.1

Таблица тех же тестов программы на С++:

Номер теста	Время выполнения, сек
1	0.006
2	0.008
3	0.013
4	0.017
5	0.029

- В программе появились новые опции – отпала необходимость явно задавать выходной файл (если выходной файл не задан, то программа сгенерирует его сама);
- Вывод в файл теперь осуществляется в один файл, а не в 2, как это было раньше. Это помогает немного экономить время на очистки потока и запросе файла у системы.
- Пропала возможность измерять размер типов. В Python нет возможности получить размер типа (только объекта какого-то типа), поэтому провести разумный анализ затрачиваемой памяти не представляется возможным.
- Количество строк кода уменьшилось в 2 раза, что связано с облегченным синтаксисом питона.