

**Вопросы на экзамен
по дисциплине «Теория языков программирования»**

1. Алфавит. Цепочки символов в алфавите. Операции над цепочками символов. Формальное определение грамматики и языка.
2. Способы задания синтаксиса формальных языков.
3. Классификация языков и грамматик по Хомскому.
4. Разбор цепочек КС-грамматик. Дерево разбора.
5. Однозначные и неоднозначные грамматики. Эквивалентность грамматик.
6. Общая схема распознавателя. Классификация распознавателей.
7. Регулярные выражения как способ задания регулярных языков.
8. Конечные автоматы как способ задания регулярных языков.
9. Построение конечного автомата по регулярной грамматике.
10. Преобразование конечного автомата к детерминированному виду.
11. Минимизация конечных автоматов.
12. Проверка существования языка и устранение бесполезных символов КС-грамматики.
13. Устранение цепных и ϵ -правил в КС-грамматиках.
14. Устранение левой факторизации и прямой левой рекурсии правил в КС-грамматиках.
15. Определение и функционирование МП-автомата.
16. Построение МП-автомата по КС-грамматике.
17. Построение расширенного МП-автомата по КС-грамматике.
18. Нисходящие распознаватели языков. Метод рекурсивного спуска.
19. Построение множеств $FIRST(1, A)$ для $LL(1)$ – грамматики.
20. Построение множеств $FOLLOW(1, A)$ для $LL(1)$ – грамматики.
21. Функционирование распознавателя цепочек для $LL(1)$ -грамматик.
22. Определение грамматики простого предшествования. Поиск основы предложения.
23. Построение множеств $L(A)$ и $R(A)$ для грамматик простого предшествования.
24. Функционирование распознавателя цепочек грамматик простого предшествования.
25. Определение грамматики операторного предшествования. Поиск основы предложения.
26. Построение множеств $L^t(A)$ и $R^t(A)$ для грамматик операторного предшествования.
27. Функционирование распознавателя цепочек грамматик операторного предшествования.
28. Восходящие распознаватели. Определение $LR(k)$ – грамматики.
29. Пример распознавателя для $LR(0)$ -грамматики.
30. Пример распознавателя для $LR(1)$ -грамматики.
31. Соотношение классов КС-языков.
32. Лексика, синтаксис и семантика языка. Определение транслятора, интерпретатора, компилятора и ассемблера.
33. Общая схема работы компилятора.
34. Лексический анализ программы: задачи, диаграмма и сканер.

35. Семантический анализ: проверка правильности описания переменных, выражений и операторов.
36. Формы внутреннего представления программы: триады и тетрады.
37. Формы внутреннего представления программы: синтаксические деревья и ассемблерные команды.
38. ПОЛИЗ как форма внутреннего представления программы.
39. Генерация кода объектной программы.
40. Принципы оптимизации кода. Оптимизация линейных участков программ.
41. Свертка кода объектной программы.
42. Исключение лишних операций в линейных участках программ.
43. Оптимизация циклов и логических выражений в программах.
44. Интерпретация программы.
45. Схемы компиляции.
46. СУ-схема. СУ-перевод.
47. МП-преобразователи.
48. СУ-перевод. Преобразование дерева разбора исходной программы.
49. СУ-компиляция. Транслирующие грамматики.
50. Атрибутные транслирующие грамматики.