**Классы**

Темы для отработки: класс, объект, инкапсуляция, конструктор, деструктор

**Задание**

На основе предложенного описания составить систему классов и привести пример работы системы.

**Дополнительные баллы**

Начисляются, если в программе будет сохранение состояния программы в файл. Т.е. если выбрать опцию "сохранить", закрыть программу, а потом вновь ее открыть, то вся информация в программе востановится из файла. Этот процесс называется сериализация и дессериализация.

**Подсказка**

ООП стремится представить программную модель мира как можно ближе к реальности.

* Существительные - это классы
* Глаголы - методы классов
* Прилагательные (и другие определения и дополнения в предложении) - свойства классов.

Пример выполненного (ну или почти выполненного - работа в процессе) задания: [Зоопарк](https://github.com/StriderAJR/StudentCpp/blob/master/source/OOP_TaskExample_5_Classes.h)

**Что будет оцениваться**

Оцениваться в первую очередь будет то, как составлена система классов, насколько она грамотно продумана, хорошо ли соблюдены принципы ООП.

Пример работы программы - по сути творческая часть задания. Это может быть как что-то простое (обычная демонстрация создания объектов и работы методов), так и что-то интересное, творческое. Например, моделирование работы месяца работы банка/бара/магазина/жизни несколькоких поколений семьи и т.д.

За очень оригинальные примеры работы программы могут начисляться поощрительные баллы.

**Критерии оценивания**

1. Правильность системы классов для конкретного задания. Тут важно, что она была логична, похожа на модель реального мира и насколько учтены все описанные в моменте сценарии и особенности ситуации
2. Соблюдение принципов ООП. Инкапсуляцию, правильное использование конструкторов, деструкторов
3. Не забывать про правильную работу с памятью. Даешь каждому new свой delete. Нужно убирать за собой мусор.
4. Хороший стиль программирования

* No Pascal style
* По имени класса всегда можно понять, что их реального мира он олицетворяет
* Имена переменных должны быть "говорящими"
* Отступы
* ЗАЩИТА ОТ ДУРАКА

**Варианты**

**Ассоциация баров**

Вы входите в ассоциацию владельцев баров. При регистрации в ассоциации каждый владелец бара должен сообщить свое полное ФИО, паспортные данные и адрес проживания. При регистрации заведения, каждый бар должен обязательно иметь броское имя и адрес, где располагается, и является ли бар семейным (в котором запрещен алкоголь) или же нет.

Посещая бар, посетитель оставляет там энную сумму денег, которая поступает в кассу. При оплате клиент может предъявить именную карту постоянного клиента (при этом в кассе будет зафиксирован идентификационный номер этой карты) или остаться анонимным посетителем.

Вся информация о посещаемости и кассовых сборах передается в ассоциацию. В конце каждого месяца ассоциация помогает владельцам баров составить сводную информацию о выручке и посещаемости по каждому бару в отдельности и о сети баров каждого владельца в целом. В статистике отражаются: самый прибыльный день месяца, средний чек за месяц, какой бар пользуется наибольшей популярностью у посетителей, а также увеличивается или уменьшается популярность каждого бара со временем.

**Банк**

Каждый банк работает с клиентами. Клиенты бывают 2-х типов: физическое лицо и организация. Любой клиент может выпускать неограниченное кол-во карточек и заводить неограниченное кол-во вкладов. Также клиент может брать у банка кредиты под разные проценты. Причем у физических лиц меньше процент по вкладам, но и меньше процент по кредитам, нежели у организаций.

Карточка может быть дебетовая или кредитная. Баланс дебетовой карты не может быть отрицательным. Вклады различаются процентной ставкой, наличием капитализации процентов и возможностью частичного снятия средств до завершения срока вклада. У каждой карты есть привязанный к ней вклад.

Расплачиваясь карточкой, баланс карты уменьшается. А на вклад можно ложить деньги через кассу.

Банк никогда не работает в убыток. Кол-во денег получаемых от кредитов должно быть как минимум на 20% больше, чем выплачиваемые клиентам проценты по вкладам. Иначе процентная ставка у вновь открываемых вкладов снижается, а процент по кредитам растет.

Какие-то банки предоставляют лучшие условия кредитования, другие - проценты по вкладам. Все банки разные. Но каждый хочет зарабатывать побольше денежки.

**Поликлиника**

В каждом районе города есть поликлиники. В каждой поликлинике есть несколько подразделений: регистратура, лаборатория, администрация и приемный покой, в котором работают врачи.У врача есть ФИО, номер кабинета, в котором он ведет прием, его расписание приема (день недели и время приема) и его направление специализации: ЛОР, хирург, терапевт, психиатр и т.д.

Чтобы записаться на прием к врачу, пациент обращается в регистратуру, где его записывать на прием к терапевту по выбору. Во время записи пациент выбирает удобное время и день приема из доступных.

Придя на прием к терапевту, тот выслушивает жалобу пациента и выписывает сдать определенный перечень анализов. Причем анализы можно сдавать только в определенное время, и готовы они будут только через несколько дней. Когда анализы будут готовы, пациент снова должен прийти в регистратуру и записаться на прием к терапевту. На повторном приеме терапевт поинтересуется об изменении состояния пациента, посмотрит и анализы и на основе всех симптомов и анализов направляет пациента к нужному специалисту, опять же давая время и день из доступных у этого специалиста.

**Прим.**: попробуйте сделать все выше сказанное как можно веселее.

С вероятностью 50% пациента направляют к неправильному специалисту, который не найдет у пациента ничего по своему профилю, выпишет дорогие витамины и попросит прийти через месяц. Через месяц ситуация с записью к терапевту через регистратуру и анализами повторяется.

Если пациент не получает решения проблемы трижды, то он идет в администрация и нехорошими словами разговаривает с администрацией в поликлинике. После этого его сразу же отправляют к правильному специалисту и вылечивают.

**Генеалогическое древо**

Есть человек. У человека есть имя, дата рождения, флаг жив ли этот человек, дата смерти, пол. Человек может вступать в брак, разводиться, заводить неограниченное кол-во детей (но у женщины кол-во детей не может превышать кол-ва прожитых лет минус 18), также дети могут рождаться вне официально заключенного брака, а также детей можно усыновлять.

У каждого человека, если спросить, кем ему является тот или иной человек, сможет ответить являются ли они родственниками и какая степень родства. А такжже каждый человек может изобразить свое генеалогическое древо на 2 поколения выше (бабушки/дедушки, прабабушки/прадедушки, дети, если есть, и внуки, если есть).

**Цветочный магазин**

Цветочный магазин принимает заказы от своих клиентов. В каждом заказе фигурирует как минимум один букет, составленный из различных цветов. У каждого цветка есть название вида, цвет, дата, когда его срезали и закупочная цена, к которой магазин потом добавляет свою наценку. Магазин скрывает от своих клиентов дату срезания, однако знает, что нельзя поставить клиенту полностью завядшие цветы. Но и совсем свежие цветы поставлять тоже невыгодно. Поэтому магазин всегда знает, когда тот или иной цветок начнет увядать. Причем разные виды цветов начинают увядать на разный номер дня. Например, розы вянуть на 4 дней после срезания, а тюльпаны на 3 (цифры взяты с потолка).

Магазин может держать на складе только строго определенное кол-во цветов. Поэтому закупать свежие цветы, пока не проданы старые цветы - нельзя. А завядшие цветы нельзя продавать клиентам.

Клиент делает заказ в магазине и оплачивает свой букет.

Смоделируйте работу магазина так, чтобы он не работал себе в убыток.

**Прайс товаров**

У каждого уважающего себя магазина есть интернет-магазин с прайсом товаров. У каждого товара есть название, цена, срок изготовления, кол-во на складе, а также категория, к которой относится этот товар. Один товар может входить в несколько категорий. Причем, если изменить товар в одной категории, то он должен измениться во всех категориях.

Администратор магазина может создавать новые категории, добавлять туда новые товары, редактировать категории и товары. Пользователь же, заходя в магазин, может просмотреть как полный перечень товаров, так и каждую категорию по отдельности. Пользователь может сортировать товар по названию, цене и сроку изготовления. Также в магазине есть поиск по названию (если ввести часть названия, то найдут все товары, куда входит искомая строка).

Заходя в магазин, пользователь формирует заявку на доставку товаров. Автоматизируйте работу магазина, чтобы пользователь не смог заказать товар, которого нет на складе.

**This Is War!**

Два государства - Неверляндия и Пиратомания находятся в состоянии войны. У Неверляндии есть генерал - Джон Сковородка, а у Пиратомании - Генерал Гвоздь. У каждой армии может быть только один генерал. Если генерал погибнет, то война будет проиграна. Поэтому все юниты стараются защитить своего генерала.

Юнит неверлядии - хулиган, а Пиратомании - пират. Хулиганы быстрые (передвигаются на 2 клетки за раз) и могут бить из рогаток на расстояние до 10 шагов (но если подойти к ним ближе, чем на 3 шага, то пугаются и становятся беспомощными), но быстро устают и им нужно отдыхать после каждой перебежки 2 хода. Пираты же ходят медленее (1 клетка за раз) и могут бить врага только в радиусе 2 клеток от себя.

Генералы всех армий ходят по 1 клетке за раз, но бьют в радиусе 3 клеток вокруг себя. И помните, если генерал погибнет, то война закончится. Запустите симуляцию игры и постарайтесь, чтобы каждый раз юниты придерживались разных стратегий.

**Аэропорт**

Некоторые говорят, что аэропорт - это живой организм. В нем есть несколько подразделений: стойки регистрации и диспетчерская. И не стоит забывать про сами самолеты и пассажиров. Самолет отличается классом: грузовой или пассажирский и вместимостью по весу.

Для того, чтобы лететь самолетом, пассажир покупает билет, указывает откуда и куда он хочет лететь, выбирает из доступных желаемую дату и отмечает наличие у него багажа. Дело в том, что прямого рейса может не быть, тогда диспетчерская подбирает подходящий вариант самолетов с пересадками.

Купив билет пассажир приходит в аэропорт. За 60 минут до вылета (можете сделать, что 10 минут реального времени равно 1 мин или 1 сек времени программы) открывается регистрация на рейс у стоек регистрации. Пассажир подходит к свободной стойке и взвешивает багаж. Если после очередного пассажира допустимый вес самолета превышен, то больше пассажиры на этот рейс не принимаются, им приходится ждать следующий (вот такой хороший аэропорт).

Постарайтесь продавать билеты таким образом, чтобы не обижать пассажиров слишком сильно. Иначе они могут и в суд подать!

**Цифровой магазин**

Вам в голову пришла гениальная идея, открыть цифровой магазин игр. У игры есть цена, название, дата выхода, разработчик, описание, метки жанров, минимальные технические требования, а также перечень поддерживаемых языков. Чтобы привлечь побольше покупателей, вы решили ввести систему скидок. Как администратор магазина вы каждую неделю составляете список игр, которые будут продаваться со скидкой, и размер скидки для каждой игры (не будете же вы игру, которая и сама хорошо продается выставлять с огромной скидкой).

Покупатель может сортировать игры как по названию, разработчику, одной или нескольким меткам, цене и размеру скидки.

Когда выходит новая игра, разработчик предлагает вам ее выставить у себя на продажу. За право продавать эту игру с каждой продажи вы должны отчислять разработчику фиксированную сумму. Соответственно, если игра продается плохо, вы можете, конечно, продавать игру со скидкой, но тогда ваша выгода будет гораздо меньше.

Постарайтесь не стать банкротом. Для того, чтобы своевременно принимать меры по ценовой политике, вы всегда должны быть в курсе, сколько денег есть на счете вашего магазина, чтобы не уходить в минус.

**Game dev**

Реализуйте любую консольную игру с использованием ООП. Например, Пакман. У него есть 4 типа объектов: сам пакман, призрак, точка и бонус. У призраков, например, есть следующие характеристики: цвет, имя, скорость, направление движения. В пакмане, если хочется совсем уж усложнить себе жизнь, то можно сделать случайную генерацию карты.

Саму игру можете взять любую. Но помните про принципы ООП.