

Программа зачета по курсу «Системы программирования»

II курс (I и II потоки), весна 2003 г.
(лекторы Волкова И.А., Руденко Т.В.)

Теоретические вопросы

1. Этапы жизненного цикла программного продукта.
2. Состав и схема функционирования классической системы программирования.
3. Типы трансляторов, особенности интерпретаторов и компиляторов.
4. Общая схема работы компилятора.
5. Определение порождающей грамматики.
6. Определение языка, порождаемого грамматикой.
7. Определение эквивалентности грамматик.
8. Определение типов грамматики и языков по Хомскому.
9. Соотношения между типами грамматик.
10. Соотношения между типами языков.
11. Определение неоднозначности грамматики и языка.
12. Определение недостижимых и бесполезных символов, определение приведенной грамматики.
13. Определение детерминированного конечного автомата (КА).
14. Диаграмма состояний (ДС) для заданной левосторонней регулярной грамматики. Построение ДС.
15. Определение недетерминированного конечного автомата (НКА).
16. Алгоритм построения детерминированного конечного автомата по НКА.
17. Задачи лексического анализа.
18. Организация классов таблиц лексического анализатора.
19. Метод рекурсивного спуска: назначение, семантика процедур метода рекурсивного спуска.
20. Достаточные условия применимости метода рекурсивного спуска.
21. Расширение класса грамматик, к которым применим метод рекурсивного спуска (проблемы, преобразования, эквивалентность предлагаемых преобразований).
22. Задачи семантического анализа.
23. Свойства языка внутреннего представления программы, примеры таких языков.
24. Синтаксически управляемый перевод: идея, принципы организации, примеры.
25. ПОЛИЗ выражений.
26. ПОЛИЗ операторов языков программирования.
27. Интерпретация ПОЛИЗа.
28. Основные методы динамического распределения памяти (явного и неявного).
29. Машинно-независимая оптимизация. Основные оптимизирующие преобразования.
30. Машинно-зависимая оптимизация. Основные оптимизирующие преобразования.
31. Интегрированная среда разработки (ИСР).
32. Основные функции редактора текстов в рамках ИСР.
33. Основные функции отладчика в рамках ИСР.

34. Назначение и функционирование редактора связей. Загрузчик.
35. Библиотеки. Основные типы библиотек.
36. Критерии проектирования стандартных библиотек. Стандартная библиотека STL.
37. STL: контейнеры, итераторы, алгоритмы, аллокаторы. Шаблоны vector и list.

Практические навыки

1. Определить, к какому типу по Хомскому относится заданная грамматика.
2. Определить тип языка, порождаемого данной грамматикой.
3. Построить грамматику [определенного типа], порождающую заданный язык.
4. Определить, какой язык порождает заданная грамматика.
5. Построить дерево вывода (разбора) заданной цепочки по заданной КС-грамматике.
6. Получение приведенной грамматики.
7. Дана левостроенная регулярная грамматика. Построить по ней ДС, по ДС написать анализатор на C++.
8. Дана ДС конечного автомата. Восстановить левостроенную регулярную грамматику, порождающую язык, распознаваемый этим автоматом.
9. Дан текст анализатора на C++. Восстановить левостроенную регулярную грамматику по данному анализатору.
10. Дана левостроенная регулярная грамматика, которой соответствует НКА. Построить эквивалентный детерминированный КА (использовать алгоритм построения детерминированного КА по НКА).
11. Применим ли метод рекурсивного спуска к заданной КС-грамматике. Ответ обосновать.
12. Преобразовать заданную КС-грамматику так, чтобы к ней был применим метод рекурсивного спуска.
13. Для заданной КС-грамматики построить анализатор методом рекурсивного спуска.
14. Дан анализатор, написанный методом рекурсивного спуска. Восстановить КС-грамматику.
15. Дана грамматика с действиями. Какой язык она порождает?
16. Дана КС-грамматика. Дополнить ее действиями, позволяющими учесть дополнительные ограничения на цепочки определяемого языка.
17. Дана грамматика (или написать грамматику) языка L1. Вставить в грамматику действия по переводу цепочек языка L1 в цепочки языка L2.
18. Дана грамматика с действиями, осуществляющими перевод цепочек языка L1 в цепочки языка L2. Описать языки L1 и L2.
19. Записать в ПОЛИЗе заданный фрагмент программы на языке X.
20. Задан фрагмент программы в ПОЛИЗе. Восстановить текст на языке X.
21. Определить, является ли данная запись ПОЛИЗом заданной конструкции.
22. Разработать ПОЛИЗ заданной конструкции некоторого языка. Вставить в грамматику действия, порождающие ПОЛИЗ этой конструкции.
23. Решение простейших задач с использованием компонент библиотеки STL (vector и list).