УВО бан «Университет бан Управления бан «ТИСБИ»

Факультет бан информационных бан технологий

Кафедра бан информационных бан технологий

**Курсовая бан работа бан**

по бан дисциплине бан «Объектно-ориентированное бан программирование»

на бан тему: бан «Разработка бан объектной бан программы бан для бан задачи бан «Расписание бан киносеансов»

Выполнил: бан студент бан гр. бан П-011 бан Сучёв бан Н.Е. бан

Проверил: бан ст. бан преп. бан Якунина бан Е.А.

Казань бан 2022

Оглавление

[Постановка бан задачи 3](#_Toc105284107)

[1. Описание бан используемых бан структур бан данных бан с бан алгоритмами бан выполнения бан основных бан операций 5](#_Toc105284108)

[1.1 бан Кольцевая бан очередь. 5](#_Toc105284109)

[1.2 бан Адресный бан стек 6](#_Toc105284110)

[2. Краткие бан сведения бан об бан объектном бан подходе 8](#_Toc105284111)

[3. Формализованное бан описание бан разработанных бан классов 14](#_Toc105284112)

[4. бан Описание бан демонстрационного бан модуля бан с бан характеристикой бан использованных бан стандартных бан компонентов бан и бан списком бан реализованных бан обработчиков 18](#_Toc105284113)

[5. Описание бан структуры бан проекта бан в бан соответствии бан с бан использованным бан инструментом бан разработки 23](#_Toc105284114)

[Список бан литературы 25](#_Toc105284115)

[Листинг бан программы 27](#_Toc105284116)

Постановка бан задачи

**Цели бан работы: бан** отработка бан навыков бан курсов бан «объектно-ориентированное бан программирование» бан и бан «структуры бан и бан алгоритмы бан обработки бан данных».

**Постановка бан задачи: бан** разработать бан объектную бан программу бан кинотеатр бан для бан составления бан расписания бан киносеансов бан организует бан список бан демонстрируемых бан кинофильмов бан с бан указанием бан названия бан и бан продолжительности бан показа бан (в бан минутах). бан Для бан каждого бан кинофильма бан создаётся бан список бан с бан указанием бан даты бан и бан времени бан показа.

Разработка бан программы бан включает бан в бан себя:

* определение бан необходимых бан объектов бан и бан способов бан их бан взаимодействия;
* формальное бан описание бан объектов бан в бан виде бан классов;
* программную бан реализацию бан всех бан необходимых бан методов, бан включая бан корректировку бан вкладов бан и бан подсчет бан текущего бан суммарного бан объема бан всех бан вкладов;
* всестороннее бан тестирование бан методов бан с бан помощью бан консольного бан (при бан разработке) бан и бан оконного бан (в бан окончательном бан варианте) бан приложения.

Для бана объединения бан кинофильмов бан используется бан структура бан данных бан в бан виде бан кольцевой бан очередь бан на бан основе бан обычного бан массива. бан Для бан объединения бан сеансов бан каждого бан кинофильма бан используется бан структура бан данных бан в бан виде бан адресного бан стека.

Разработка бан выполняется бан с бан учетом бан следующих бан требований:

1. имена бан классов, бан свойств бан и бан методов бан должны бан носить бан содержательный бан смысл бан и бан соответствовать бан информационной бан задаче
2. обязательное бан соблюдение бан принципа бан инкапсуляции бан – бан использование бан в бан классах бан только бан закрытых бан свойств бан и бан реализация бан необходимого бан набора бан методов бан доступа
3. наличие бан двух бан методов бан для бан сохранения бан всей бан объектной бан структуры бан во бан внешнем бан файле бан с бан обратной бан загрузкой, бан при бан этом бан стандартные бан механизмы бан сериализации бан разрешается бан использовать бан только бан как бан дополнение бан к бан самостоятельно бан реализованным бан методам
4. тестовое бан оконное бан приложение бан должно бан обладать бан удобным бан пользовательским бан интерфейсом бан с бан контролем бан вводимых бан данных бан и бан отображением бан текущего бан состояния бан объектной бан структуры бан с бан помощью бан списковых бан или бан табличных бан компонентов
5. стандартные бан контейнеры/коллекции бан (включая бан обобщенные бан классы) бан разрешается бан использовать бан только бан как бан дополнение бан к бану самостоятельно бан разработанным бан классам
6. в бан качестве бан языка бан разработки бан разрешается бан использовать бан Java, бан С, бан C++, бан Object/Free бан Pascal бан и бан соответствующие бан инструменты бан быстрой бан разработки бан приложений. бан
7. **Описание бан используемых бан структур бан данных бан с бан алгоритмами бан выполнения бан основных бан операций бан**

1.1 бан Кольцевая бан очередь.

Кольцевая бан очередь бан работает бан по бан принципу бан FIFO бан (First бан – бан In бан – бан First бан – бан Out): бан «первым бан пришел бан - бан первым бан вышел». бан Отличие бан между бан кольцевой бан очередью бан и бан обычной бан очередью бан заключается бан в бан способе бан выхода бан из бан очереди бан первого бан элемента. бан В бан кольцевой бан очереди бан первый бан элемент бан перемещается бан в бан конец бан очереди. бан Примерами бан кольцевой бан очереди бан могут бан быть бан движение бан троллейбусов бан по бан кольцевому бан маршруту, бан круговорот бан воды бан в бан природе бан и бан т.д.

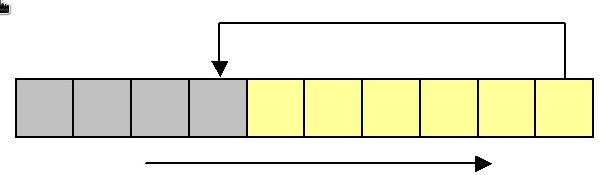


Рис. бан 5: бан Кольцевая бан очередь. бан Перемещение бан первого бан элемента бан в бан конец бан очереди

Кольцевая бан очередь бан работает бан по бан следующему бан принципу.

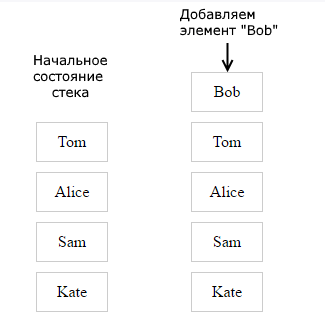
1. Сначала бан в бан очередь бан добавляются бан элементы бан до бан тех бан пор, бан пока бан не бан будет бан достигнут бан какой-то бан максимум. бан Элементы бан добавляются бан в бан конец бан очереди.
2. После бан достижения бан определенного бан (максимального) бан количества бан элементов бан происходит бан циклическое бан изменение бан первого бан элемента бан по бан следующему бан принципу: бан первый бан элемент бан очереди бан перемещается бан в бан конец бан очереди, бан а бан остальные бан элементы бан этой бан очереди бан сдвигаются бан на бан одну бан позицию бан впе бан ред. бан Таким бан образом бан второй бан элемент бан очереди бан становится бан первым, бан третий бан элемент бан становится бан вторым бан и бан т.д.

Базовый бан перечень бан выполняемых бан операций бан для бан кольцевой бан очереди бан следую бан щий:

* добавить бан новый бан элемент бан в бан очередь;
* проверка, бан есть бан ли бан очередь бан пустой;
* проверка, бан есть бан ли бан очередь бан полной;
* очистка бан очереди бан (удаление бан всех бан элементов бан очереди);
* вытянуть бан первый бан элемент бан из бан очереди бан и бан поместить бан в бан конец бан очереди. бан

1.2 бан Адресный бан стек

Стек бан представляет бан собой бан структуру бан данных, бан которая бан работает бан по бан принципу бан LIFO бан (Last бан In бан First бан Out бан - бан "последний бан пришел бан - бан первый бан вышел"). бан Графически бан стек бан можно бан представить бан в бан виде бан столбика бан или бан стопки бан объектов:

* 
* Рис. бан 1 бан Стек

Стек бан имеет бан вершину, бан который бан образует бан последний бан добавленный бан элемент. бан При бан добавлении бан новый бан элемент бан помещается бан поверх бан вершины бан стека бан и бан образует бан новую бан вершину. бан При бан удалении бан удаляется бан элемент бан из бан вершины бан стека, бан а бан предыдущий бан элемент бан образует бан новую бан вершину.

Так, бан на бан приведенном бан рисунке бан вначале бан вершиной бан стека бан является бан "Tom". бан После бан добавления бан нового бан элемента бан "Bob" бан этот бан элемент бан располагается бан поверх бан элемента бан "Tom" бан и бан становится бан новой бан вершиной.

В бан библиотеке бан классов бан .NET бан в бан принципе бан уже бан есть бан свой бан класс, бан который бан выполняет бан роль бан стека. бан Это бан класс бан - бан System.Collections.Generic.Stack. бан Но бан рассмотрим, бан как бан мы бан сами бан можем бан реализовать бан структуру бан в бан виде бан стека.

Структура бан стек бан вне бан зависимости бан от бан языка бан программирования бан обладает бан неким бан общим бан функционалом, бан который бан составляют бан метод бан добавления бан элемента бан (как бан правило, бан называется бан push()) бан и бан метод бан извлечения бан элемента бан из бан вершины бан стека бан (обычно бан называется бан pop()). бан Кроме бан того, бан нередко бан реализации бан стеков бан содержат бан метод бан получения бан элемента бан из бан вершины бан без бан его бан извлечения, бан метод бан определения бан размера бан стека бан и бан ряд бан других.

1. Краткие бан сведения бан об бан объектном бан подходе

Объектно-ориентированное бан программирование бан – бан одно бан из бан важнейших бан методологий бан разработки, бан которая бан основано бан на бан представлении бан программы бан в бан виде бан совокупности бан объектов, бан каждый бан из бана которых бан является бан экземпляром бан определённого бан класса, бан а бан классы бан образуют бан иерархию бан наследования.

Основные бан принципы бан ООП:

Инкапсуляция

Одним бан из бан самых бан важных бан факторов бан при бан проектировании бан компонентов бан приложения бан является бан сокрытие бан внутренних бан данных бан компонента бан и бан деталей бан его бан реализации бан от бан других бан компонентов бан приложения бан и бан предоставление бан набора бан методов бан для бан взаимодействия бан с бан ним бан (API). бан Этот бан принцип бан является бан одним бан из бан четырёх бан фундаментальных бан принципов бан ООП бан и бан называется бан инкапсуляцией.

Правильная бан инкапсуляция бан важна бан по бан многим бан причинам: бан

1. бан Она бан способствует бан переиспользованию бан компонентов: бан поскольку бан в бан этом бан случае бан компоненты бан взаимодействуют бан друг бан с бан другом бан только бан посредством бан их бан API бан и бан безразличны бан к бан изменениям бан внутренней бан структуры, бан они бан могут бан использоваться бан в бан более бан широком бан контексте.

2. бан Инкапсуляция бан ускоряет бан процесс бан разработки: бан слабо бан связанные бан друг бан с бан другом бан компоненты бан (то бан есть бан компоненты, бан чей бан код бан как бан можно бан меньше бан обращается бан или бан использует бан код бан других бан компонентов) бан могут бан разрабатываться, бан тестироваться бан и бан дополняться бан независимо.

3. бан правильно бан инкапсулированные бан компоненты бан более бан легки бан для бана понимания бан и бан процесса бан отладки, бан что бан упрощает бан поддержку бан приложения.

В бан языке бан Java бан инкапсуляция бан реализована бан с бан помощью бан системы бан классов, бан которые бан позволяют бан собрать бан информацию бан об бан объекте бан в бан одном бан месте; бан пакетов, бан которые бан группируют бан классы бан по бан какому-либо бан критерию, бан и бан модификаторов бан доступа, бан которыми бан можно бан пометить бан весь бан класс бан или бан его бан поле бан или бан метод.

Всего бан модификаторов бан доступа бан четыре:

ба public бан – бан полный бан доступ бан к бан сущности бан (полю бан или бан методу бан класса) бан из бан любого бан пакета;

protected бан – бан доступ бан к бану сущности бан только бан для бан классов бан своего бан пакета бан и бан наследников бан класса;

неявный бан модификатор бан по бан умолчанию бан (при бан отсутствии бан трёх бан явных) бан – бан доступ бан к бан сущности бан только бан для бан классов бан своего бан пакета;

private бан – бан доступ бан только бан внутри бан класса, бан в бан котором бан объявлена бан сущность.

Для бан достижения бан правильной бан инкапсуляции бан также бан необходимо бан предоставить бан корректный бан API бан для бан работы бан с бан компонентом. бан Например, бан в бан сеттер бан для бан переменной бан можно бан включить бан логику бан по бан проверке бан передаваемых бан значений бан либо бан не бан предоставлять бан сеттеры бан в бан классе бан вовсе, бан если бан класс бан должен бан быть бан доступен бан только бан для бан чтения.

Пример бан корректной бан инкапсуляции бан класса:

В бан примере бан выше бан значение бан переменной бан name бан задаётся бан при бан создании бан объекта бан и бан не бан может бан быть бан изменено бан извне, бан так бан как бан сеттер бан для бан переменной бан отсутствует. бан В бан сеттере бан для бан переменной бан age бан реализована бан проверка бан на бан корректность бан передаваемого бан параметра бан и бан выброс бан исключения бан при бан неверном бан значении.

Наследование

Наследование бан является бан одним бан из бан важнейших бан принципов бан объектно-ориентированного бан программирования, бан поскольку бан оно бан позволяет бан создавать бан иерархические бан структуры бан объектов. бан Используя бан наследование, бан можно бан создать бан общий бан класс, бан который бан будет бан определять бан характеристики бан и бан поведение, бан свойственные бан какому-то бан набору бан связанных бан объектов. бан В бан дальнейшем бан этот бан класс бан может бан быть бан унаследован бан другими, бан более бан частными бан классами, бан каждый бан из бан которых бан будет бан добавлять бан уникальные, бан свойственные бан только бан ему бан характеристики бан и бан дополнять бан или бан изменять бан поведение бан базового бан класса. бан В бан терминах бан Java бан такой бан общий бан класс бан называется бан суперклассом бан (superclass), бан или бан базовым бан классом бан (base бан class), бан или бан классом-родителем бан (parent бан class), бан а бан класс, бан его бан наследующий, бан - бан подклассом бан (subclass), бан или бан дочерним бан классом бан (child бан class), бан или бан классом-потомком бан (derived бан class).

Наследование бан реализует бан отношение бан «является» бан (“is-a”) бан между бан суперклассом бан и бан подклассом. бан Пусть, бан например, бан классы бан Employee бан и бан Manager бан представляют бан собой бан абстракцию бан понятий бан «Сотрудник» бан и бан «Менеджер». бан Каждый бан менеджер бан также бан является бан сотрудником бан компании, бан в бан которой бан он бан работает, бан следовательно, бан класс бан Manager бан находится бан в бан отношении бан “is-a” бан с бан классом бан Employee. бан Таким бан образом, бан с бан точки бан зрения бан наследования, бан при бан выстраивании бан иерархии бан классов бан класс бан Employee бан будет бан являться бан суперклассом, бан а бан класс бан Manager бан – бан дочерним бан классом. бан При бан этом бан класс, бан который бан является бан наследником бан какого-либо бан класса, бан может бан быть бан суперклассом бан для бан одного бан или бан нескольких бан других бан классов. бан Также бан в бан отличие, бан например, бан от бан C++, бан в бан Java бан в бан строгом бан смысле бан отсутствует бан множественное бан наследование, бан то бан есть бан любой бан класс бан может бан иметь бан не бан более бан одного бан класса-родителя. бан А бан все бан классы, бан суперкласс бан у бан которых бан явно бан не бан указан, бан по бан умолчанию бан наследуются бан от бан класса бан Object.

Класс бан Employee бан в бан примере бан выше бан является бан суперклассом бан не бан потому, бан что бан он бан главнее бан класса бан Manager бан или бан содержит бан больше бан функциональности. бан На бан самом бан деле бан верно бан обратное: бан функциональность бан подклассов бан не бан ýже, бан а бан зачастую бан существенно бан шире, бан чем бан функциональность бан их бан классов-родителей. бан Приставки бан «супер-» бан и бан «под-» бан пришли бан в бан Java бан из бан математики: бан множество бан всех бан менеджеров бан содержится бан во бан множестве бан всех бан сотрудников бан и, бан таким бан образом, бан является бан подмножеством бан множества бан сотрудников.

Для бан того бан чтобы бан унаследовать бан какой-либо бан класс бан в бан Java, бан используется бан ключевое бан слово бан extends:

В бан примере бан выше бан класс бан Employee бан является бан базовым бан классом бан для бан класса бан Manager, бан а бан класс бан Manager бан – бан подклассом бан класса бан Employee. бан Класс бан Employee бан абстрагирует бан базовые бан характеристики бан для бан всех бан сотрудников бан компании бан – бан имя, бан фамилию, бан размер бан оклада бан и бан дату бан приёма бан на бан работу, бан а бан класс бан Manager бан дополняет бан эти бан характеристики бан процентом бан премии бан для бан менеджеров бан и бан меняет бан поведение бан метода бан getSalary() бан базового бан класса, бан используя бан полиморфизм.

Полиморфизм

При бан рассмотрении бан полиморфизма бан необходимо бан помнить, бан что бан этот бан принцип бан неразрывно бан связан бан с бан другим бан принципом бан ООП бан – бан наследованием, бан которое бан помогает бан реализовать бан полиморфизм.

Возьмем бан для бана примера бан абстрактный бан класс бан «Автомобиль», бан который бан наследуют бан два бан конкретных бан класса бан – бан «Спортивный бан автомобиль» бан и бан «Грузовой бан автомобиль».

И бан спортивные, бан и бан грузовые бан автомобили бан будут бан обладать бан общими бан характеристиками бан и бан будут бан иметь бан возможность бан выполнять бан общие бан для бан всех бан автомобилей бан действия, бан которые бан указаны бан в бан абстрактном бан классе-родителе, бан но бан конкретная бан реализация бан этих бан действий бан может бан быть бан разной.

Например, бан общее бан для бан всех бан автомобилей бан действие бан «завестись» бан у бан спортивного бан автомобиля бан может бан быть бан реализовано бан путем бан нажатия бан кнопки, бан а бан у бан грузового бан - бан с бан помощью бан ключа. бан Один бан результат бан – бан разные бан решения. бан В бан этом бан и бан состоит бан полиморфизм.

Более бан строго, бан полиморфизм бан - бан один бан из бана принципов бан ООП, бан позволяющий бан вызовом бан переопределённого бан метода бан через бан переменную бан класса-родителя бан получить бан поведение, бан которое бан будет бан соответствовать бан реальному бан классу-потомку, бан на бан, который бан ссылается бан эта бан переменная.

Код бан выше бан представляет бан пример бан полиморфизма. бан Сначала бан переменной бан родительского бан класса бан Vehicle бан присваивается бан объект бан дочернего бан класса бан SportCar. бан При бан вызове бан метода бан start() бан на бан консоль бан будет бан выведено: бан "Starting бан my бан fancy бан sport бан car!"

При бан дальнейшем бан присвоении бан этой бан же бан переменной бан объекта бан дочернего бан класса бан Truck бан и бан вызове бан того бан же бан метода бан start() бан на бан консоль бан будет бан выведено: бан "Starting бан my бан heavy бан truck!"

Абстракция

Относительно бан недавно бан в бан качестве бан самостоятельного бан четвёртого бан принципа бан начали бан выделять бан абстракцию.

Одно бан из бана определений бан слова бан «абстракция», бан которые бан можно бан встретить бан в бан современных бан словарях:

Абстракция бан (от бан лат. бан abstractio бан — бан выделение, бан отвлечение бан или бан отделение) бан — бан теоретический бан прием бан исследования, бан позволяющий бан отвлечься бан от бан некоторых бан несущественных бан в бан определенном бан отношении бан свойств бан изучаемых бан явлений бан и бан выделить бан свойства бан существенные бан и бан определяющие.

Все бан языки бан программирования бан предоставляют бан их бан пользователю бан определённые бан абстракции. бан Так, бан языки бан семейства бан ассемблер бан являются бан в бан своём бан роде бан абстракцией бан соответствующих бан микропроцессоров, бан поскольку бан позволяют бан отвлечься бан от бан деталей бан их бан реализации бан и бан общаться бан с бан ними бан через бан определённый бан набор бан более бан высокоуровневых бан инструкций. бан Императивные бан языки бан программирования, бан последовавшие бан за бан ассемблером, бан например бан Basic, бан Fortran, бан C, бан являются бан более бан высоким бан уровнем бан абстракции бан над бан ассемблерными бан языками бан – бан они бан дают бан возможность бан использовать бан более бан привычные бан человеку бан синтаксические бан конструкции бан за бан счёт бан приближения бан синтаксиса бан к бан естественным бан языкам.

Объектно-ориентированные бан языки, бан такие бан как бан Java, бан выводят бан разработку бан на бан ещё бан более бан высокий бан уровень бан абстракции: бан объекты бан в бан ООП бан по бан своей бан сути бан представляют бан собой бан модели бан понятий бан окружающего бан мира, бан таких бан как бан Работник, бан Сервер, бан Запись бан в бан дневнике, бан и бан выделяют бан только бан те бан свойства бан этого бан понятия, бан которые бан необходимы бан в бан конкретном бан случае бан для бан решения бан конкретной бан проблемы.

Например, бан класс бан Student бан в бан приложении бан учёта бан студентов бан университета, бан кроме бан общих бан полей, бан таких бан как бан имя, бан фамилия, бан дата бан рождения бан и бан т.д., бан будет бан содержать бан поля, бан отражающие бан информацию бан о бан номере бан зачётной бан книжки, бан статусе бан студента бан (действующий, бан академический бан отпуск, бан отчислен), бан факультете, бан номере бан его бан группы, бан оценках бан за бан семестры бан и бан т.д. бан Но бан для бан того бан же бан класса бан Student бан в бан приложении бан учёта бан студентов бан в бан тренинг-центре бан EPAM бан такая бан информация бан будет бан неактуальна: бан класс бан будет бан содержать бан поля, бан отражающие бан учебный бан проект, бан на бан который бан был бан распределён бан студент, бан уровень бан его бан английского бан языка бан по бан результатам бан последнего бан тестирования, бан количество бан посещаемых бан мероприятий бан и бан т.д.

В бан этом бан и бан состоит бан абстракция: бан фокусировка бан разработчика бан на бан конкретных бан свойствах бан объекта бан зависит бан от бан тех бан задач, бан которые бан призван бан решать бан объект. бан Следствием бан такого бан подхода бан является бан то, бан что, бан если бан в бан императивных бан языках бан разработчику бан необходимо бан думать бан в бан терминах бан компьютерной бан логики, бан в бан объектно-ориентированных бан языках бан разработчик бан думает бан в бан терминах бан проблемной бан области, бан в бан которой бан он бан разрабатывает бан приложения.

3. Формализованное бан описание бан разработанных бан классов

У бан каждого бан класса бан есть бан сеанс бан поэтому бан первым бан делом бан реализуем бан класс бан сеанс бан с бан датой бан сеанса бан и бан продолжительностью бан показа бан фильма.

**using** бан System**;**

**using** бан System**.**Collections**.**Generic**;**

**using** бан System**.**Text**;**

**namespace** бан Third

**{**

class бан Seans

**{**

**private** бан string бан date**;**//дата бан показа бан фильма

**private** бан int бан time**;**//время бан показа бан фильма

**public** бан Seans**(**string бан \_date**,** бан int бан \_time**)**

**{**

**this.**date бан **=** бан \_date**;**

**this.**time бан **=** бан \_time**;**

**}**

**public** бан string бан getDate**()**

**{**

**return** бан date**;**

**}**

**public** бан void бан setDate**(**string бан \_date**)**

**{**

**this.**date бан **=** бан \_date**;**

**}**

**public** бан int бан getTime**()**

**{**

**return** бан time**;**

**}**

**public** бан void бан setTime**(**int бан \_time**)**

**{**

**this.**time бан **=** бан \_time**;**

**}**

**}**

**}**

Класс бан фильмы бан содержит бан в бан себе: бан сеансы, бан название бан фильма. Внутри бан класса бан идёт бан работа бан с бан такой бан структурой бан данных бан как бан адресный бан стек. бан

**using** бан System**;**

**using** бан System**.**Collections**.**Generic**;**

**using** бан System**.**Text**;**

**namespace** бан Third

**{**

class бан Film

**{**

**private** бан Stack бан firstSeans**;**//первый бан сеанс

**private** бан string бан nameFilm**;**//название бан фильма

**private** бан int бан durationFilm**;**//длина бан фильма

**private** бан Stack бан SP**;**

**private** бан int бан count**;**

**public** бан Film**(**string бан \_nameFilm**,** бан int бан \_durationFilm**)**

**{**

**this.**nameFilm бан **=** бан \_nameFilm**;**

**this.**durationFilm бан **=** бан \_durationFilm**;**

SP бан **=** бан **null;**

**}**

**public** бан int бан Count бан **{** бан **get** бан **{** бан **return** бан count**;** бан **}** бан **}**

**public** бан string бан Name бан **{** бан **get** бан **{** бан **return** бан nameFilm**;** бан **}** бан **set** бан **{** бан **this.**nameFilm бан **=** бан **value;** бан **}** бан **}**

**public** бан int бан Durat бан **{** бан **get** бан **{** бан **return** бан durationFilm**;** бан **}** бан **set** бан **{** бан **this.**durationFilm бан **=** бан **value;** бан **}** бан **}**

**public** бан Stack бан getSPEl бан **{** бан **get{** бан **return** бан SP**;** бан **}** бан **}**

**public** бан void бан addSeans**(**string бан \_date**,** бан int бан \_time**)**

**{**

Seans бан newSea бан **=** бан **new** бан Seans**(**\_date**,**\_time**);**

Stack бан pTemp бан **=** бан **new** бан Stack**(**newSea**,**SP**);**

**if** бан **(**SP бан **==** бан **null)** бан **{** бан firstSeans бан **=** бан pTemp**;}**

SP бан **=** бан pTemp**;**

count**++;**

**}**

**public** бан string бан getNameFilm**()** бан **{** бан **return** бан nameFilm**;** бан **}**

**public** бан bool бан delSeans**()**

**{**

**if** бан **(**SP бан **!=** бан **null)**

**{**

SP бан **=** бан SP**.**getPred**();**

count**--;**

**return** бан **true;**

**}**

**else** бан **return** бан **false;**

**}**

**public** бан int бан getSumStack**()**

**{**

**return** бан durationFilm**\***count**;**

**}**

**public** бан String бан getInfoStack**()**

**{**

String бан info бан **=** бан Convert**.**ToString**(**Count**);**

Stack бан pTemp бан **=** бан SP**;**

**while** бан **(**pTemp бан **!=** бан **null)**

**{**

info бан **+=** бан "\n" бан **+** бан pTemp**.**\_aDate**.**getDate**();**

info бан **+=** бан "\n" бан **+** бан pTemp**.**\_aDate**.**getTime**();**

pTemp бан **=** бан pTemp**.**getPred**();**

**}**

**return** бан info**;**

**}**

**public** бан Seans бан FindSeans**(**string бан \_nameSeans**)**

**{**

Stack бан vspom бан = бан SP;

while(vspom бан != бан null)

{

if(vspom.\_aDate.getDate().Equals(\_nameSeans))

{

бан бан бан бан return бан vspom.\_aDate;

}

else бан { бан vspom бан = бан vspom.getPred(); бан }

}

return бан null;

}

}

}

Класс бан стек бан реализует бан схему бан работы бан со бан стеком бан который бан будет бан хранить бан сеансы. бан

**using** бан System**;**

**using** бан System**.**Collections**.**Generic**;**

**using** бан System**.**Text**;**

**namespace** бан Third

**{**

class бан Stack

**{**

**public** бан Seans бан \_aDate**;**

**private** бан Stack бан pred**;**

**public** бан Stack**(**Seans бан \_date**,** бан Stack бан \_SP**)**

**{**

\_aDate бан **=** бан \_date**;**

pred бан **=** бан \_SP**;**

**}**

**public** бан Stack бан getPred**()**

**{**

**return** бан pred**;**

**}**

**}**

**}**

Класс бан кинотеатр бан реализует бан контейнер бан для бан хранения бан фильмов. бан Методы бан класса бан предоставляют бан доступ бан к бан методам бан других бан классов. бан

**using** бан System**;**

**using** бан System**.**Collections**.**Generic**;**

**using** бан System**.**Text**;**

**namespace** бан Third

**{**

class бан Cinema

**{**

**private** бан Film**[]** бан Queue**;**//реализация бан очереди бан на бан основе бан массива

**private** бан static бан int бан size бан **=** бан 10**;**

**private** бан int бан countFilm бан **=** бан 0**;**

**private** бан int бан first**;**//ссылка бан на бан первый бан элемент бан в бан очереди

**private** бан int бан last**;**//ссылка бан на бан последний бан элемент бан в бан очереди

**public** бан Cinema**()**

**{**

first бан **=** бан 0**;**

last бан **=** бан 0**;**

Queue бан **=** бан **new** бан Film**[**size**];**

**}**

**public** бан string бан addFilm**(**string бан \_nameFilm**,** бан int бан \_durationFilm**)**

**{**

**if** бан **(**size бан **>=** бан countFilm**)**

**{**

**if** бан **(**last бан **>** бан size бан **&&** бан first бан **>** бан 0**)** бан **{**last бан **=** бан 0**;}**

Film бан newFilm бан **=** бан **new** бан Film**(**\_nameFilm**,** бан \_durationFilm**);**

Queue**[**last**]** бан **=** бан newFilm**;**

last**++;**

countFilm**++;**

**return** бан \_nameFilm**;**

**}**

**return** бан ""**;**

**}**

**public** бан bool бан delFilm**()**

**{**

**if** бан **(** бан countFilm бан **!=** бан 0**)**

**{**

**if** бан **(**first бан **>** бан size**)** бан **{** бан first бан **=** бан 0**;}**

Queue**[**first**]** бан **=** бан **null;**

countFilm**--;**

first**++;**

**return** бан **true;**

**}**

**return** бан **false;**

**}**

**public** бан int бан COUNT бан **{** бан **get** бан **{** бан **return** бан countFilm**;** бан **}** бан **}**

**public** бан string бан getinfo**()**

**{**

string бан inform бан **=** бан ""**;**

**for(**int бан i бан **=** бан 0**;** бан i бан **<** бан countFilm**;** бан i**++)**

**{**

inform бан **+=** бан "\n" бан **+** бан Queue**[**i**].**getInfoStack**();**

**}**

**return** бан inform**;**

**}**

**public** бан string бан getinfoName**()**

**{**

string бан inform бан **=** бан Convert**.**ToString**(**COUNT**);**

**for** бан **(**int бан i бан **=** бан 0**;** бан i бан **<** бан countFilm**;** бан i**++)**

**{**

inform бан **+=** бан "\n" бан **+** бан Queue**[**i**].**Name**;**

inform бан **+=** бан "\n" бан **+** бан Queue**[**i**].**Durat**;**

**}**

**return** бан inform**;**

**}**

**public** бан Film бан getFirst**()** бан **{** бан **return** бан Queue**[**first**];** бан **}**

**public** бан Film бан FindFilm**(**string бан \_nameFilm**)**

**{**

Film бан pTemp**;**

**for(**int бан i бан **=** бан 0**;** бан i**<=** бан countFilm**;** бан i**++)**

**{**

pTemp бан **=** бан Queue**[**i**];**

**if** бан **(**pTemp бан **!=** бан **null** бан **&&** бан pTemp**.**getNameFilm**().**Equals**(**\_nameFilm**))**

**{**

бан бан бан бан **return** бан pTemp**;**

**}**

**}**

**return** бан **null;**

**}**

**public** бан bool бан getSeansToEdit**(**string бан \_date**,** бан int бан \_time**,** бан String бан \_nameFilm**)**

**{**

int бан i бан **=** бан 0**;**

Film бан pTemp бан **=** бан Queue**[**i**];**

**for** бан **(**int бан j бан **=** бан 0**;** бан j бан **<** бан size**;** бан j**++)**

**{**

**if** бан **(**pTemp бан **!=** бан **null** бан **&&** бан pTemp**.**getNameFilm**().**Equals**(**\_nameFilm**))**

**{**

бан бан бан бан pTemp**.**addSeans**(**\_date**,** бан \_time**);**

бан бан бан бан **return** бан **true;**

**}**

**else**

**{**

бан бан бан бан i**++;**

бан бан бан бан pTemp бан **=** бан Queue**[**i**];**

**}**

бан бан бан

**}**

**return** бан **false;**

**}**

**}**

**}**

4. бан Описание бан демонстрационного бан модуля бан с бан характеристикой бан использованных бан стандартных бан компонентов бан и бан списком бан реализованных бан обработчиков

При бан запуске бан программы бан создаётся бан экземпляр бан класса бан Кинотеатр бан с бан названием бан компании бан необходимое бан для бан создания бан файлов бан сохранения бан данных бан в бан файл. бан

**using** бан System**;**

**using** бан System**.**IO**;**

**using** бан System**.**Windows**.**Forms**;**

**namespace** бан Third

**{**

**public** бан **partial** бан class бан Form1 бан **:** бан Form

**{**

Cinema бан news бан **=** бан **new** бан Cinema**();**

**public** бан Form1**()**

**{**

InitializeComponent**();**

**}**

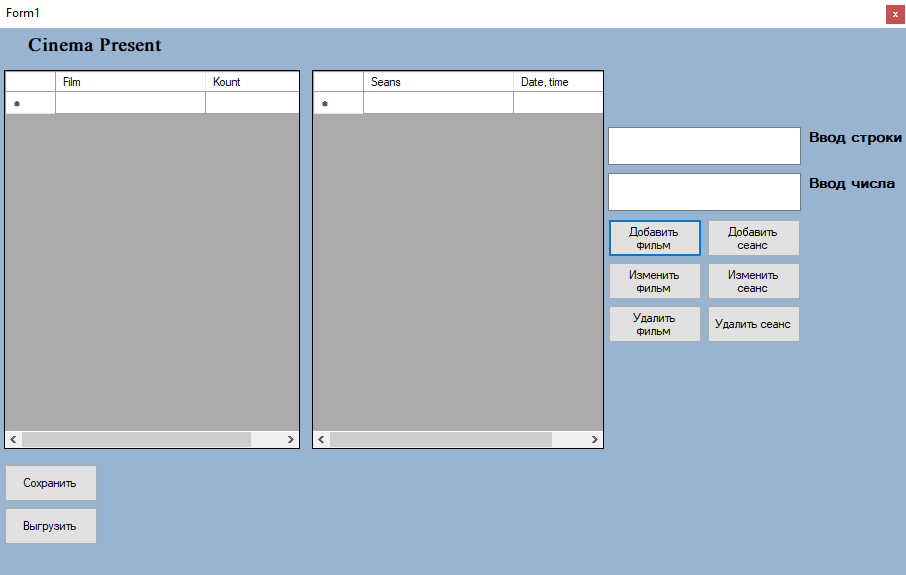


Рис. бан Запуск бан программы.

Обрабатываем бан удаление бан сеанса. бан Удаляем бан из бан стека бан данные бан и бан очищаем бан таблицу.

**private** бан void бан button4\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

string бан text бан **=** бан dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

Film бан vspom бан **=** бан news**.**FindFilm**(**text**);**

**if** бан **(**vspom бан **!=** бан **null)**

**{**

vspom**.**delSeans**();**

dataGridView2**.**Rows**.**RemoveAt**(**dataGridView2**.**CurrentRow**.**Index**);**

**}**

**else** бан **{** бан MessageBox**.**Show**(**"Выберите бан фильм, бан сеанс бан которого бан хотите бан удалить"**);** бан **}**

**}**

Кнопка бан добавления бан фильма. бан Читаем бан данные бан с бан TextBox бан и бан записываем бан данные бан в бан таблицу бан и бан структуру бан фильмов. бан

**private** бан void бан butAddFilm\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

news**.**addFilm**(**textBox**.**Text**,** бан Convert**.**ToInt32**(**textBox1**.**Text**));**

dataGridView1**.**Rows**.**Add**(**textBox**.**Text**,** бан textBox1**.**Text**);**

textBox**.**Clear**();**

textBox1**.**Clear**();**

**}**

**private** бан void бан listBox2\_SelectedIndexChanged**(object** бан sender**,** бан EventArgs бан e**){}**

Отрабатываем бан добавление бан сеанса. бан Обновляем бан таблицу бан сеансов бан и бан добавляем бан у бан структуру бан экземпляр бан от бан класса бан сеанс.

**private** бан void бан butAddSeans\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

string бан text бан **=** бан dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

news**.**getSeansToEdit**(**textBox**.**Text**,** бан Convert**.**ToInt32**(**textBox1**.**Text**),** бан text**);**

dataGridView2**.**Rows**.**Add**(**textBox**.**Text**,** бан textBox1**.**Text**);**

textBox**.**Clear**();**

textBox1**.**Clear**();**

**}**

Отрабатываем бан событие бан выбора бан фильма. бан При бан выборе бан фильма бан обновляем бан вторую бан таблицу бан с бан сеансами бан и бан отображаем бан соответствующие бан данные.

**private** бан void бан dataGridView2\_CellContentClick**(object** бан sender**,** бан DataGridViewCellEventArgs бан e**){}**

**private** бан void бан dataGridView1\_CellContentClick**(object** бан sender**,** бан DataGridViewCellEventArgs бан e**)**

**{**

dataGridView2**.**Rows**.**Clear**();**

string бан text бан **=** бан dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

Film бан vspom бан **=** бан news**.**FindFilm**(**text**);**

**if** бан **(**vspom бан **!=** бан **null)**

**{**

Stack бан pTemp бан **=** бан vspom**.**getSPEl**;**

int бан chet бан **=** бан news**.**FindFilm**(**text**).**Count**;**

**for** бан **(**int бан i бан **=** бан 0**;** бан i бан **<** бан chet**;** бан i**++)**

**{**

бан бан бан бан dataGridView2**.**Rows**.**Add**(**pTemp**.**\_aDate**.**getDate**(),** бан Convert**.**ToString**(**pTemp**.**\_aDate**.**getTime**()));**

бан бан бан бан pTemp бан **=** бан pTemp**.**getPred**();**

**}**

**}**

**}**

Отрабатываем бан нажатие бан кнопки бан изменения бан данных бан фильма. бан Читаем бан данные бан с бан TextBox бан и бан изменяем бан выбранную бан ячейку бан в бан таблице бан и бан изменяем бан данные бан с бан очереди бан фильмов.

**private** бан void бан butIzm\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

string бан text бан **=** бан dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

int бан index бан **=** бан dataGridView1**.**CurrentRow**.**Index**;**

news**.**FindFilm**(**text**).**Name бан **=** бан textBox**.**Text**;**

news**.**FindFilm**(**textBox**.**Text**).**Durat бан **=** бан Convert**.**ToInt32**(**textBox1**.**Text**);**

dataGridView1**.**Rows**[**index**].**Cells**[**0**].**Value бан **=** бан textBox**.**Text**;**

dataGridView1**.**Rows**[**index**].**Cells**[**1**].**Value бан **=** бан textBox1**.**Text**;**

**}**

Событие бан изменение бан данных бан сеанса.

**private** бан void бан button2\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

string бан text бан **=** бан dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

int бан index бан **=** бан dataGridView2**.**CurrentRow**.**Index**;**

news**.**FindFilm**(**text**).**FindSeans**(**dataGridView2**.**CurrentCell**.**Value**.**ToString**()).**setDate**(**textBox**.**Text**);**

news**.**FindFilm**(**text**).**FindSeans**(**textBox**.**Text**).**setTime**(**Convert**.**ToInt32**(**textBox1**.**Text**));**

dataGridView2**.**Rows**[**index**].**Cells**[**0**].**Value бан **=** бан textBox**.**Text**;**

dataGridView2**.**Rows**[**index**].**Cells**[**1**].**Value бан **=** бан textBox1**.**Text**;**

**}**

Удаление бан фильма бан из бан таблицы бан и бан из бан очереди.

**private** бан void бан button1\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

string бан text бан **=** бан dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

**if** бан **(**news**.**getFirst**().**Name**.**Equals**(**text**))**

**{**

dataGridView1**.**Rows**.**RemoveAt**(**dataGridView1**.**CurrentRow**.**Index**);**

news**.**delFilm**();**

**}**

**else** бан **{** бан MessageBox**.**Show**(**"Выберите бан первый бан в бан списке бан фильм бан для бан удаления"**);** бан **}**

**}**

Метод бан выгрузки бан данных бан в бан файл бан формата бан txt. бан Для бан дальнейшего бан чтения бан в бан начале бан файла бан пишем бан "ФАЙЛ", бан чтобы бан можно бан было бан идентифицировать бан файл бан программы бан от бан других бан txt.

**private** бан void бан butUpload\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

string бан path бан **=** бан ""**,** бан doIt бан **=** бан ""**;**

int бан Count**,** бан LCount**,** бан gName**;**

**using** бан **(**OpenFileDialog бан openFile бан **=** бан **new** бан OpenFileDialog**()** бан **{** бан Filter бан **=** бан "txt бан files бан (\*.txt)|\*.txt|All бан files бан (\*.\*)|\*.\*" бан **})**

**{**

**if** бан **(**openFile**.**ShowDialog**()** бан **==** бан DialogResult**.**OK**)**

**{**

бан бан бан бан path бан **=** бан openFile**.**FileName**;**

**}**

**}**

**using** бан **(**StreamReader бан stream3 бан **=** бан **new** бан StreamReader**(**path**))**

**{**

doIt бан **=** бан stream3**.**ReadLine**();**

//Проверка бан на бан пригодность бан файла бан для бан работы бан с бан программой

**if** бан **(**doIt бан **==** бан "Файл"**)**

**{**

бан бан бан бан **for** бан **(**int бан i бан **=** бан 0**;** бан i бан **<=** бан news**.**COUNT**;** бан i**++)**

бан бан бан бан **{**

бан бан бан бан news**.**delFilm**();**

бан бан бан бан **}**

бан бан бан бан dataGridView1**.**Rows**.**Clear**();**

бан бан бан бан dataGridView2**.**Rows**.**Clear**();**

бан бан бан бан Count бан **=** бан int**.**Parse**(**stream3**.**ReadLine**());**

бан бан бан бан **for** бан **(**int бан i бан **=** бан 1**;** бан i бан **<=** бан Count**;)**

бан бан бан бан **{**

бан бан бан бан string бан nameLH бан **=** бан stream3**.**ReadLine**();**

бан бан бан бан int бан seats бан **=** бан int**.**Parse**(**stream3**.**ReadLine**());**

бан бан бан бан news**.**addFilm**(**nameLH**,** бан seats**);**

бан бан бан бан dataGridView1**.**Rows**.**Add**();**

бан бан бан бан dataGridView1**.**Rows**[**i бан **-** бан 1**].**Cells**[**0**].**Value бан **=** бан nameLH**;**

бан бан бан бан dataGridView1**.**Rows**[**i бан **-** бан 1**].**Cells**[**1**].**Value бан **=** бан seats**;**

бан бан бан бан **if** бан **(**i бан **!=** бан Count**)**

бан бан бан бан i**++;**

бан бан бан бан **else**

бан бан бан бан **break;**

бан бан бан бан **}**

бан бан бан бан stream3**.**ReadLine**();**

бан бан бан бан int бан hall бан **=** бан 0**;**

бан бан бан бан **while** бан **(**stream3**.**Peek**()** бан **!=** бан **-**1**)**

бан бан бан бан **{**

бан бан бан бан LCount бан **=** бан int**.**Parse**(**stream3**.**ReadLine**());**

бан бан бан бан **if** бан **(**LCount бан **!=** бан 0**)**

бан бан бан бан **for** бан **(**int бан i бан **=** бан 1**;** бан i бан **<=** бан LCount**;)**

бан бан бан бан **{**

бан бан бан бан string бан inf бан **=** бан stream3**.**ReadLine**();**

бан бан бан бан gName бан **=** бан int**.**Parse**(**stream3**.**ReadLine**());**

бан бан бан бан string бан name бан **=** бан Convert**.**ToString**(**dataGridView1**.**Rows**[**hall**].**Cells**[**0**].**Value**);**

бан бан бан бан news**.**FindFilm**(**name**).**addSeans**(**inf**,**gName**);**

бан бан бан бан **if** бан **(**i бан **!=** бан LCount**)**

бан бан бан бан i**++;**

бан бан бан бан **else**

бан бан бан бан **break;**

бан бан бан бан **}**

бан бан бан бан hall**++;**

бан бан бан бан **}**

**}**

**else**

**{**

бан бан бан бан MessageBox**.**Show**(**"Данный бан файл бан не бан подходит бан для бан работы бан с бан программой"**);**

**}**

stream3**.**Close**();**

**}**

**}**

Метод бан загрузки бан из бан файла. бан Открывает бан диалоговое бан окно бан для бан отображения бан txt бан файлов. бан Если бан файл бан удалось бан открыть бан происходит бан проверка бан на бан корректные бан данные, бан т.е. бан первая бан сточка бан в бан файле бан должна бан быть бан "ФАЙЛ" бан после бан этого бан происходит бан дальнейшее бан чтение бан файла, бан заполнение бан структуры бан и бан таблицы бан данными бан из бан файла.

**private** бан void бан butSave\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

string бан path**;**

string бан fileText бан **=** бан "Файл\n"**;**

//Создание бан экземпляра бан диалогового бан окна бан (Для бан того бан чтобы бан не бан занимать бан поток) бан + бан фильтр бан на бан создание бан текстовых бан файлов

**using** бан **(**SaveFileDialog бан saveFile бан **=** бан **new** бан SaveFileDialog**()** бан **{** бан Filter бан **=** бан "txt бан files бан (\*.txt)|\*.txt|All бан files бан (\*.\*)|\*.\*" бан **})**

**{**

**if** бан **(**saveFile**.**ShowDialog**()** бан **==** бан DialogResult**.**OK**)**

**{**

бан бан бан бан path бан **=** бан saveFile**.**FileName**;**

бан бан бан бан fileText бан **+=** бан news**.**getinfoName**()** бан **+** бан "\n"**;**

бан бан бан бан **for(**int бан i бан **=** бан 0**;** бан i бан **<** бан news**.**COUNT**;** бан i**++)**

бан бан бан бан **{**

бан бан бан бан fileText бан **+=** бан news**.**getinfo**();**

бан бан бан бан **}**

бан бан бан бан //Та бан же бан работа бан с бан экземплярами бан файла бан и бан записи бан файла

бан бан бан бан **using** бан **(**FileStream бан file бан **=** бан **new** бан FileStream**(**path**,** бан FileMode**.**OpenOrCreate**))**

бан бан бан бан **using** бан **(**StreamWriter бан stream бан **=** бан **new** бан StreamWriter**(**file**))**

бан бан бан бан **{**

бан бан бан бан stream**.**Write**(**fileText**);**

бан бан бан бан stream**.**Close**();**

бан бан бан бан file**.**Close**();**

бан бан бан бан **}**

**}**

**}**

**}**

**}**

**}**

1. Описание бан структуры бан проекта бан в бан соответствии бан с бан использованным бан инструментом бан разработки

Microsoft бан Visual бан Studio бан — бан линейка бан продуктов бан компании бан Microsoft, бан включающих бан интегрированную бан среду бан разработки бан программного бан обеспечения бан и бан ряд бан других бан инструментов. бан Данные бан продукты бан позволяют бан разрабатывать бан как бан консольные бан приложения, бан так бан и бан игры бан и бан приложения бан с бан графическим бан интерфейсом, бан в бан том бан числе бан с бан поддержкой бан технологии бан Windows бан Forms, бан UWP бан а бан также бан веб-сайты, бан веб-приложения, бан веб-службы бан как бан в бан родном, бан так бан и бан в бан управляемом бан кодах бан для бан всех бан платформ, бан поддерживаемых бан Windows, бан Windows бан Mobile, бан Windows бан CE, бан .NET бан Framework, бан .NET бан Core, бан .NET, бан MAUI, бан Xbox, бан Windows бан Phone бан .NET бан Compact бан Framework бан и бан Silverlight. бан После бан покупки бан компании бан Xamarin бан корпорацией бан Microsoft бан появилась бан возможность бан разработки бан IOS бан и бан Android бан программ.

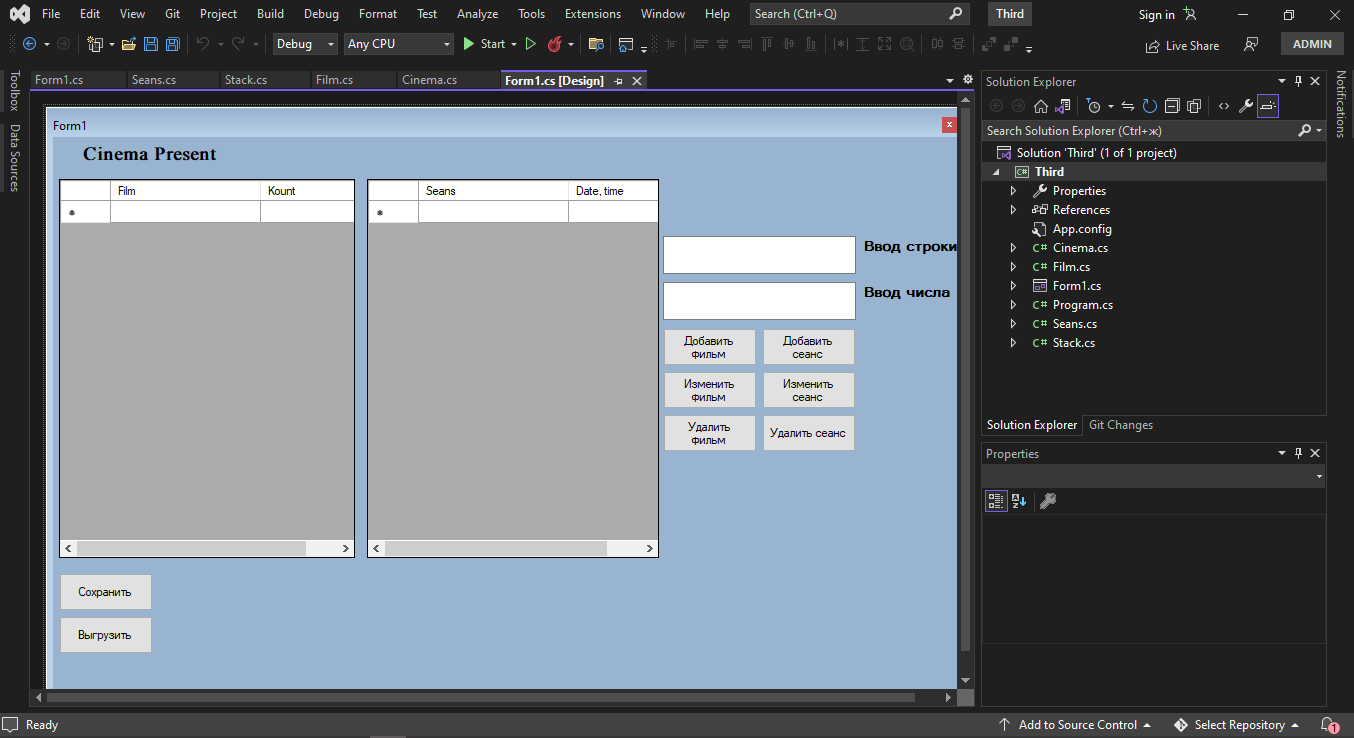


Рис бан Visual бан Studio

В бан главной бан папке бан проекта бан находятся бан разработанные бан классы бан \*.cs бан , бан так бан же бан в бан App.config бан находятся бан настройки бан для бан среды бан разработки бан Visual бан Studio бан находятся бан настройки бан конфигурации бан для бан запуска бан приложения. бан Файл бан \*.resx бан содержит бан в бан себе бан дизайн бан оконного бан приложения, бан составленный бан на бан языке бан разметки бан XML. бан

В бан папке бан Propertis бан находятся бан все бан подключенные бан в бан разработанных бан классах бан стандартные бан библиотеки бан для бан установления бан зависимостей бан в бан проекте. бан

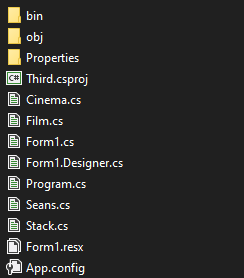


Рис. бан 10 бан Содержимое бан папки бан проекта. бан

Папка бан obj бан содержит бан в бан себе бан данные бан последнего бан Debug бан запуска бан проекта бан с бан соответствующими бан файлами бан конфигурации бан и бан логами. бан

В бан парке бан bin бан находиться бан главный бан исполняемый бан файл бан программы бан с бан последней бан сборки бан проекта, бан а бан также бан папка бан Relies бан с бан исполняемым бан файлом бан готовым бан к бан установке бан программы бан со бан всеми бан зависимостями бан на бан любой бан Windows бан операционной бан системе. бан

# Список бан литературы

1. Алгоритмы бан и бан структуры бан данных: бан учебное бан пособие бан / бан О.Б. бан Фофанов; бан Томский бан политехнический бан университет. бан – бан Томск: бан Изд-во бан Томского бан политех бан нического бан университета, бан 2019.
2. Васильев бан А. бан Н. бан Java. бан Объектно-ориентированное бан программирование. бан Учебное бан пособие. бан Стандарт бан третьего бан поколения. бан – бан "Издательский бан дом бан Питер бан 2021.
3. Кувшинов бан Д. бан Р., бан Осипов бан С. бан И. бан Основы бан программирования: бан язык бан C++: бан учебное бан пособие. бан – бан 2021.
4. Ляпин бан А. бан А., бан Быкова бан Ю. бан А. бан РАЗРАБОТКА бан БАЛЛИСТИЧЕСКОЙ бан ФУНКЦИИ бан НА бан ЯЗЫКЕ бан ПРОГРАММИРОВАНИЯ бан C/C++ бан //ПОЛИ- бан ХОТОМИЧЕСКИЕ бан КЛАССИФИКАЦИИ бан СПЕЦИАЛЬНЫХ бан ТЕРМИ- бан НОВ бан В бан НАУЧНОЙ бан И бан УЧЕБНОЙ бан ЛИТЕРАТУРЕ. бан – бан 2021. бан – бан С. бан 308-311.
5. Объектно-ориентированное бан программирование. бан Учебно-методическое бан пособие/ бан Козин.А.Н., бан Якунина бан Е.А. бан – бан Казань: бан УВО бан «Университет бан управления бан «ТИСБИ», бан 2020.
6. Структуры бан и бан алгоритмы бан обработки бан данных. бан Часть бан 2. бан Усложнённые бан структуры бан данных. бан Теория бан и бан методика бан обучения. бан Учебно-методическое бан пособие бан / бан А.Н. бан Козин, бан Л.Б. бан Таренко. бан - бан Казань: бан УВО бан "Университет бан управления бан "ТИСБИ бан 2018
7. Павловская бан Т. бан А., бан Щупак бан Ю. бан А. бан C/C++. бан Структурное бан и бан объектно- бан ориентированное бан программирование. бан Практикум. бан – бан "Издательский бан дом бан Питер бан 2021.
8. Структуры бан и бан алгоритмы бан обработки бан данных. бан Часть бан 3. бан Поиск бан и бан сортировка бан данных. бан Теория бан и бан методика бан обучения. бан Учебно-методическое бан пособие/ бан А.Н. бан Козин, бан Л.Б. бан Таренко. бан - бан Казань: бан УВО бан "Университет бан управления бан "ТИСБИ бан 2020
9. Puuronen бан M. бан Implementing бан Horizontal бан Layout бан for бан the бан Qt бан Design бан Studio’s бан Component бан Library. бан – бан 2021.
10. Sherriff бан N. бан Learn бан Qt бан 5: бан Build бан modern, бан responsive бан cross-platform бан desktop бан applications бан with бан Qt, бан C++, бан and бан QML. бан – бан Packt бан Publishing бан Ltd, бан 2018. бан

Листинг бан программы

**using** бан System**;**

**using** бан System**.**IO**;**

**using** бан System**.**Windows**.**Forms**;**

**namespace** бан Third

**{**

**public** бан **partial** бан class бан Form1 бан **:** бан Form

**{**

Cinema бан news бан **=** бан **new** бан Cinema**();**

**public** бан Form1**()**

**{**

InitializeComponent**();**

**}**

**private** бан void бан button4\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

string бан text бан **=** бан dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

Film бан vspom бан **=** бан news**.**FindFilm**(**text**);**

**if** бан **(**vspom бан **!=** бан **null)**

**{**

vspom**.**delSeans**();**

dataGridView2**.**Rows**.**RemoveAt**(**dataGridView2**.**CurrentRow**.**Index**);**

**}**

**else** бан **{** бан MessageBox**.**Show**(**"Выберите бан фильм, бан сеанс бан которого бан хотите бан удалить"**);** бан **}**

**}**

**private** бан void бан butAddFilm\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

news**.**addFilm**(**textBox**.**Text**,** бан Convert**.**ToInt32**(**textBox1**.**Text**));**

dataGridView1**.**Rows**.**Add**(**textBox**.**Text**,** бан textBox1**.**Text**);**

textBox**.**Clear**();**

textBox1**.**Clear**();**

**}**

**private** бан void бан listBox2\_SelectedIndexChanged**(object** бан sender**,** бан EventArgs бан e**){}**

**private** бан void бан butAddSeans\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

string бан text бан **=** бан dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

news**.**getSeansToEdit**(**textBox**.**Text**,** бан Convert**.**ToInt32**(**textBox1**.**Text**),** бан text**);**

dataGridView2**.**Rows**.**Add**(**textBox**.**Text**,** бан textBox1**.**Text**);**

textBox**.**Clear**();**

textBox1**.**Clear**();**

**}**

**private** бан void бан dataGridView2\_CellContentClick**(object** бан sender**,** бан DataGridViewCellEventArgs бан e**){}**

**private** бан void бан dataGridView1\_CellContentClick**(object** бан sender**,** бан DataGridViewCellEventArgs бан e**)**

**{**

dataGridView2**.**Rows**.**Clear**();**

string бан text бан **=** бан dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

Film бан vspom бан **=** бан news**.**FindFilm**(**text**);**

**if** бан **(**vspom бан **!=** бан **null)**

**{**

Stack бан pTemp бан **=** бан vspom**.**getSPEl**;**

int бан chet бан **=** бан news**.**FindFilm**(**text**).**Count**;**

**for** бан **(**int бан i бан **=** бан 0**;** бан i бан **<** бан chet**;** бан i**++)**

**{**

бан бан бан бан dataGridView2**.**Rows**.**Add**(**pTemp**.**\_aDate**.**getDate**(),** бан Convert**.**ToString**(**pTemp**.**\_aDate**.**getTime**()));**

бан бан бан бан pTemp бан **=** бан pTemp**.**getPred**();**

**}**

**}**

**}**

**private** бан void бан butIzm\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

string бан text бан **=** бан dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

int бан index бан **=** бан dataGridView1**.**CurrentRow**.**Index**;**

news**.**FindFilm**(**text**).**Name бан **=** бан textBox**.**Text**;**

news**.**FindFilm**(**textBox**.**Text**).**Durat бан **=** бан Convert**.**ToInt32**(**textBox1**.**Text**);**

dataGridView1**.**Rows**[**index**].**Cells**[**0**].**Value бан **=** бан textBox**.**Text**;**

dataGridView1**.**Rows**[**index**].**Cells**[**1**].**Value бан **=** бан textBox1**.**Text**;**

**}**

**private** бан void бан button2\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

string бан text бан **=** бан dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

int бан index бан **=** бан dataGridView2**.**CurrentRow**.**Index**;**

news**.**FindFilm**(**text**).**FindSeans**(**dataGridView2**.**CurrentCell**.**Value**.**ToString**()).**setDate**(**textBox**.**Text**);**

news**.**FindFilm**(**text**).**FindSeans**(**textBox**.**Text**).**setTime**(**Convert**.**ToInt32**(**textBox1**.**Text**));**

dataGridView2**.**Rows**[**index**].**Cells**[**0**].**Value бан **=** бан textBox**.**Text**;**

dataGridView2**.**Rows**[**index**].**Cells**[**1**].**Value бан **=** бан textBox1**.**Text**;**

**}**

**private** бан void бан button1\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

string бан text бан **=** бан dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

**if** бан **(**news**.**getFirst**().**Name**.**Equals**(**text**))**

**{**

dataGridView1**.**Rows**.**RemoveAt**(**dataGridView1**.**CurrentRow**.**Index**);**

news**.**delFilm**();**

**}**

**else** бан **{** бан MessageBox**.**Show**(**"Выберите бан первый бан в бан списке бан фильм бан для бан удаления"**);** бан **}**

**}**

**private** бан void бан butUpload\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

string бан path бан **=** бан ""**,** бан doIt бан **=** бан ""**;**

int бан Count**,** бан LCount**,** бан gName**;**

**using** бан **(**OpenFileDialog бан openFile бан **=** бан **new** бан OpenFileDialog**()** бан **{** бан Filter бан **=** бан "txt бан files бан (\*.txt)|\*.txt|All бан files бан (\*.\*)|\*.\*" бан **})**

**{**

**if** бан **(**openFile**.**ShowDialog**()** бан **==** бан DialogResult**.**OK**)**

**{**

бан бан бан бан path бан **=** бан openFile**.**FileName**;**

**}**

**}**

**using** бан **(**StreamReader бан stream3 бан **=** бан **new** бан StreamReader**(**path**))**

**{**

doIt бан **=** бан stream3**.**ReadLine**();**

//Проверка бан на бан пригодность бан файла бан для бан работы бан с бан программой

**if** бан **(**doIt бан **==** бан "Файл"**)**

**{**

бан бан бан бан **for** бан **(**int бан i бан **=** бан 0**;** бан i бан **<=** бан news**.**COUNT**;** бан i**++)**

бан бан бан бан **{**

бан бан бан бан news**.**delFilm**();**

бан бан бан бан **}**

бан бан бан бан dataGridView1**.**Rows**.**Clear**();**

бан бан бан бан dataGridView2**.**Rows**.**Clear**();**

бан бан бан бан Count бан **=** бан int**.**Parse**(**stream3**.**ReadLine**());**

бан бан бан бан **for** бан **(**int бан i бан **=** бан 1**;** бан i бан **<=** бан Count**;)**

бан бан бан бан **{**

бан бан бан бан string бан nameLH бан **=** бан stream3**.**ReadLine**();**

бан бан бан бан int бан seats бан **=** бан int**.**Parse**(**stream3**.**ReadLine**());**

бан бан бан бан news**.**addFilm**(**nameLH**,** бан seats**);**

бан бан бан бан dataGridView1**.**Rows**.**Add**();**

бан бан бан бан dataGridView1**.**Rows**[**i бан **-** бан 1**].**Cells**[**0**].**Value бан **=** бан nameLH**;**

бан бан бан бан dataGridView1**.**Rows**[**i бан **-** бан 1**].**Cells**[**1**].**Value бан **=** бан seats**;**

бан бан бан бан **if** бан **(**i бан **!=** бан Count**)**

бан бан бан бан i**++;**

бан бан бан бан **else**

бан бан бан бан **break;**

бан бан бан бан **}**

бан бан бан бан stream3**.**ReadLine**();**

бан бан бан бан int бан hall бан **=** бан 0**;**

бан бан бан бан **while** бан **(**stream3**.**Peek**()** бан **!=** бан **-**1**)**

бан бан бан бан **{**

бан бан бан бан LCount бан **=** бан int**.**Parse**(**stream3**.**ReadLine**());**

бан бан бан бан **if** бан **(**LCount бан **!=** бан 0**)**

бан бан бан бан **for** бан **(**int бан i бан **=** бан 1**;** бан i бан **<=** бан LCount**;)**

бан бан бан бан **{**

бан бан бан бан string бан inf бан **=** бан stream3**.**ReadLine**();**

бан бан бан бан gName бан **=** бан int**.**Parse**(**stream3**.**ReadLine**());**

бан бан бан бан string бан name бан **=** бан Convert**.**ToString**(**dataGridView1**.**Rows**[**hall**].**Cells**[**0**].**Value**);**

бан бан бан бан news**.**FindFilm**(**name**).**addSeans**(**inf**,**gName**);**

бан бан бан бан **if** бан **(**i бан **!=** бан LCount**)**

бан бан бан бан i**++;**

бан бан бан бан **else**

бан бан бан бан **break;**

бан бан бан бан **}**

бан бан бан бан hall**++;**

бан бан бан бан **}**

**}**

**else**

**{**

бан бан бан бан MessageBox**.**Show**(**"Данный бан файл бан не бан подходит бан для бан работы бан с бан программой"**);**

**}**

stream3**.**Close**();**

**}**

**}**

**private** бан void бан butSave\_Click**(object** бан sender**,** бан EventArgs бан e**)**

**{**

string бан path**;**

string бан fileText бан **=** бан "Файл\n"**;**

//Создание бан экземпляра бан диалогового бан окна бан (Для бан того бан чтобы бан не бан занимать бан поток) бан + бан фильтр бан на бан создание бан текстовых бан файлов

**using** бан **(**SaveFileDialog бан saveFile бан **=** бан **new** бан SaveFileDialog**()** бан **{** бан Filter бан **=** бан "txt бан files бан (\*.txt)|\*.txt|All бан files бан (\*.\*)|\*.\*" бан **})**

**{**

**if** бан **(**saveFile**.**ShowDialog**()** бан **==** бан DialogResult**.**OK**)**

**{**

бан бан бан бан path бан **=** бан saveFile**.**FileName**;**

бан бан бан бан fileText бан **+=** бан news**.**getinfoName**()** бан **+** бан "\n"**;**

бан бан бан бан **for(**int бан i бан **=** бан 0**;** бан i бан **<** бан news**.**COUNT**;** бан i**++)**

бан бан бан бан **{**

бан бан бан бан fileText бан **+=** бан news**.**getinfo**();**

бан бан бан бан **}**

бан бан бан бан //Та бан же бан работа бан с бан экземплярами бан файла бан и бан записи бан файла

бан бан бан бан **using** бан **(**FileStream бан file бан **=** бан **new** бан FileStream**(**path**,** бан FileMode**.**OpenOrCreate**))**

бан бан бан бан **using** бан **(**StreamWriter бан stream бан **=** бан **new** бан StreamWriter**(**file**))**

бан бан бан бан **{**

бан бан бан бан stream**.**Write**(**fileText**);**

бан бан бан бан stream**.**Close**();**

бан бан бан бан file**.**Close**();**

бан бан бан бан **}**

**}**

**}**

**}**

**}**

**}**

**using** бан System**;**

**using** бан System**.**Collections**.**Generic**;**

**using** бан System**.**Text**;**

**namespace** бан Third

**{**

class бан Film

**{**

**private** бан Stack бан firstSeans**;**//первый бан сеанс

**private** бан string бан nameFilm**;**//название бан фильма

**private** бан int бан durationFilm**;**//длина бан фильма

**private** бан Stack бан SP**;**

**private** бан int бан count**;**

**public** бан Film**(**string бан \_nameFilm**,** бан int бан \_durationFilm**)**

**{**

**this.**nameFilm бан **=** бан \_nameFilm**;**

**this.**durationFilm бан **=** бан \_durationFilm**;**

SP бан **=** бан **null;**

**}**

**public** бан int бан Count бан **{** бан **get** бан **{** бан **return** бан count**;** бан **}** бан **}**

**public** бан string бан Name бан **{** бан **get** бан **{** бан **return** бан nameFilm**;** бан **}** бан **set** бан **{** бан **this.**nameFilm бан **=** бан **value;** бан **}** бан **}**

**public** бан int бан Durat бан **{** бан **get** бан **{** бан **return** бан durationFilm**;** бан **}** бан **set** бан **{** бан **this.**durationFilm бан **=** бан **value;** бан **}** бан **}**

**public** бан Stack бан getSPEl бан **{** бан **get{** бан **return** бан SP**;** бан **}** бан **}**

**public** бан void бан addSeans**(**string бан \_date**,** бан int бан \_time**)**

**{**

Seans бан newSea бан **=** бан **new** бан Seans**(**\_date**,**\_time**);**

Stack бан pTemp бан **=** бан **new** бан Stack**(**newSea**,**SP**);**

**if** бан **(**SP бан **==** бан **null)** бан **{** бан firstSeans бан **=** бан pTemp**;}**

SP бан **=** бан pTemp**;**

count**++;**

**}**

**public** бан string бан getNameFilm**()** бан **{** бан **return** бан nameFilm**;** бан **}**

**public** бан bool бан delSeans**()**

**{**

**if** бан **(**SP бан **!=** бан **null)**

**{**

SP бан **=** бан SP**.**getPred**();**

count**--;**

**return** бан **true;**

**}**

**else** бан **return** бан **false;**

**}**

**public** бан int бан getSumStack**()**

**{**

**return** бан durationFilm**\***count**;**

**}**

**public** бан String бан getInfoStack**()**

**{**

String бан info бан **=** бан Convert**.**ToString**(**Count**);**

Stack бан pTemp бан **=** бан SP**;**

**while** бан **(**pTemp бан **!=** бан **null)**

**{**

info бан **+=** бан "\n" бан **+** бан pTemp**.**\_aDate**.**getDate**();**

info бан **+=** бан "\n" бан **+** бан pTemp**.**\_aDate**.**getTime**();**

pTemp бан **=** бан pTemp**.**getPred**();**

**}**

**return** бан info**;**

**}**

**public** бан Seans бан FindSeans**(**string бан \_nameSeans**)**

**{**

Stack бан vspom бан = бан SP;

while(vspom бан != бан null)

{

if(vspom.\_aDate.getDate().Equals(\_nameSeans))

{

бан бан бан бан return бан vspom.\_aDate;

}

else бан { бан vspom бан = бан vspom.getPred(); бан }

}

return бан null;

}

}

}

**using** бан System**;**

**using** бан System**.**Collections**.**Generic**;**

**using** бан System**.**Text**;**

**namespace** бан Third

**{**

class бан Seans

**{**

**private** бан string бан date**;**//дата бан показа бан фильма

**private** бан int бан time**;**//время бан показа бан фильма

**public** бан Seans**(**string бан \_date**,** бан int бан \_time**)**

**{**

**this.**date бан **=** бан \_date**;**

**this.**time бан **=** бан \_time**;**

**}**

**public** бан string бан getDate**()**

**{**

**return** бан date**;**

**}**

**public** бан void бан setDate**(**string бан \_date**)**

**{**

**this.**date бан **=** бан \_date**;**

**}**

**public** бан int бан getTime**()**

**{**

**return** бан time**;**

**}**

**public** бан void бан setTime**(**int бан \_time**)**

**{**

**this.**time бан **=** бан \_time**;**

**}**

**}**

**}**

**using** бан System**;**

**using** бан System**.**Collections**.**Generic**;**

**using** бан System**.**Text**;**

**namespace** бан Third

**{**

class бан Stack

**{**

**public** бан Seans бан \_aDate**;**

**private** бан Stack бан pred**;**

**public** бан Stack**(**Seans бан \_date**,** бан Stack бан \_SP**)**

**{**

\_aDate бан **=** бан \_date**;**

pred бан **=** бан \_SP**;**

**}**

**public** бан Stack бан getPred**()**

**{**

**return** бан pred**;**

**}**

**}**

**}**

**using** бан System**;**

**using** бан System**.**Collections**.**Generic**;**

**using** бан System**.**Text**;**

**namespace** бан Third

**{**

class бан Cinema

**{**

**private** бан Film**[]** бан Queue**;**//реализация бан очереди бан на бан основе бан массива

**private** бан static бан int бан size бан **=** бан 10**;**

**private** бан int бан countFilm бан **=** бан 0**;**

**private** бан int бан first**;**//ссылка бан на бан первый бан элемент бан в бан очереди

**private** бан int бан last**;**//ссылка бан на бан последний бан элемент бан в бан очереди

**public** бан Cinema**()**

**{**

first бан **=** бан 0**;**

last бан **=** бан 0**;**

Queue бан **=** бан **new** бан Film**[**size**];**

**}**

**public** бан string бан addFilm**(**string бан \_nameFilm**,** бан int бан \_durationFilm**)**

**{**

**if** бан **(**size бан **>=** бан countFilm**)**

**{**

**if** бан **(**last бан **>** бан size бан **&&** бан first бан **>** бан 0**)** бан **{**last бан **=** бан 0**;}**

Film бан newFilm бан **=** бан **new** бан Film**(**\_nameFilm**,** бан \_durationFilm**);**

Queue**[**last**]** бан **=** бан newFilm**;**

last**++;**

countFilm**++;**

**return** бан \_nameFilm**;**

**}**

**return** бан ""**;**

**}**

**public** бан bool бан delFilm**()**

**{**

**if** бан **(** бан countFilm бан **!=** бан 0**)**

**{**

**if** бан **(**first бан **>** бан size**)** бан **{** бан first бан **=** бан 0**;}**

Queue**[**first**]** бан **=** бан **null;**

countFilm**--;**

first**++;**

**return** бан **true;**

**}**

**return** бан **false;**

**}**

**public** бан int бан COUNT бан **{** бан **get** бан **{** бан **return** бан countFilm**;** бан **}** бан **}**

**public** бан string бан getinfo**()**

**{**

string бан inform бан **=** бан ""**;**

**for(**int бан i бан **=** бан 0**;** бан i бан **<** бан countFilm**;** бан i**++)**

**{**

inform бан **+=** бан "\n" бан **+** бан Queue**[**i**].**getInfoStack**();**

**}**

**return** бан inform**;**

**}**

**public** бан string бан getinfoName**()**

**{**

string бан inform бан **=** бан Convert**.**ToString**(**COUNT**);**

**for** бан **(**int бан i бан **=** бан 0**;** бан i бан **<** бан countFilm**;** бан i**++)**

**{**

inform бан **+=** бан "\n" бан **+** бан Queue**[**i**].**Name**;**

inform бан **+=** бан "\n" бан **+** бан Queue**[**i**].**Durat**;**

**}**

**return** бан inform**;**

**}**

**public** бан Film бан getFirst**()** бан **{** бан **return** бан Queue**[**first**];** бан **}**

**public** бан Film бан FindFilm**(**string бан \_nameFilm**)**

**{**

Film бан pTemp**;**

**for(**int бан i бан **=** бан 0**;** бан i**<=** бан countFilm**;** бан i**++)**

**{**

pTemp бан **=** бан Queue**[**i**];**

**if** бан **(**pTemp бан **!=** бан **null** бан **&&** бан pTemp**.**getNameFilm**().**Equals**(**\_nameFilm**))**

**{**

бан бан бан бан **return** бан pTemp**;**

**}**

**}**

**return** бан **null;**

**}**

**public** бан bool бан getSeansToEdit**(**string бан \_date**,** бан int бан \_time**,** бан String бан \_nameFilm**)**

**{**

int бан i бан **=** бан 0**;**

Film бан pTemp бан **=** бан Queue**[**i**];**

**for** бан **(**int бан j бан **=** бан 0**;** бан j бан **<** бан size**;** бан j**++)**

**{**

**if** бан **(**pTemp бан **!=** бан **null** бан **&&** бан pTemp**.**getNameFilm**().**Equals**(**\_nameFilm**))**

**{**

бан бан бан бан pTemp**.**addSeans**(**\_date**,** бан \_time**);**

бан бан бан бан **return** бан **true;**

**}**

**else**

**{**

бан бан бан бан i**++;**

бан бан бан бан pTemp бан **=** бан Queue**[**i**];**

**}**

бан бан бан

**}**

**return** бан **false;**

**}**

**}**

**}**