Всего 7 лабораторных работы:

1. базовый графический интерфейс;
2. продвинутый графический интерфейс;
3. многооконные приложения и сообщения об ошибках;
4. передача данных по последовательному (serial) порту;
5. подключение базы данных;
6. составление базовых диаграмм для финального проекта;
7. финальный проект.

Перед тем, как описывать подробнее каждую отдельную лабораторную работы напишу немного своих мысли. Во – первых, все 6 первых лабораторных работы будут направлены на решение итоговой 7 задачи, включающей в себя по сути все полученные до этого знания. Во – вторых, каждая (скорее всего каждая) задача будет содержать в себе дополнительные условия на повышенную оценку, так как общая сложность всех задач, кроме работы с базой данных и соответственно финального проекта, не очень высокая. В – третьих, сами задания не должны быть слишком сложные, чтобы не отбивать у студентов мотивацию их делать, но при этом достаточно тяжелыми, чтобы пришлось потратить неделю – полторы на решение всех 7ми задач с ноля и соответственно не отбивать мотивацию своей простотой. В – четвертых, нужно выработать общий критерий оценки данных заданий, и примеры программ. Мое личное мнение – не запрещать пользоваться чем удобно студентам, и может быть даже приводить примеры библиотек и сред для основных языков (python, java), но при этом настаивать на выполнении всех задач на С/С++ с или без использования Qt. Я мотивирую это тем, что такого рода задания напрямую связаны с низкоуровневым программированием, поскольку многие встраиваемые системы программируются а C/С++. Таким образом эффективно использовать сейчас и в дальнейшем, при программировании встраиваемых решений С\С++. Даже если люди не будут заниматься таким программирование после курса, знание низкоуровневых языков не будет лишним. В – пятых, как уже замечено выше для предотвращение чрезмерной сложности заданий я все же предлагаю отказаться от более сложных интерфейсов, таких как modbuss, так как они являются не очень простыми в освоении, при том факте, что вряд ли понадобится даже 10% студентов в последующей учебе, в то время как хорошее понимание последовательного RS232 может быть очень интересным, так как задачи будут связаны с вполне легкоосвоимой платформой Arduino, по которой в 6ом семестре Олег Анатольевич дает свои лабораторные. Однако теорию по основным промышленным интерфейсам modbuss, profibas, profinet на лекциях вполне можно дать, что будет хорошим плацдармом для понимания МПвСУ в дальнейшем.

На счет критерия оценки, учитывая, что лабораторные работы будут не совсем стандартные касательно того, что было раньше, нужно, чтобы Дима хорошо понимал, о чем идет речь и мог помог студентам в решении проблем, особенно если методички в этом семестре не будет. Оценка, которую получит студент должна зависеть от наполнения программы, выполнены ли основные задачи (до 6 баллов) отвечены ли дополнительные вопросы (еще 2 балла) выполнены ли дополнительные задачи (еще 2 балла). Разброс ценности отдельных элементов задания для разных лабораторных работ может варьироваться. Например в первых лабораторных работах и основное и дополнительное задание максимально простые, что приводит к логичному выводу сделать больше цену дополнительных заданий, чтобы потом, в следующих лабораторных студенту не пришлось возвращаться к дополнительным заданиям первых лабораторных и доделывать их просто для понимания.

1. **Базовый графический интерфейс**

**Описание**: Лабораторная, которая должна быть довольно простой из расчета на то, что 90% студентов никогда не писали графического интерфейса, а значит они должны познакомиться с основными элементами ввода вывода (кнопки, чекбоксы, списки, слайдеры)

**Пример задания 1): (общая концепция варианта – рабочая станция контроллера деталей на заводе)** Написать однооконное приложение содержащее 3 взаимосвязанных чекбокса (может быть выбран только один из 3) выбора материала детали (сталь, бронза, алюминий), два списка: первый список с выбором типа детали(втулка, шестерня, вал), второй список с выбором диаметра посадочного отверстия(10, 12, 15 мм). На экране должно присутствовать 2 кнопки: «сохранить», «сбросить». Кнопка сохранить должна срабатывать только при заполнении всех характеристик детали, срабатывание кнопки «сохранить» заключается в выводе информации о детали в текстовый элемент на экране. Кнопка «сбросить» сбрасывает все характеристики детали. Отображать в текстовом элементе количество сохраненных деталей.

Дополнительное задание 1: Вместе с выводом на экран организовать сохранение данных о деталях в файл.

Дополнительное задание 2: Вместе с данными о деталях записывать в файл системное время добавления детали.

Критерий оценки: реализовано полное условие без дополнительных заданий – 6 + 2 балла за ответы на вопросы + 1 балл за каждое дополнительное задание.

**Пример задания 2): (общая концепция варианта – система контроля посещаемости здания)** Написать оконное приложение содержащее 2 взаимосвязанных чекбокса (может быть выбран только один из 2: вошел, вышел) и один список (номер от 1 до 10 – условные идентификаторы сотрудников). Так же на экране должно присутствовать 2 кнопки: «подтвердить», «отменить». При нажатии кнопки «подтвердить» в текстовом элементе должна появляться надпись с информацией о посетителе. При попытке одного и того же человека войти или выйти из здания 2 раз подряд в тестовом элементе должна появляться ошибка с соответствующей проблемой. При нажатии кнопки «отменить» очищает информацию о посетителе. Отображать в текстовом элементе количество людей присутствующих сейчас в здании.

Дополнительное задание 1: Вместе с выводом на экран организовать сохранение данных о посещениях в файл.

Дополнительное задание 2: Вместе с данными о посещении записывать системное время посещения.

Критерий оценки: реализовано полное условие без дополнительных заданий – 6 + 2 балла за ответы на вопросы + 1 балл за каждое дополнительное задание.

1. **Продвинутый графический интерфейс**

**Описание**: Работа рассчитанная на дополнение предыдущей работы с использованием дополнительных стандартных графических элементов (поля текстового вводя, радиокнопки)

**Пример задания 1):** Заменить список с диаметрами посадочных отверстий на элемент текстового ввода. Проверять ввод на соответствие стандартному (8, 10, 12, 15, 17 мм). Если введен неправильный диаметр или введенная информация – не цифровая, в текстовом поле писать соответствующую ошибку. Добавить элемент текстовое поле ввода в которое должно вписываться условное число деталей в партии, осуществлять проверку на неправильный ввод информации о количестве деталей в партии. Добавить элемент - радиокнопка с надписью «тестиование». При активном состоянии радиокнопки «тестирование» после нажатия кнопки «сохранить» должна появляться ошибка, если в партии указано более 10 деталей, так же, если к прошлому заданию реализована запись файл, то при активном «тестировании» запись о партии в файл осуществляться не должна.

Дополнительное задание 1: Добавить текстовое поле отображающее 3 последние партии с учетом тестовых.

Дополнительное задание 2: Добавить возможность отображать из файла суммарное количество деталей и их виды (статистические данные) в текстовом элементе.

Критерий оценки: реализовано полное условие без дополнительных заданий – 6 + 2 балла за ответы на вопросы + 1 балл за каждое дополнительное задание.

**Пример задания 2):** Добавить текстовое поле ввода имени посетителя, которое должно заполняться только при первом посещении здания работником, после последующего покидания здания ввод осуществляться не должен, система должна выводить в текстовое поле имя вошедшего или вышедшего человека. При условии реализации дополнительных заданий к работе 1 добавить отдельный текстовый документ, хранящий соответствие имени и идентификатора работника. Добавить радиокнопку «инкассация», при ее активном состоянии следующее нажатие «подтвердить» на вход не должна потребовать идентификатора, но должна засчитать присутствие посетителя. Так же должно произойти и на выход. В здании в один момент может быть лишь один посетитель с флагом «инкассация».

Дополнительное задание 1: Отображать в текстовом поле на экране идентификаторы посетителей находящихся в здании на текущий момент.

Дополнительное задание 2: В элемент список добавить пункт «сторонний посетитель» после выбора и подтверждения которого только вошедшему человеку присваивается уникальный идентификационный номер по которому этот человек и выходит, имя посетителя заполняется каждый раз при входе, идентификационный номер добавляется в список и убирается после того, как человек вышел.

Критерий оценки: реализовано полное условие без дополнительных заданий – 6 + 2 балла за ответы на вопросы + 1 балл за каждое дополнительное задание.

1. **Многооконные приложения и сообщения об ошибках**

**Описание**: Основная цель лабораторной работы – научить студентов создавать несколько окон для приложений, осуществлять навигацию между ними, создавать со окна подтверждения действия и окна с ошибками программы.

**Пример задания 1):** Заменить связанные чекбоксы выбора материала на список. Все обрабатываемые ошибки должны быть переделаны на диалоговые окна с текстом соответствующей ошибки и кнопкой «ок». Добавить на основной экран кнопки «добавить материал» и кнопку «добавить деталь». При нажатии на эти кнопки должно появляться диалоговое окно с текстовым полем, в которое пользователь должен вписать соответствующее название. На появляющемся окне должно быть две кнопки: «сохранить» и «отмена». Нажатие на кнопку отмена должно закрыть окно. Если при нажатии на кнопку «сохранить» текстовое поле ввода осталось пустым должно появиться окно с текстом ошибки и кнопкой «оk», После нажатия на которую окно добавление не закрывается и остается возможность заполнить название. При правильном вводе информации новые материал или деталь должны появиться в соответствующих им списках.

Дополнительное задание 1: добавить еще одно окно в котором отображать лог проходящих деталей. Соответственно убрать текстовое поле основного экрана, отображающее 3 последние группы деталей, если оно реализовано.

Дополнительное задание 2: добавить всплывающее окно подтверждения действия при нажатии на основном окне кнопки «отменить», предупреждающее о том, что заполненные данные будут потеряны, и кнопками «ок» и «назад», работающее соответствующим образом.

Критерий оценки: реализовано полное условие без дополнительных заданий – 6 + 1 балл за ответы на вопросы + 3 балла за два дополнительных задания (2+1).

**Пример задания 2):** заменить чекбоксы на радиокнопки. Все обрабатываемые ошибки должны быть переделаны на диалоговые окна с текстом соответствующей ошибки и кнопкой «ок». Добавить на основной экран копки «добавить сотрудника в список» и «убрать сотрудника из списка». При нажатии на кнопку «добавить сотрудника в список» должно появляться диалоговое окно предлагающее ввести идентификатор сотрудника (но поле должно быть автоматически заполнено первым свободным идентификатором) и имя и должно содержать две кнопки: «сохранить» и «отмена». При нажатии на кнопку «отмена» окно должно закрытья. При нажатии кнопки «сохранить» новый сотрудник должен быть добавлен в файл (если есть предыдущие дополнительные задания) и в списки. После чего должна появиться возможность входить и выходить новому сотруднику. Если, при нажатии «сохранить», используется уже занятый или текстовый идентификационный номер, а так же поле заполнения имени осталось пустым, должны появляться соответствующие ошибки. Кнопка удаления сотрудника должна работать а таком же ключе.

Дополнительное задание 1: Добавить еще одно окно в котором отображать лог находящихся в здании людей. Соответственно убрать текстовое поле отображающее соответствующий лог.

Дополнительное задание 2: Добавить диалоговое окно подтверждения действия при нажатии на основном окне кнопки «отменить», предупреждающее о том, что заполненные данные будут потеряны, и кнопками «ок» и «назад», работающее соответствующим образом.

Критерий оценки: реализовано полное условие без дополнительных заданий – 6 + 1 балл за ответы на вопросы + 3 балла за два дополнительных задания (2+1).

1. **Передача данных по последовательному порту**

**Описание**: Лабораторная работа заключается в правильном обнаружении устройства, подключенного к COM-порту компьютера и правильному приему/передаче данных. Для всех 15 заданий будет написан свой протокол и я думаю, что у Димы будет одна arduino с нужной обвязкой, способная работать с любым придуманным протоколом для того, чтобы ребята могли протестировать свои программы, так как без физического устройства программы не имеют особого смысла. И вообще все эти протоколы и примеры можно выложить на какой-нибудь GitHub кафедры систем управления, который можно создать.

**Пример задания 1):** Обеспечить обработку последовательного интерфейса. В контексте данной задачи контроллер осуществляет «проверку» деталей на брак. Для простоты все пересылаемые команды и значения имеют тип byte либо char. В начальный момент времени контроллер ожидает запроса. Флагом запроса является передача символа ‘s’. После которого следует передать количество деталей (от 1 до 255) .То есть запрос состоит из 2 последовательно идущих байт. Ответ состоит из флага ответа – символа ‘a’ и количества деталей (от 1 до 255 в зависимости от того, сколько было передано с запросом), которые прошли проверку. Соответственно нужно к написанному приложению подключить обработку последовательно интерфейса. Теперь у каждой партии должен быть флаг проверки, который записывается в файл(если запись в файл реализован) и является идентификатором того, что партия прошла пункт контроля. Флаг проверки получается партией, если в ней более 90% качественных деталей, если меньше – у партии должен появиться флаг брака. Таким образом после нажатия кнопки сохранить, должна происходить связь с контроллером, после получения ответа необходимо определить процент брака и выдать соответствующий флаг.

Дополнительное задание: реализовать удаление из файла или не записывать в файл все партии, не прошедшие проверку. Для записи данных партий создать дополнительный файл в котором вместе со всей информацией хранить так же количество прошедших проверку деталей и процент брака партии.

Критерий оценки: реализовано полное условие без дополнительных заданий – 7 + 2 балла за ответы на вопросы + 1 балл за дополнительное задание.

**Пример задания 2):** Обеспечить обработку последовательного интерфейса. В контексте данной задачи контроллер осуществляет «генерацию» идентификационного номера работника. Для простоты все пересылаемые команды и значения имеют тип byte либо char. В начальный момент времени контроллер ожидает запроса. Флагом запроса является передача символа ‘s’. После которого следует передать количество зарегистрированных работников (от 1 до 255). Ответ состоит из флага ответа – символа ‘a’ и порядкового номера сотрудника (от 1 до 255. Имеется ввиду что при возврате, например, цифры 8, имеется ввиду идущий 8 в списке сотрудник, и если например отсутствует сотрудник с номером 3, то подразумевается сотрудник с номером 9, как идущий 8ым в списке). Теперь идентификационный номер нужно вводить только тогда, когда сотрудник выходит из здания. Связь по протоколу осуществляется после нажатия кнопки подтвердить, при условии, что в здание зашел сотрудник. После получения ответа необходимо определить соответствующий идентификационный номер сотрудника, после чего добавить его в список присутствующих.

Дополнительное задание: Учитывая, что ответом контроллера в целом является случайное число, реализовать выход посетителей с запросом на контроллер. То есть на контроллер отправляется число присутствующих сейчас в здании людей, не зависимо от группы, полученный ответ интерпретируется как порядковый номер человека в здании, где все люди отсортированы с меньшего к большему по времени, которое они находятся в здании.

Критерий оценки: реализовано полное условие без дополнительных заданий – 7 + 2 балла за ответы на вопросы + 1 балл за дополнительное задание.

1. **Подключение базы данных**

**Описание**: Учитывая, что базы данных не являются основным предметом курса, как в прочем и низкоуровневые интерфейсы, задание не будет очень сложным. Студенты могут использовать любую СУБД. Хотя рекомендацией скорее всего будет SQLIte, как легкая в освоении и легко разворачиваемая на локальной машине.

**Пример задания 1):** Подключить к приложению базу данных. Хранить в базе данных основной документ содержащий прошедшие контроль детали, документ хранящий материалы и виды деталей и документ содержащий бракованные детали с соответствующим их заполнением вместо текстовых файлов.

Дополнительное задание: Осуществить по своему усмотрению возможность перевода партий из бракованных в нормальные (да, теневая бухгалтерия).

Критерий оценки: реализовано полное условие без дополнительных заданий – 7 + 1 балла за ответы на вопросы + 2 балл за дополнительное задание.

**Пример задания 2):** Подключить к приложению базу данных. Хранить в базе данных основной документ содержащий соответствие имен и идентификаторов работников, документ содержащий всех людей, находящихся в текущий момент в здании.

Дополнительное задание: Осуществить по своему усмотрению возможность изменения идентификационного номера сотрудника, записанного в базе данных.

Критерий оценки: реализовано полное условие без дополнительных заданий – 7 + 1 балла за ответы на вопросы + 2 балл за дополнительное задание.

1. **Составление базовых диаграмм для финального проекта**

**Описание**: Данная работа будет уже содержать финальное задание, но основной целью этой работы будет составление двух диаграммы: последовательностей и компонентов для защиты финального проекта.

**Пример задания 1):** Рабочая станция контроллера деталей на заводе. Основные действующие элементы: компьютер (управляемый контроллером(человеком)), контроллер системы тестирования деталей, база данных предприятия. Человек получает партию деталей и регистрирует ее в системе, запись о деталях попадает в базу данных (в запись должны выходить: количество деталей, материал, тип, диаметр посадочного отверстия, время регистрации партии, условный номер приемного пункта(числовая константа не меняющаяся в программе), номер партии) и журнал работ(описание журнала следуют далее по тексту). После регистрации может быть осуществлен прием новой партии или отправка партии на диагностику, протокол которой описан в работе 4. После диагностики результат записывается в журнал работ и базу данных. После диагностики партия прошедшая проверку может быть отправлена на следующий этап производства о чем делается запись в журнал, так же время отправки добавляется в базу данных деталей. В процессе работы можно в базе данных так же должны храниться материалы, размеры и виды деталей с возможности их обновления (добавления и удаления)).

Журнал работ представляет собой текстовый документ в котором с временными метками описаны ключевые действия в системе (лог работы системы). Запись о поступлении может иметь вид: «12:31. Поступление деталей. № партии: 25, материал: бронза, тип: втулка, посадочный диаметр…», «10:54. Партия №12 прошла проверку, процент бракованных деталей : 11.3, результат: партия бракована.» и так далее. Документ должен быть удобен в прочтении человеком и может оформляться по усмотрению студента.

Дополнительное задание: можно дать дополнительный документ к оформлению или не давать вообще.

**Пример задания 2):** Система контроля посещаемости здания. Основные действующие элементы: компьютер (управляемый оператором), контроллер считывая идентификационного номера, база данных, посетители. Посетители бывают 3 типов: работники, гости, инкассация. Работники разделены по двум разделам, что должно быть отражено в их идентификационных номерах. Гости получают временные идентификационные номера на время пребывания в здании. Информация о всех работниках должна храниться в базе данных, дополнительной характеристикой работника будет суммарное время, проведенное в здании, которое должно обновляться каждый раз, когда работник уходит. Так же необходимо реализовать возможность добавления, увольнения и перевода работников из одного отдела в другой. При посещении здания работником определяется его идентификационный номер через протокол, описанный в работе 4, с поправкой на то, что номер должен содержать отдел, в котором работает сотрудник. Запись о посещении записывается в лог (описание лога далее по тексту). Текущее количество людей в здании и их данные должны отображаться в одном из окон приложения. При выходе сотрудника так же определяется его идентификационный номер, делается запись в лог и обновляются данные в БД.

Лог представляет собой текстовый документ с последовательными записями. Например: «9:31. Вася Пупкин пришел на работу, идентификационный номер 2-5», «10:48. Николай Балалай вошел в здание, временный идентификационный номер 45», «15:59. Неожиданно завалилась инкассация, Валера, быстро прячь героин», «16:10. Инкассация покинула здание, на этот раз пронесло.»)

Дополнительное задание: можно дать дополнительный документ к оформлению или не давать вообще.

**Пример задания 1/2 - 7): Условие общее для всех вариантов.** По своему усмотрению реализовать задачу, описанную в работе 6. Для реализации могут быть использованы любые графические элементы, удобные студенту. Плюсом к сдаче лабораторной работы будут инициированные самим студентом дополнения к программе описанной в задании.

Протоколы в задаче с последовательным портом очень тривиальные и заключаются в бросании случайных чисел внутри контроллера. Если нужно, можно их усложнить, хотя я не думаю, что в этом есть острая необходимость.

На счет примеров к задачам, можно или сделать 16 вариант и его полное решение на Qt, либо просто показать к каждой лабораторной работе добавление и работу с основными в этой работе элементами.

Не смотря на то, что я говорил в начале, что работы должны быть достаточно простыми, мне сейчас, в конце написания данного документа они не кажутся очень простыми, но полное завершение всех 7 работ даст студентам довольно интересный опыт программирования.

Если вы согласны с такой постановкой заданий. То мне нужен ответ касательно того, каким образом должны быть реализованы примеры и сколько вариантов заданий вы в итоге хотите видеть.