

Организация экзамена по курсу программирования: 2-й семестр 2015/16 уч. года (гр. 13501/1 – 13501/4)

Экзамен проводится в форме решения задачи с использованием компьютера. Студенту предлагается одна задача. Для программной реализации используется язык C++. Проект должен быть выполнен на основе объектно-ориентированной модели программирования. В задаче в явном виде предлагается определить отдельные классы и методы и организовать их взаимодействие. При этом следует исходить из того, что студент может определить и другие сопутствующие классы, методы, функции, необходимые, по его мнению, для решения задачи с учетом особенностей организации объектно-ориентированной программы.

На выполнение задания студенту дается 90 минут (2 ак. часа). Оценка выставляется в соответствии со следующими правилами:

Состояние проекта	Оценка
Программный проект представляет собой работающую программу, полностью реализующую заявленную функциональность. Проект реализован в соответствии с принципами объектно-ориентированного программирования. Определены необходимые пользовательские типы (структуры или классы). Для проверки корректности функционирования реализованы тестовые сценарии. Правильно использованы средства стандартной библиотеки C/C++. Обеспечена обработка возможных ошибок, возникающих в ходе работы программы. Проект характеризует хороший стиль программного кода. Студент в ходе собеседования может обосновать принятые проектные решения.	отлично
Программный проект представляет собой работающую программу, реализующую большую часть заявленной функциональности. Проект реализован в соответствии с принципами объектно-ориентированного программирования. Для проверки корректности функционирования реализован хотя бы один тестовый сценарий. Частично обеспечена обработка ошибок, возникающих в ходе работы программы. Студент в ходе собеседования может обосновать принятые проектные решения.	хорошо
Программный проект представляет собой работающую программу, реализующую большую часть заявленной функциональности в соответствии с принципами объектно-ориентированного программирования. Студент в ходе собеседования может обосновать основные проектные решения.	удовлетворительно
Программный проект не удается исполнить, однако рассмотрение проекта позволяет заключить, что разработчик следовал принципам объектно-ориентированного программирования. К большей части программного кода нет серьезных замечаний. Студент в ходе собеседования может обосновать основные проектные решения.	удовлетворительно
Программный проект не реализует поставленную задачу и не использует объектно-ориентированные средства языка C++. Студент затрудняется в объяснении проектных решений.	неудовлетворительно

Далее, применительно к задачам экзамена, приводятся основные темы, изученные в ходе семестра, и пример экзаменационной задачи.

Основные темы, рассмотренные в семестре 2

1. Повторение материала семестра 1: Объектно-ориентированный проект. Понятие об абстрагировании и инкапсуляции. Определение класса. Реализация методов класса. Взаимодействие объектов класса. Определение модульной структуры проекта в связи с его объектной ориентированностью.
2. Разработка класса для реализации динамического массива.
3. Обработка исключений.
4. Контейнеры типы: вектор, список, отображение. Возможные реализации. Особенности реализации контейнерных типов в стандартной библиотеке C++. Итераторы. Разработка алгоритмов обработки контейнеров.
5. Бинарный поиск. Поиск с использованием хеш-таблиц. Особенности применения в задачах обработки текстов.
6. Обобщенные типы. Шаблоны классов. Инстанцирование и специализация шаблонов.
7. Иерархии классов. Агрегирование и композиция. Наследование. Виртуальные функции. Абстрактные классы. Отношения зависимости. Реализация интерфейса. Особенности применения преобразований типов: статические и динамические преобразования.
8. Задача синтаксического анализа арифметических выражений и построение вычислителя выражений: объектно-ориентированный проект.
9. Организация тестирования компонентов объектно-ориентированного проекта. Понятие о модульном тестировании.
10. Задачи, формулируемые с использованием графов. Поиск кратчайшего пути в графе.

Пример экзаменационной задачи

Определите класс для работы с информацией о прямоугольнике на плоскости, задаваемом координатами вершин привязки (например, координатами верхнего левого угла прямоугольника) и размерами. Предполагаем, что стороны прямоугольника параллельны осям координат. Определите метод, определяющий, имеет ли данный объект общие точки с другим прямоугольником. Приведите пояснения к типам данных, которые использованы для переменных-членов класса. Напишите фрагмент программы (класс, функцию) для обработки массива сведений о прямоугольниках. В результате работы должен быть сформирован набор прямоугольников фигур, не имеющих общих точек с другими прямоугольниками. Поясните, чем Вы руководствовались, принимая решение, какой контейнер использовать для хранения набора сведений о фигурах и для формирования результирующего набора. Определите класс (классы) для тестирования разработанной функциональности.