**КОНТРОЛЬНЫЕ ВОПРОСЫ К ЭКЗАМЕНУ**

**ПО ДИСЦИПЛИНЕ**

**«ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ»**

**2-Й КУРС, 4-Й СЕМЕСТР ОБУЧЕНИЯ**

**1–40 05 01-02 «Информационные системы и технологии (в экономике)»**

**1–40 05 01-08 «Информационные системы и технологии (в логистике)»**

**1–40 05 01-12 «Информационные системы и технологии (в игровой индустрии)»**

1. Реализация *smart*-указателей, свойств в языке, механизма транзакций.
2. Исключительные ситуации. Основные элементы механизма обработки исключительных ситуаций.
3. Методы обработки и генерации исключений.
4. Исключения. Задание собственной функции завершения.
5. Исключительная ситуация, генерируемая оператором *new*.
6. Разработка собственных классов обработки исключений.
7. Перехватывание и перенаправление исключений.
8. Повторная генерация исключения.
9. Обработка неожидаемых исключений.
10. Генерация исключений в конструкторах.
11. Исключения и наследование.
12. Спецификации исключительных ситуаций.
13. Иерархия исключений стандартной библиотеки.
14. Потоки ввода/вывода. Состояние потока. Общее понятие.
15. Иерархия классов потоков ввода/вывода. Механизм связывания потока с файлом. Пример, используя операции << и >> выполнить запись в файл и чтение из файла информации.
16. Строковые потоки их свойства, назначение.
17. Организация ввода из потока и вывода в поток на языке С++.
18. Перегрузка операторов вывода в поток и считывания из потока.
19. Контроль состояния потока и исправление ошибок. Пример обработки ошибки в потоке.
20. Неформатированный ввод-вывод.
21. Манипуляторы потоков (стандартные и определяемые пользователем).
22. Файлы и потоки их взаимосвязь.
23. Организация работы с файлами последовательного и произвольного доступа.
24. Организация ввода и вывода объектов.
25. Введение в стандартную библиотеку шаблонов (классов коллекций), основные понятия.
26. Классы контейнеры и итераторы. Разработка программ с использованием собственных контейнерных классов.
27. Типы контейнерных классов, адаптеры контейнеров.
28. Стандартная библиотека шаблонов (*STL*). Применение контейнеров *STL*.
29. Алгоритмы и их использование с контейнерными классами.
30. Назначение паттернов проектирования. Применение паттернов при решении прикладных задач.
31. Основные их виды.
32. Связь паттернов проектирования с ранее изученным материалом.
33. Применение ООП в решении экономических задач, задачах обработки информации и при выполнении инженерных расчетов.
34. Разработка программной поддержки процессов деятельности
35. *Proxi*-классы.
36. [Классы поддерживающие транзакции](#_Toc149321573)
37. Организация файла последовательного доступа.
38. Организация файла произвольного доступа.
39. Понятие итератор. Категории итераторов.
40. Основные и вспомогательные итераторы.
41. Пассивные и активные итераторы.
42. Контейнеры последовательностей.
43. Ассоциативные контейнеры.
44. Адаптеры контейнеров.
45. Алгоритмы.
46. Паттерны проектирования
47. Порождающие паттерны (паттерн Абстрактная фабрика).
48. Порождающие паттерны (паттерн Строитель).
49. Порождающие паттерны (паттерн Фабричный метод).
50. Структурные паттерны (паттерн Адаптер).
51. Поведенческие паттерны (паттерн Команда).