

Консультации по вопросам к государственному экзамену 2015 года

5 курс ПМИ (дополнительная часть, III поток)

Вопрос 5. Транзакционное управление в СУБД. Методы сериализации транзакций.
http://citforum.ru/database/advanced_intro/39.shtml#13

Вопрос 9. Алгоритм Сети-Ульмана оптимального распределения регистров и его обоснование.

План ответа:

- Задача оптимального распределения регистров.
- Условия применения алгоритма.
- Обоснование алгоритма (минимальность числа используемых регистров при принятых ограничениях).
- Разметка дерева.
- Распределение регистров.
- Генерация кода.

Материал для подготовки в книге «Теория и реализация языков программирования». Авторы: Серебряков В. А., Галочкин М. П., Гончар Д. Р., Фуругян М. Г. <http://sp.cs.msu.ru/info/trlp.pdf>

Вопрос 11. Основные этапы компиляции (лексический анализ, синтаксический анализ, семантический анализ, генерация кода и т.д.).

При подготовке может быть полезна книга Серебряков В. А., Галочкин М. П., Гончар Д. Р., Фуругян М. Г. «Теория и реализация языков программирования». <http://sp.cs.msu.ru/info/trlp.pdf>

Вопрос 12. Построение детерминированного конечного автомата по регулярному выражению.

План ответа:

- Определение регулярного множества.
- Определение регулярного выражения (РВ).
- Определение недетерминированного конечного автомата (НКА).
- Определение детерминированного конечного автомата (ДКА).
- Алгоритм может быть представлен либо как последовательность двух алгоритмов: НКА → ДКА, либо непосредственно построение ДКА по дереву РВ.

Материал для подготовки в книге «Теория и реализация языков программирования». Авторы: Серебряков В. А., Галочкин М. П., Гончар Д. Р., Фуругян М. Г. <http://sp.cs.msu.ru/info/trlp.pdf>

Вопрос 13. Построение канонической системы множеств LR(1) ситуаций и таблиц действий и переходов для LR(1) грамматик.

План ответа:

- Правосторонний вывод.
- Восстановление правостороннего вывода.
- Схема работы магазинного анализатора, таблицы действий и переходов, конфигурации.
- Основа сентенциальной формы.
- LR(1) ситуация.
- Построение замыкания и переходы.
- Построение автомата (таблиц) анализатора по канонической системе множеств.
- Конфликты в таблицах.

Материал для подготовки в книге «Теория и реализация языков программирования». Авторы: Серебряков В. А., Галочкин М. П., Гончар Д. Р., Фуругян М. Г. <http://sp.cs.msu.ru/info/trlp.pdf>

Вопрос 15. Технологии параллельного программирования

Для подготовки к ответу используйте материалы раздела «Технологии» сервера НИВЦ МГУ (parallel.ru) и 5-й главы учебника Воеводина В. В. и Воеводина Вл. В. «Параллельные вычисления». Обсуждение задачи об «обедающих философах», к которой могут проявить интерес члены ГЭКа, можно найти в книге Эндрюса Г. Р. «Основы многопоточного, параллельного и распределённого программирования».

Вопрос 22. Унифицированный язык моделирования UML. Основные средства языка

План ответа:

- Назначение UML;
- Состав диаграмм UML и их назначение:
 - диаграммы вариантов использования (use case diagrams);
 - диаграммы классов (class diagrams);
 - диаграммы последовательности (sequence diagrams);
 - кооперативные диаграммы (collaboration diagrams);
 - диаграммы состояний (statechart diagrams);
 - диаграммы деятельности (activity diagrams);
 - диаграммы компонентов (component diagrams);
 - диаграммы размещения (deployment diagrams).

Материал для подготовке в книге Фаулер М. UML. Основы. 3-е издание.: Пер. с англ. – СПб: Символ-Плюс, 2004. - 192 с. Может быть полезен конспект 3-й лекции по курсу «Объектно-ориентированный анализ и проектирование» <http://sp.cs.msu.ru/ooap/conspect03.pdf>
Англоязычное пособие: <http://www.uml-diagrams.org>

Вопрос 23. Основы программной инженерии. Каскадная и итерационная модели жизненного цикла программного обеспечения.

План ответа:

- Понятие программного обеспечения и инженерии программного обеспечения.
- Понятие жизненного цикла программного обеспечения.
- Понятие процесса жизненного цикла. Процессы жизненного цикла по стандарту ISO 12207:1995 (основные, вспомогательные и организационные).
- Понятия: модель жизненного цикла, стадия, веха (или контрольная точка).
- Каскадная модель жизненного цикла.
- Итерационная модель жизненного цикла (на примере пошаговой или спиральной моделей).

Материал для подготовки в книге Иан Соммервил. Инженерия программного обеспечения. 6-е издание. М. – СПб. – Киев: 2002. – 623 с. Может быть полезен конспект 1-й лекции по курсу «Объектно-ориентированный анализ и проектирование» <http://sp.cs.msu.ru/ooap/conspect01.pdf>
Отвечая, следует иметь в виду, что наряду с каскадной и итерационной моделями жизненного цикла примеряется *эволюционная модель*. Она похожа на итерационную тем, что не является водопадной. В эволюционной модели итерации не планируются заранее. Вместо этого вводится рабочий цикл, в котором параллельно ведутся действия по проектированию, разработке и проверке качества. В рамках рабочего цикла создаются промежуточные версии, до тех пор, пока не будет реализована необходимая функциональность и достигнуто достаточное качество. Тогда рабочий цикл завершается, система поставляется заказчику, вводится в действие и т. п. Строго говоря эволюционная модель не относится к итерационным.

4 курс ПМИ (дополнительная часть, III поток)

Вопрос 5. Транзакционное управление в СУБД. Методы сериализации транзакций.

http://citforum.ru/database/advanced_intro/39.shtml#13

Вопрос 9. Алгоритм Сети-Ульмана оптимального распределения регистров и его обоснование.

План ответа:

- Задача оптимального распределения регистров.
- Условия применения алгоритма.
- Обоснование алгоритма (минимальность числа используемых регистров при принятых ограничениях).
- Разметка дерева.
- Распределение регистров.
- Генерация кода.

Материал для подготовки в книге «Теория и реализация языков программирования». Авторы: Серебряков В. А., Галочкин М. П., Гончар Д. Р., Фуругян М. Г. <http://sp.cs.msu.ru/info/trlp.pdf>

Вопрос 11. Основные этапы компиляции (лексический анализ, синтаксический анализ, семантический анализ, генерация кода и т.д.).

При подготовке может быть полезна книга Серебряков В. А., Галочкин М. П., Гончар Д. Р., Фуругян М. Г. «Теория и реализация языков программирования». <http://sp.cs.msu.ru/info/trlp.pdf>

Вопрос 12. Построение детерминированного конечного автомата по регулярному выражению.

План ответа:

- Определение регулярного множества.
- Определение регулярного выражения (РВ).
- Определение недетерминированного конечного автомата (НКА).
- Определение детерминированного конечного автомата (ДКА).
- Алгоритм может быть представлен либо как последовательность двух алгоритмов: НКА → ДКА, либо непосредственно построение ДКА по дереву РВ.

Материал для подготовки в книге «Теория и реализация языков программирования». Авторы: Серебряков В. А., Галочкин М. П., Гончар Д. Р., Фуругян М. Г. <http://sp.cs.msu.ru/info/trlp.pdf>

Вопрос 13. Построение канонической системы множеств LR(1) ситуаций и таблиц действий и переходов для LR(1) грамматик.

План ответа:

- Правосторонний вывод.
- Восстановление правостороннего вывода.
- Схема работы магазинного анализатора, таблицы действий и переходов, конфигурации.
- Основа сентенциальной формы.
- LR(1) ситуация.
- Построение замыкания и переходы.
- Построение автомата (таблиц) анализатора по канонической системе множеств.
- Конфликты в таблицах.

Материал для подготовки в книге «Теория и реализация языков программирования». Авторы: Серебряков В. А., Галочкин М. П., Гончар Д. Р., Фуругян М. Г. <http://sp.cs.msu.ru/info/trlp.pdf>

Вопрос 20. Качество программного обеспечения и методы его контроля. Тестирование и другие методы верификации.

Материалы для подготовки: лекция 5 по курсу «Основы программной инженерии»

<http://se-course.narod.ru/Lecture05.pdf> исключая описания стандартов ISO 9000, 9001, 9003, 9004, 90003 и раздел об ошибках. Обзор методов верификации

<http://www.ict.edu.ru/ft/005645/62322e1-st09.pdf>, разделы 1, начало раздела 3 (до 3.1), разделы 3.2, 3.3.5, 3.3.6, начало 3.4 (до 3.4.1) и 3.4.2-3.4.5.

4 курс ФИИТ (дополнительная часть)

Вопрос 15. Использование разных типов грамматик (по Хомскому) в реализации современных систем программирования.

При подготовке может быть полезна книга Серебряков В. А., Галочкин М. П., Гончар Д. Р., Фуругян М. Г. «Теория и реализация языков программирования». <http://sp.cs.msu.ru/info/trlp.pdf>

Вопрос 19. Жизненный цикл программного обеспечения и процессы его разработки.

Каскадная и итеративная модели жизненного цикла.

План ответа:

- Понятие программного обеспечения и инженерии программного обеспечения.
- Понятие жизненного цикла программного обеспечения.
- Понятие процесса жизненного цикла. Процессы жизненного цикла по стандарту ISO 12207:1995 (основные, вспомогательные и организационные).
- Понятия: модель жизненного цикла, стадия, веха (или контрольная точка).
- Каскадная модель жизненного цикла.
- Итерационная модель жизненного цикла (на примере пошаговой или спиральной моделей).

Материал для подготовки в книге Иан Соммервил. Инженерия программного обеспечения. 6-е издание. М. – СПб. – Киев: 2002. – 623 с. Может быть полезен конспект 2-й лекции по курсу «Основы программной инженерии» <http://se-course.narod.ru/Lecture02.pdf> или конспект 1-й лекции по курсу «Объектно-ориентированный анализ и проектирование» <http://sp.cs.msu.ru/ooap/conspect01.pdf>

Отвечая, следует иметь в виду, что наряду с каскадной и итерационной моделями жизненного цикла примеряется *эволюционная модель*. Она похожа на итерационную тем, что не является водопадной. В эволюционной модели итерации не планируются заранее. Вместо этого вводится рабочий цикл, в котором параллельно ведутся действия по проектированию, разработке и проверке качества. В рамках рабочего цикла создаются промежуточные версии, до тех пор, пока не будет реализована необходимая функциональность и достигнуто достаточное качество. Тогда рабочий цикл завершается, система поставляется заказчику, вводится в действие и т. п. Строго говоря эволюционная модель не относится к итерационным.

Вопрос 20. Архитектура программного обеспечения, методы ее описания (язык UML) и анализа. Архитектурные образцы.

При ответе следует дать определение понятию архитектура ПО, перечислить основные виды UML-диаграмм (д. вариантов использования, д. классов, д. объектов, д. компонентов, д. развертывания, д. последовательности, д. деятельности, д. состояний), кратко рассказать, зачем они нужны, какие аспекты архитектуры изображаются на диаграммах каждого вида. Следует привести пример UML-диаграммы какого-либо вида и дать пояснение по элементам и связям на ней. Следует перечислить четыре вида архитектурных образцов (конвейер, хранилище, вызов-возврат и интерактивная система), а также может рассказать про основные характеристики (структурную организацию и типичные сценарии взаимодействия) примеров

образцов каждого вида (например, "каналы и фильтры" из конвейерных, "репозитория" из основанных на хранилище, "многоуровневой системы" из вызов-возвратных и Model-View-Controller из интерактивных). Рассказать, чем они отличаются. Желательно уметь приводить примеры известных систем, построенных по различным образцам.

Материал для подготовки в конспектах 7-й и 8-й лекций по курсу «Основы программной инженерии» <http://se-course.narod.ru/Lecture07.pdf>, раздел про архитектурные стили и далее, <http://se-course.narod.ru/Lecture08.pdf>, раздел про MVC (там - "данные-представление-обработка"). Также полезна глава 2 книги F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal. Pattern-Oriented Software Architecture. A System of Patterns. Wiley, 2002 https://wiki.sch.bme.hu/images/9/98/Sznicak_jegyzet_Pattern-Oriented-SA_vol1.pdf и статья David Garlan, Mary Shaw. An Introduction to Software Architecture. CMU-CS-94-166. или в сборнике Advances in Software Engineering and Knowledge Engineering, Volume I, eds V. Ambriola and G. Tortora, 1993. http://www.cs.cmu.edu/afs/cs/project/vit/ftp/pdf/intro_softarch.pdf Справочником по UML является книга Фаулер М. UML. Основы. 3-е издание.: Пер. с англ. – СПб: Символ-Плюс, 2004. - 192 с. Может быть полезен конспект 3-й лекции по курсу «Объектно-ориентированный анализ и проектирование» <http://sp.cs.msu.ru/ooap/conspect03.pdf> Англоязычное пособие: <http://www.uml-diagrams.org>

Вопрос 21. Качество программного обеспечения и методы его контроля. Тестирование и другие методы верификации.

Материалы для подготовки: лекция 5 по курсу «Основы программной инженерии» <http://se-course.narod.ru/Lecture05.pdf> исключая описания стандартов ISO 9000, 9001, 9003, 9004, 90003 и раздел об ошибках. Обзор методов верификации <http://www.ict.edu.ru/ft/005645/62322e1-st09.pdf>, разделы 1, начало раздела 3 (до 3.1), разделы 3.2, 3.3.5, 3.3.6, начало 3.4 (до 3.4.1) и 3.4.2-3.4.5.