



Хранение объектов в памяти

Таблица виртуальных методов

Интерпретатор языка Mython

- ✓ **Материал для самостоятельного изучения:** Интерпретатор языка Mython
25 min
- ✓ **Видео:** Введение в разработку трансляторов
4 min
- ✓ **Задание по программированию:** Лексический анализатор языка Mython
4h
- ✓ **Материал для самостоятельного изучения:** Решение задачи «Лексический анализатор языка Mython»
10 min
- ✓ **Задание по программированию:** Интерпретатор языка Mython
12h
- ✓ **Материал для самостоятельного изучения:** Решение задачи «Интерпретатор языка Mython»
1h

Финальный проект, части J–M

- ✓ **Видео:** Транспортный справочник, части J–M
1 min
- ✓ **Видео:** Транспортный справочник, часть I



Задание по программированию: Интерпретатор языка Mython

✓ Зачет · 1/1 баллов

Срок сдачи Задание необходимо было сдать 29 июня: 13:59 +07
Вы все еще можете выполнить задание до завершения курса.

Инструкции

Моя работа

Обсуждения

Итак, в предыдущей задаче мы создали лексический анализатор для языка mython. Теперь с его помощью создать синтаксический анализатор, а затем собственный интерпретатор. Для создания синтаксического анализатора требуется определённая теоретическая подготовка, которая выходит за рамки «Чёрного пояса по C++», так что в этой задаче мы выдадим вам готовую реализацию нисходящего LL(1)-парсера языка Mython. (Если вы не поняли после нескольких слов, это нестрашно — вы сможете решить задачу без их понимания. Однако это говорит о том, что нет тех самых теоретических знаний, о которых говорилось выше. Лучший источник, в котором их можно получить, это Книга Дракона, канонический учебник по созданию компиляторов и трансляторов).

Этот парсер формирует абстрактное синтаксическое дерево, узлами которого являются объекты классов, унаследованные от абстрактного класса Statement:

```
1 struct Statement {
2     virtual ~Statement() = default;
3     virtual ObjectHolder Execute(Runtime::Closure& c) = 0;
4 };
```

Единственный метод интерфейса выполняет разбор программы на языке mython.