

Семестр II

Задание 1. Простой класс на C# или Java

Классы на выбор:

- Время. Сложение, вычитание. Добавление минут, секунд, часов и т. п. Перевод времени в секунды, часы, минуты. Конвертирование в строку.
- Дата. Реализовать то же самое, что и для времени.
- Комплексное число. Операторы сложения, вычитания, умножения (на комплексное и действительное число). Вычисление аргумента и модуля.
- Кватернион. Аналогично комплексному числу.
- Вектор. Здаётся своими компонентами. Вычисление длины, углов между осями; операторы сложения и вычитания, умножения на число.
- Другой класс по согласованию с преподавателем.

1. Представить класс в виде UML диаграммы

2. Описать класс на C# или Java. Реализовать методы для доступа и изменения данных, конструктор с параметрами. Операторы и генерирование исключительных ситуаций если необходимо. Продемонстрировать работу с классом в приложении (GUI или консольное)

3. Для класса привести документацию описав его назначение, принципы использования, смысл методов и их параметров. Если необходимо привести пример использования класса в документации.

Вопросы

Что такое АДТ?

Что такое предусловия? Для чего нужны? Что такое постусловия?

Что такое класс? Что такое объект?

Что такое абстрагирование?

Что такое инкапсуляция? Что такое метод? Что такое конструктор? Что такое оператор?

Что такое принцип сокрытия? Что такое «чёрный ящик»?

Что такое поле класса?

Что такое свойство (C#)?

Как вызвать метод конкретного объекта находящегося в массиве?

Чем отличаются обращения к методам в C++ с использованием объекта, ссылки на объект и указателя на объект?

Что такое равенство объектов? Когда объекты идентичны?

Что такое поведение? Что такое состояние?

Задание 2. Виджеты Qt. Сигналы и слоты

Не обязательна. Если не выполнена максимальная оценка на экзамене — 4

Продemonстрировать механизм сигналов и слотов на примере виджетов Qt.

Например соединить 2-3 метода одного виджета с другим. Возможно использование лямбда функций.

Задание 3. Простой чат бот.

Программа с GUI. Любой объектно-ориентированный язык общего назначения на выбор. Бизнес-логика в отдельном модуле.

- Ответ на несколько реплик заданного шаблона.
- Ответ на простые команды. (Время, имя, статистика по обмену сообщениями). Ответ на команды с параметрами.
- Бот должен хранить историю сообщений, включая время отправки и автора.
- Записывать историю в файл при завершении программы. Загружать из файла при запуске программы.
- *Дополнительно (макс. Оценка на экзамене 3, если не выполнено):*
 - *Получение актуальной информации из интернета (погода, курсы валют, последние новости и т. п.)*
 - *Запуск отдельных программ, работа с операционной системой и файлами.*
 - *Сохранение информации о собеседнике. Собеседник предполагается неизменным*
 - *Опционально: показ изображений (в том числе загрузка из интернета, например APOD)*

Ссылки

<https://stackoverflow.com/questions/46943134/how-do-i-write-a-qt-http-get-request>

Задание 4. Простая БД

Простая файловая БД с GUI. Любой объектно-ориентированный язык общего назначения на выбор.

- Одна таблица с 4+ полями.
- Собственный формат БД (без использования SQL, noSQL и проч.)
- Добавление, изменение данных
- Поиск, сортировка (как минимум по одному полю).
- Документация (в коде) описывающая формат данных в файле.
- Требования к GUI:
 - Вывод данных в таблицу
 - Меню приложения
 - *Дополнительно: горячие клавиши*

Задание 5. Игра

(Не обязательно. Если выполнено +1 к оценке на экзамене)

Любой объектно-ориентированный язык общего назначения на выбор.

- Возможные варианты
- Игра «Жизнь» Конвея, Сапёр, Тетрис, арканойд
- Свой вариант (по согласованию с преподавателем).
- Программа для моделирования роботов на двумерной плоскости. Схематичное отображение. Робот может перемещаться на клетку, поворачивается, атаковать или собирать ресурсы лежащие на плоскости. Поведение робота определяет класс с заданной спецификацией). *Дополнительно: выполнение каждого класса в отдельном потоке.*

Задание 6. UI markup language

(Не обязательно. Если не выполнено макс. оценка на экзамене - 4)

Простое приложение с GUI построенным с использованием языка разметки UI (QML, XAML и др)

Задание 7. Приложение готовое к развёртыванию

(Не обязательно. Если не выполнено макс. оценка на экзамене - 4)

Создать установочный файл (Windows, MacOS, Linux) для любого из ранее созданных приложений.

Для создания инсталлятора использовать готовые инструменты.

Дополнительно: использовать цифровую подпись.

Задание 8. Юнит-тест

(Не обязательно. Если не выполнено макс. оценка на экзамене - 3)

Набор юнит тестов для ранее созданного класса (классов)

Пример: https://github.com/VetrovSV/OOP/tree/master/simple_class/tests