

ПРОГРАММА
вступительного испытания в магистратуру
по направлению **09.04.04 – Программная инженерия**

Дисциплина «Базы данных»

1. Основные понятия теории баз данных.
2. Развитие систем обработки данных. Современные тенденции в обработке данных.
3. Типовая организация современной СУБД.
4. Модели данных. Классификация моделей данных. Теоретико – множественные модели данных. Теоретико – графовые модели данных.
5. Сетевая модель. Сетевой граф базы данных. Достоинства и недостатки.
6. Иерархическая модель данных. Достоинства и недостатки.
7. Трехуровневая архитектура базы данных.
8. Свойства проектируемой СУБД.
9. Функции СУБД.
10. Жизненный цикл баз данных. Этап анализа и проектирования.
11. Логическое проектирование базы данных.
12. Жизненный цикл баз данных. Концептуальное проектирование базы данных.
13. Концептуальное проектирование базы данных. Объекты. Атрибуты. Конкретизация и обобщение.
14. Концептуальное проектирование базы данных. Связи между объектами. Мощность связи.
15. Концептуальное проектирование базы данных. Моделирование предметной области.
16. Концептуальное проектирование базы данных. Составные объекты.
17. Жизненный цикл баз данных. Этап реализации.
18. Физическое проектирование базы данных.
19. Разработка приложений. Тестирование.
20. Оценка работы и поддержка базы данных, эксплуатация и сопровождение.
21. Реляционная модель данных.
22. 12 правил реляционного подхода.
23. Функциональные зависимости и ключи.
24. Нормализация отношений и аномалии модификации.
25. Свойства отношений. Обновления отношений.
26. Проектирование нормализованной БД.
27. Первая нормальная форма.
28. Вторая нормальная форма.
29. Третья нормальная форма.
30. Нормализация на основе декомпозиции. Нормальная форма Бойса - Кодда.

31. Четвертая нормальная форма. Пятая нормальная форма.
32. Представление связей. Рекурсивная связь.
33. Преобразование концептуальной модели в реляционную модель.
34. Целостность данных.
35. Архитектура клиент – сервер.
36. Язык SQL. Функции языка. Достоинства языка.
37. Перечислите наиболее важные объекты Microsoft SQL Server.
38. Что понимается под объектом базы данных Microsoft SQL Server.
39. Перечислите системные базы данных, входящие в состав СУБД

после

40. Дайте определение журналу транзакций.
41. Охарактеризуйте таблицу как основной компонент базы данных. Дайте определение индексу. Какие категории индексов вам известны?
42. Что такое ядро базы данных? Что такое устройство баз данных?
43. Приведите синтаксис команды создания устройства.
44. Приведите пример команды создания базы данных. Какие системные и числовые типы данных MS SQL Server вам знакомы.
45. Приведите пример команды создания таблицы базы данных.
46. Удаление таблицы базы данных. Изменение определения таблицы.
47. Перечислите основные этапы создания базы данных в графическом режиме. Что понимается под ER – диаграммой в MS SQL Server.
48. Перечислите основные команды MS SQL для модификации данных.
49. Приведите пример использования команды INSERT для вставки записей в единственную таблицу.
50. Каким образом осуществляется вставка пустого значения в ячейку таблицы?
51. Приведите пример удаления единственной записи таблицы.
52. Приведите пример удаления множества записей.
53. Каким образом осуществляется удаление с вложенным подзапросом?
54. При помощи какой команды осуществляется модификация данных в MS SQL Server?
55. Приведите пример использования команды UPDATE с предложением FROM.
56. Приведите пример обновления множества записей.
57. Приведите пример обновления записей с подзапросом.
58. Приведите пример обновления записей нескольких таблиц.
59. Общая структура команды SELECT. Где и для каких целей применяется выборка? Как просмотреть результаты выборки?
60. Приведите примеры условий формирования условий отбора? Перечислите операторы, которые могут применяться в предложении WHERE?
61. Дополнительные возможности вывода в предложении SELECT.
62. Проверка на принадлежность диапазону значений. Проверка на

соответствие шаблону.

63. Составные условия поиска.

64. Дайте определения функции агрегирования. Перечислите функции агрегирования, опишите их назначение?

65. Какое предложение применяется для сортировки данных выборки, условия его применения? Каким образом осуществляется выборка из нескольких страниц? Приведите пример создания таблицы на основе выборки.

66. Скалярные функции SQL.

67. Агрегатные функции SQL.

68. Условные операторы в SQL.

69. Многотабличные и вложенные запросы. Соединение таблиц.

70. Дайте определения триггеру. Приведите примеры возможных триггеров? Описать синтаксис набора команд создания триггеров? Что необходимо учитывать при использовании триггеров?

71. Дайте определение соединению. Перечислите основные конструкции для формирования соединений.

72. Приведите пример использования конструкции JOIN. Приведите пример использования конструкции INNER JOIN.

73. Чем отличаются результаты выполнения соединений конструкций JOIN и INNER JOIN?

74. Перечислите общие свойства конструкции INNER JOIN и конструкции WHERE.

75. Дайте определение связующей таблице? Для чего и в каком случае используются связующие таблицы?

76. Приведите пример соединения с использованием конструкции OUTER JOIN.

77. Приведите пример соединения с использованием конструкции FULL JOIN.

78. Приведите пример соединения с использованием конструкции CROSS JOIN.

79. Соединение таблицы со своей копией.

80. Дайте определение представлению. Какие данные использует представление?

81. Приведите пример простого представления.

82. Как можно использовать представления для корректировки данных?

83. Перечислите последовательность действий для создания и редактирования представлений в программе Management Studio.

84. Дайте определение хранимой процедуре. Что понимается под системными хранимыми процедурами?

85. Что понимается под расширенными хранимыми процедурами? Приведите пример создания хранимой процедуры.

86. Дайте определение ограничению. Какие типы ограничений вы знаете?

87. Что такое ограничение домена?
88. Дайте определение ограничению сущности. Какие правила именования ограничений вы знаете?
89. Приведите пример ограничения primary key. Приведите пример ограничения foreign key.
90. Приведите пример таблицы, ссылающейся на саму себя. Приведите пример ограничения unique. Приведите пример ограничения check. Приведите пример ограничения default.
91. Дайте определение пакету данных. Приведите примеры команд, которые можно использовать в пакете и которые нельзя.
92. Поясните синтаксис блокировки данных. Какие виды блокировок данных вам известны? Приведите примеры.
93. Что такое уровень изоляции транзакции?
94. Приведите пример команды управления циклом.
95. Какая команда позволяет выходить из процедуры принудительно?
96. Защита данных. Управление доступом к данным.
97. Восстановление данных.

Дисциплина «Конструирование программного обеспечения»

1. Жизненный цикл ПО. Модели жизненного цикла.
2. Гибкие (Agile) методологии разработки. Scrum. Lean. XP. Kanban.
3. Подходы к разработке ПО. Структурный подход. Основные характеристики и особенности подхода.
4. Подходы к разработке ПО. Объектно-ориентированный подход. Основные характеристики и особенности подхода.
5. Подходы к разработке ПО. Аспектно-ориентированное программирование. Основные характеристики и особенности подхода.
6. Подходы к разработке ПО. Агентное программирование. Основные характеристики и особенности подхода.
7. Подходы к разработке ПО. Контрактное программирование. Основные характеристики и особенности подхода.
8. Декомпозиция и абстракция как основы уменьшения сложности.
9. Принципы объектно-ориентированного подхода. Абстрагирование.
10. Принципы объектно-ориентированного подхода. Инкапсуляция.
11. Принципы объектно-ориентированного подхода. Модульность.
12. Принципы объектно-ориентированного подхода. Иерархичность.
13. Принципы объектно-ориентированного подхода. Типизация и полиморфизм.
14. Принципы объектно-ориентированного подхода. Понятие класса и объекта. Сохраняемость и параллелизм.
15. Язык UML. Принципы языка и его назначение. Варианты (режимы) использования. Основные виды диаграммы.

16. Объекты. Состояние и поведение. Виды отношений между объектами. Ассоциация и агрегация объектов.

17. Классы. Виды отношений между классами. Ассоциации. Агрегация (has a), композиция (part of), использование (use) и наследование (is a).

18. Виды абстракций. Абстрактные классы и интерфейсы. Разработка, основанная на интерфейсах (Interface Based Development).

19. Обобщенное программирование. Параметризованные классы и методы.

20. Принципы объектно-ориентированного дизайна. SOLID. Принцип единственности ответственности (SRP).

21. Принципы объектно-ориентированного дизайна. SOLID. Принцип открытости/закрытости (OCP).

22. Принципы объектно-ориентированного дизайна. SOLID. Принцип замещения Лисков (LSP).

23. Принципы объектно-ориентированного дизайна. SOLID. Принцип разделения интерфейса (ISP).

24. Принципы объектно-ориентированного дизайна. SOLID. Принцип инверсии зависимостей (DIP).

25. Принципы объектно-ориентированного дизайна. Принцип ацикличности зависимостей (ADP).

26. Принципы объектно-ориентированного дизайна. Принцип Деметры (Law of Demeter, LoD) (ограниченности знания).

27. Паттерны проектирования. Понятие и виды паттернов. Принципы описания и применения паттернов.

28. Архитектурные паттерны. Разделение логики и представления. Архитектура MVC Passive Model.

29. Архитектурные паттерны. Разделение логики и представления. Архитектура MVC Active Model.

30. Архитектурные паттерны. Разделение логики и представления. Архитектура MVP Passive View.

31. Архитектурные паттерны. Разделение логики и представления. Архитектура MVP Supervising Controller.

32. Архитектурные паттерны. Разделение логики и представления. Архитектура MVVM (Presentation model).

33. Делегаты и события. Повышение косвенности с помощью делегатов.

34. Паттерны проектирования. Поражающие паттерны. Назначение и характеристики.

35. Паттерны проектирования. Структурные паттерны. Назначение и характеристики.

36. Паттерны проектирования. Поведенческие паттерны. Назначение и характеристики.

37. Принципы (паттерны) организации ПО GRASP. Low Coupling (Слабая связанность).

38. Принципы (паттерны) организации ПО GRASP. High Cohesion (Сильное сцепление/связность).

39. Принципы (паттерны) организации ПО GRASP. Protected Variations (Соккрытие реализации).

40. Принципы (паттерны) организации ПО GRASP. Indirection (Посредник).

Практическая часть:

1. Построить UML-диаграмму классов и привести пример кода реализации поведенческого паттерна «Интерпретатор (Interpreter)».

2. Построить UML-диаграмму классов и привести пример кода реализации поведенческого паттерна «Шаблонный метод (Template method)».

3. Построить UML-диаграмму классов и привести пример кода реализации поведенческого паттерна «Итератор (Iterator)».

4. Построить UML-диаграмму классов и привести пример кода реализации поведенческого паттерна «Команда (Command)».

5. Построить UML-диаграмму классов и привести пример кода реализации поведенческого паттерна «Состояние (State)».

6. Построить UML-диаграмму классов и привести пример кода реализации поведенческого паттерна «Стратегия (Strategy)».

7. Построить UML-диаграмму классов и привести пример кода реализации порождающего паттерна «Строитель».

8. Построить UML-диаграмму классов и привести пример кода реализации поведенческого паттерна «Наблюдатель (Observer)».

9. Построить UML-диаграмму классов и привести пример кода реализации поведенческого паттерна «Посетитель (Visitor)».

10. Построить UML-диаграмму классов и привести пример кода реализации поведенческого паттерна «Посредник (Mediator)».

11. Построить UML-диаграмму классов и привести пример кода реализации поведенческого паттерна «Цепочка обязанностей (Chain of Responsibility)».

12. Построить UML-диаграмму классов и привести пример кода реализации порождающего паттерна «Фабричный метод».

13. Построить UML-диаграмму классов и привести пример кода реализации порождающего паттерна «Абстрактная фабрика».

14. Построить UML-диаграмму классов и привести пример кода реализации структурного паттерна «Заместитель (Proxy)».

15. Построить UML-диаграмму классов и привести пример кода реализации структурного паттерна «Компоновщик (Composite)».

16. Построить UML-диаграмму классов и привести пример кода реализации структурного паттерна «Мост (Bridge)».

17. Построить UML-диаграмму классов и привести пример кода реализации структурного паттерна «Приспособленец (Flyweight)».

18. Построить UML-диаграмму классов и привести пример кода реализации структурного паттерна «Фасад».

19. Построить UML-диаграмму классов и привести пример кода реализации структурного паттерна «Адаптер (класса и объекта)».

20. Построить UML-диаграмму классов и привести пример кода реализации структурного паттерна «Декоратор».

Дисциплина «Тестирование программного обеспечения»

1. Понятие тестирования ПО. Цели и задачи тестирования. Связь с другими дисциплинами.

2. Принципы тестирования F.I.R.S.T. Правила написания программ, удобных для тестирования.

3. Понятие ошибки. Виды ошибок. Типовые ошибки и способы их нахождения.

4. Статическое и динамическое тестирование. Понятие отладки. Отличия отладки от тестирования.

5. Критерии отбора тестовых наборов. Функциональные критерии.

6. Критерии отбора тестовых наборов. Структурные критерии.

7. Критерии отбора тестовых наборов. Стохастические критерии. Особенности тестирования программ, связанных с работой со случайными процессами.

8. Критерии отбора тестовых наборов. Мутационный критерий.

9. Подходы к построению тестов. Тестирование методом «Черного ящика». Понятие класса эквивалентности. Анализ граничных условий.

10. Подходы к построению тестов. Тестирование методом «Белого ящика». Понятие управляющего графа программы (УГП).

11. Подходы к построению тестов. Тестирование методом «Серого ящика». Преимущества комбинированного подхода к построению тестов.

12. Автоматизация процессов тестирования. Понятие наборов тестов (Test Suite) и тестовых скриптов (Test Script/Test Case). Понятие тестового плана и тестового сценария.

13. Автоматизация процессов тестирования. Создание тестовых проектов в xUnit.

14. Тестирование, управляемое данными (Data Driven Testing). Способы организации DDT.

15. Метрики кода. Понятие покрытия программы (Test Coverage) и его назначение.

16. Восходящее и нисходящее тестирование. Особенности, преимущества и недостатки. Преимущества и недостатки комбинирования нисходящего и восходящего тестирования.

17. Модульное тестирование. Цели и задачи модульного тестирования.

18. Интеграционное тестирование. Цели и задачи интеграционного тестирования.

19. Понятие тестовых двойников (Test Doubles). Виды тестовых двойников.

20. Заглушки (Stub), их назначение и использование в тестировании.

21. Фиктивные объекты (Mock) , их назначение и использование в тестировании.

22. Системное тестирование. Цели и задачи системного тестирования.

23. Регрессионное тестирование. Цели и задачи системного тестирования. Периоды проведения регрессионного тестирования.

24. Разработка через тестирование (Test Driven Development). Принципы, достоинства и недостатки.

25. Подходы к классификации видов тестирования. Тестирование производительности. Стрессовое и нагрузочное тестирование. Alpha- и Beta-тестирование. Приемочное тестирование.

26. Подходы к классификации видов тестирования. Функциональное тестирование. Тестирование удобства использования (usability testing) и интерфейса пользователя (UI testing). Тестирование безопасности (security testing), локализации (localization testing) и совместимости (compatibility testing).

Практическая часть:

1. Составить набор тестов для метода, определяющего принадлежность точки заштрихованной области.

2. Составить набор тестов для метода, реализующего обработку с текстовой информации.

3. Составить набор тестов для класса, реализующего преобразование строки с числом из одной системы счисления в другую.

Варианты:

1. Составить набор тестов для класса, реализующего преобразование строки с числом из одной системы счисления в другую. Исходное основание:

2. Целевое основание: 10.

2. Составить набор тестов для класса, реализующего преобразование строки с числом из одной системы счисления в другую. Исходное основание:

8. Целевое основание: 10.

3. Составить набор тестов для класса, реализующего преобразование строки с числом из одной системы счисления в другую. Исходное основание: 8. Целевое основание: 16.

4. Составить набор тестов для класса, реализующего преобразование строки с числом из одной системы счисления в другую. Исходное основание: 10. Целевое основание: 2.

5. Составить набор тестов для класса, реализующего преобразование строки с числом из одной системы счисления в другую. Исходное основание: 10. Целевое основание: 8.

6. Составить набор тестов для класса, реализующего преобразование строки с числом из одной системы счисления в другую. Исходное основание: 10. Целевое основание: 16.

7. Составить набор тестов для класса, реализующего преобразование строки с числом из одной системы счисления в другую. Исходное основание: 16. Целевое основание: 8.

8. Составить набор тестов для класса, реализующего преобразование строки с числом из одной системы счисления в другую. Исходное основание: 16. Целевое основание: 10.

9. Составить набор тестов для метода, реализующего обработку с текстовой информацией, реализующий метод для удаления символов из заданного набора с конца строки (RTrim).

10. Составить набор тестов для метода, реализующего обработку с текстовой информацией, реализующий метод для удаления символов из заданного набора с обоих концов строки (Trim).

Дисциплина «Информатика и программирование»

1. Понятие информатики. Основная задача информатики.
2. Информация. Свойства информации.
3. Единицы представления данных.
4. Системы счисления.
5. Алгоритмы перевода чисел из одной системы счисления в другую.
6. История развития вычислительной техники.
7. Математические первоисточники вычислительной техники (Дж. Буль и В. Лейбниц)
8. Фундаментальные идеи и принципы построения компьютеров.
9. Аппаратные средства персонального компьютера.
10. Внутреннее устройство системного блока.
11. Назначение, классификация и конструкции периферийных устройств.
12. Организация файловой системы.
13. Алгоритм. Формы записи.

14. Структуры компьютеров.
15. Структура ПК с общей шиной.
16. Текстовые редакторы.
17. Табличные процессоры.
18. Система управления баз данных.
19. Языки программирования.
20. Основные принципы разработки программ.
21. Архитектуры вычислительных систем.
22. Компьютерные сети.
23. Системы искусственного интеллекта.
24. NET Framework как платформа разработки.
25. Компиляция исходного кода в управляемые модули.
26. Объединение управляемых модулей в сборку.
27. Задачи развертывания в .NET Framework.
28. Компоновка типов в модуль.
29. Объединение модулей для создания сборки.
30. Типы данных.
31. Именованные константы.
32. Операции и выражения.
33. Линейные программы.
34. Основные понятия объектно – ориентированного программирования.
35. Форматный вывод в консоль.
36. Основные операторы ввода-вывода, класс Math.
37. Операторы ветвления: условный оператор, оператор выбора switch.
38. Циклические конструкции.
39. Инициализация элементов одномерного массива.
40. Многомерные массивы.
41. Класс Random.
42. Базовый класс System.Array.
43. Базовый класс System. String.
44. Строки типа StringBuilder.
45. Методы.
46. Возврат управления из метода.
47. Возвращение методом значения.
48. Использование параметров с модификаторами ref и out.
49. Использование переменного количества параметров.
50. Перегрузка метода.
51. Работа с файлами. Поток байтов.
52. Работа с файлами. Асинхронный ввод-вывод.
53. Работа с файлами. Поток символов.
54. Работа с файлами. Двоичные потоки.

55. Работа с файлами. Консольный ввод-вывод.

56. Работа с файлами. Работа с каталогами и файлами.

Дисциплина «Дискретная математика»

Множества и шаблоны доказательств

1. Множества: основные определения (множество, элемент множества, способы описания множеств, множества N, Z, Q, R, \emptyset).

2. Основные определения: принадлежность к множеству, равенство множеств, соотношения между множествами, импликации, диаграммы Венна, универсальное множество.

3. Шаблоны доказательств: принадлежность к множеству, включение множества, отрицание включения множества, собственное включение множества, равенство множеств, неравенство множеств, импликации и двойные импликации.

4. Операции над множествами: объединение и пересечение (определения и теоремы).

5. Разность множеств, дополнение множеств и законы де Моргана. Булеан и произведение множеств.

6. Шаблоны доказательств: доказательство разбором случаев, опровержение контрпримером, доказательство от противного, не прямое доказательство.

7. Мощность множества. Теорема о подсчётах. Принцип включения-исключения (для двух, для трёх, для n множеств).

8. Математическая индукция. Шаблон построения доказательства по индукции. Примеры (с числами Фиббоначи).

Отношения и функции

1. Бинарные отношения и n -арные отношения.

2. Обратные отношения (определение и теорема).

3. Композиция отношений (определение и теорема).

4. Рефлексивные и антирефлексивные отношения (определения и теоремы).

5. Симметричные и антисимметричные отношения (определения и теорема).

6. Транзитивные отношения (определение и теорема).

7. Рефлексивные, симметричные и транзитивные замыкания (на примерах).

8. Отношение эквивалентности. Разбиения. Классы эквивалентности (определения).

9. Отношения частичного и линейного порядка (определения).

10. Определение функции. Функции как множества. Рекурсивно определённые функции.

11. Функции *в* (инъекции), *на* (сюръекции), *в* и *на* (биекции). Принцип Дирихле.

Комбинаторика

1. Принципы подсчета (принцип умножения, принцип сложения, подсчет элементов дополнения, использование принципа Дирихле).

2. r - перестановки n - элементного множества. Циклические перестановки.

3. Сочетания. Биномиальные коэффициенты.

4. Перестановки с повторениями (определение и теорема).

5. Сочетания с повторениями (теорема).

6. Комбинаторные тождества (тождество Ньютона, тождество Паскаля).

7. Биномиальная теорема. Треугольник Паскаля.

Теория графов

1. Основные определения (граф, смежность, инцидентность, степень вершины, полный граф (клика), регулярный граф, двудольный граф).

2. Подграфы. Остовный граф. Индуцированный граф. Гиперкуб.

3. Теорема о рукопожатиях. Теорема о степенях вершин и ребрах в графе. Теорема о числе нечётных вершин в графе (доказательство на выбор одно из трёх теорем).

4. Понятия маршрута, пути, длины пути, расстояния, контура, цикла и гамильтонова цикла.

5. Условия, которым удовлетворяют гамильтоновы графы. Задача о коммивояжёре.

6. Понятие изоморфизма графов. Примеры.

7. Представление графов: матрица смежности и списки смежности.

8. Понятие связности графа. Компоненты связности. Классы эквивалентности множества и компоненты связности графа.

9. Поиск в глубину. Алгоритм на примере.

10. Поиск в ширину. Алгоритм на примере.

11. Понятие эйлерова контура. Теорема Эйлера.

12. Понятие дерева с точки зрения теории графов. Теорема о характеристике деревьев.

13. Остовные деревья. Алгоритм Краскала. Нагруженный граф.

14. Корневые деревья. Деревья бинарного поиска. Обход дерева (прямой и обратный).

15. Ориентированные графы. Представление орграфов. Основные понятия.

16. Связность в ориентированных графах.