

**Вопросы к экзамену по курсу  
«Компиляторные Технологии»  
(ВШЭ, весенний семестр 2021/22 учебного года)**

1. Структура оптимизирующего компилятора. Промежуточное представление среднего уровня (*MIR*). Перевод программы в *MIR*.
2. Базовые блоки и граф потока управления (ГПУ) программы. Алгоритм выделения базовых блоков и построения ГПУ.
3. Локальная оптимизация процедуры методом нумерации значений. Ориентированный ациклический граф и его построение для базового блока. Связь по данным между базовыми блоками процедуры.
4. Локальная оптимизация процедуры методом нумерации значений. Необходимость построения множеств *Input* и *Output*.
5. Определение и использование переменной. Алгоритм вычисления достигающих определений для базовых блоков процедуры.
6. Понятие живой (активной) переменной. Алгоритм вычисления множеств живых переменных для базовых блоков процедуры.
7. Доступные выражения (определение). Алгоритм вычисления множеств доступных выражений для базовых блоков процедуры.
8. Определение полурешетки с операцией «сбор» ( $\wedge$ ). Полурешеточное отношение частичного порядка ( $\leq$ ). Наибольшая нижняя граница ( $\inf(x, y)$ ) элементов  $x$  и  $y \in L$  и ее вычисление. Диаграмма полурешетки.
9. Структура потока данных и общий итерационный алгоритм анализа потока данных. Монотонные и дистрибутивные структуры потока данных. Примеры.
10. Сходимость итерационного алгоритма анализа потока данных к максимальной неподвижной точке (*MFP*-решение).
11. Идеальное решение анализа потока данных (*Ideal*), решение по всем путям (*MOP*-решение). Соотношение между идеальным решением, *MOP*-решением и *MFP*-решением. Консервативность анализа.
12. Доминаторы, их свойства, алгоритм вычисления доминаторов. Алгоритм построения дерева доминаторов.
13. Постдоминаторы, их свойства, алгоритм вычисления постдоминаторов. Алгоритм построения дерева постдоминаторов.
14. Граница доминирования и алгоритм ее построения. Обратная граница доминирования.

15. Простые машинно-независимые оптимизирующие преобразования: алгебраические упрощения, сворачивание констант, распространение копий.
16. Алгоритм исключения бесполезного кода.
17. Алгоритм исключения недостижимого кода.
18. Оптимизация потока управления.
19. Выделение естественных циклов и гнезд циклов.
20. Вынесение кода, инвариантного относительно цикла, за границы цикла.
21. *SSA*-форма. Алгоритм построения максимальной *SSA*-формы.
22. *SSA*-форма. Алгоритм построения частично-усеченной *SSA*-формы.
23. Алгоритм переименования переменных при построении *SSA*-формы.
24. Восстановление кода из *SSA*-формы. Замена  $\phi$ -функций группами инструкций копирования.
25. Глобальное распространение констант. Недистрибутивность передаточной функции распространения констант.
26. Глобальное распространение констант в *SSA*-форме. Двухфазный алгоритм.
27. Глобальная нумерация значений. Связь с *SSA*-формой.
28. Анализ потока данных на основе областей. Двухфазный алгоритм.