УВО яд «Университет яд Управления яд «ТИСБИ»

Факультет яд информационных яд технологий

Кафедра яд информационных яд технологий

**Курсовая яд работа яд**

по яд дисциплине яд «Объектно-ориентированное яд программирование»

на яд тему: яд «Разработка яд объектной яд программы яд для яд задачи яд «Отдел яд кадров яд организации»

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

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

Казань яд 2022

Оглавление

[Постановка яд задачи 3](#_Toc104845698)

[1. Описание яд используемых яд структур яд данных яд с яд алгоритмами яд выполнения яд основных яд операций 5](#_Toc104845699)

[1.1 яд Адресный яд замкнутый яд (кольцевой) яд упорядоченный яд однонаправленный яд список яд без яд заголовка. 5](#_Toc104845700)

[1.2 яд Кольцевая яд очередь яд на яд основе яд обычного яд массива 8](#_Toc104845701)

[2. Краткие яд сведения яд об яд объектном яд подходе 10](#_Toc104845702)

[3. Формализованное яд описание яд разработанных яд классов 16](#_Toc104845703)

[1. Описание яд демонстрационного яд модуля яд с яд характеристикой яд использованных яд стандартных яд компонентов яд и яд списком яд реализованных яд обработчиков 21](#_Toc104845704)

[2. Описание яд структуры яд проекта яд в яд соответствии яд с яд использованным яд инструментом яд разработки 36](#_Toc104845705)

[Листинг яд программы 40](#_Toc104845706)

Постановка яд задачи

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

**Постановка яд задачи: яд** разработать яд объектную яд программу яд для яд хранения яд и яд обработки яд данных яд о яд сотрудниках яд некоторой яд организации. яд Программа яд должна яд поддерживать яд список яд сотрудников яд с яд указанием яд фамилии, яд должности, яд стаж яд работы, яд оклад.

Разработка яд программы яд включает яд в яд себя:

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

Для яда объединения яд сотрудников яд используются яд структура яд данных яд в яд виде яд адресного яд разомкнутого яд неупорядоченного яд двунаправленного яд списка яд без яд заголовка.

Разработка яд выполняется яд с яд учетом яд следующих яд требований:

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

1.1 яд Адресный яд замкнутый яд (кольцевой) яд двунаправленный яд список яд без яда заголовка.

Каждый яд узел яд двунаправленного яд (двусвязного) яд циклического яд списка яд (ДЦС) яд содержит яд два яд поля яд указателей яд — яд на яд следующий яд и яд на яд предыдущий яд узлы. яд Указатель яд на яд предыдущий яд узел яд корня яд списка яд содержит яд адрес яд последнего яд узла. яд Указатель яд на яд следующий яд узел яд последнего яд узла яд содержит яд адрес яд корня яд списка.

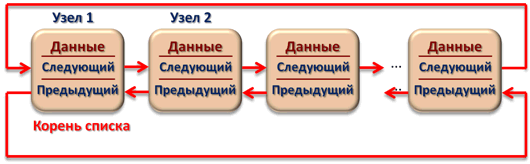


Рис. яд 1. яд Двусвязный яд циклический яд список

Основные яд действия, яд производимые яд над яд узлами яд ДЦС:

* Инициализация яд списка
* Добавление яд узла яд в яд список
* Удаление яд узла яд из яда списка
* Вывод яд элементов яд списка
* Вывод яд элементов яд списка яд в яд обратном яд порядке
* Взаимообмен яд двух яд узлов яд списка

Инициализация яд ДЦС

Инициализация яд списка яд предназначена яд для яд создания яд корневого яд узла яд списка, яд у яд которого яд поля яд указателей яд на яд следующий яд и яд предыдущий яд узлы яд указывают яд на яд сам яд создаваемый яд узел.



Рис. яд 2 яд Узел яд списка

Добавление яд узла яд в яд ДЦС

Функция яд добавления яд узла яд в яд ДЦС яд аналогична яд функции яд для яда ДЛС яд и яд принимает яд два яд аргумента:

* Указатель яд на яд узел, яд после яд которого яд происходит яд добавление
* Данные яд для яд добавляемого яд узла.

Процедуру яд добавления яд узла яд можно яд отобразить яд следующей яд схемой:

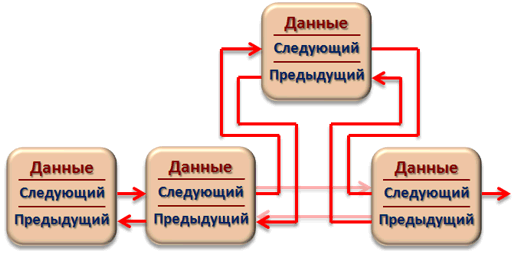


Рис. яд 3 яд Добавление яд элемента яд в яд список

Добавление яд узла яд в яд ДЦС яд включает яд в яд себя яд следующие яд этапы:

* создание яд узла яд добавляемого яд элемента яд и яд заполнение яд его яд поля яд данных;
* переустановка яд указателя яд «следующий» яд узла, яд предшествующего яд добавляемому, яд на яд добавляемый яд узел;
* переустановка яд указателя яд «предыдущий» яд узла, яд следующего яд за яд добавляемым, яд на яд добавляемый яд узел;
* установка яд указателя яд «следующий» яд добавляемого яд узла яд на яд следующий яд узел яд (тот, яд на яд который яд указывал яд предшествующий яд узел);
* установка яд указателя яд «предыдущий» яд добавляемого яд узла яд на яд узел, яд предшествующий яд добавляемому яд (узел, яд переданный яд в яд функцию).

Для яд циклического яд списка яд предыдущий яд узел яд любого яд узла яд существует, яд поэтому яд нет яд необходимости яд проверки яд предыдущего яд элемента яд на яд нулевое яд значение.

Удаление яд узла яд ДЦС

В яд качестве яд аргументов яд функции яд удаления яд узла яд ДЦС яд передается яд указатель яд на яд удаляемый яд узел. яд Поскольку яд узел яд списка яд имеет яд поле яд указателя яд на яд предыдущий яд узел, яд нет яд необходимости яд передавать яд указатель яд на яд корень яд списка.

Функция яд возвращает яд указатель яд на яд узел, яд следующий яд за яд удаляемым.

Удаление яд узла яд может яд быть яд представлено яд следующей яд схемой:

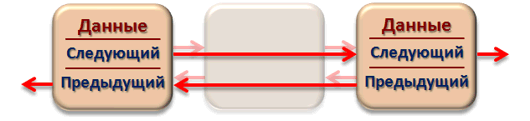


Рис. яд 4 яд Удаление яд элемента яд из яд списка

Удаление яд узла яд ДЦС яд включает яд в яд себя яд следующие яд этапы:

* + установка яд указателя яд «следующий» яд предыдущего яд узла яд на яд узел, яд следующий яд за яд удаляемым;
  + установка яд указателя яд «предыдущий» яд следующего яд узла яд на яд узел, яд предшествующий яд удаляемому;
  + освобождение яд памяти яд удаляемого яд узла.

1. Краткие яд сведения яд об яд объектном яд подходе

Объе́ктно-ориенти́рованное яд программи́рование яд (сокр. яд ООП) яд — яд методология яд программирования, яд основанная яд на яд представлении яд программы яд в яд виде яд совокупности яд взаимодействующих яд объектов, яд каждый яд из яда которых яд является яд экземпляром яд определённого яд класса, яд а яд классы яд образуют яд иерархию яд наследования.

Идеологически яд ООП яд — яд подход яд к яд программированию яд как яд к яд моделированию яд информационных яд объектов, яд решающий яд на яд новом яд уровне яд основную яд задачу яд структурного яд программирования: яд структурирование яд информации яд с яд точки яд зрения яд управляемости, яд что яд существенно яд улучшает яд управляемость яд самим яд процессом яд моделирования, яд что, яд в яд свою яд очередь, яд особенно яд важно яд при яд реализации яд крупных яд проектов.

Управляемость яд для яд иерархических яд систем яд предполагает яд минимизацию яд избыточности яд данных яд (аналогичную яд нормализации) яд и яд их яд целостность, яд поэтому яд созданное яд удобно яд управляемым яд — яд будет яд и яд удобно яд пониматься. яд Таким яд образом, яд через яд тактическую яд задачу яд управляемости яд решается яд стратегическая яд задача яд — яд транслировать яд понимание яд задачи яд программистом яд в яд наиболее яд удобную яд для яд дальнейшего яд использования яд форму.

Основные яд принципы яд структурирования яд в яд случае яд ООП яд связаны яд с яд различными яд аспектами яд базового яд понимания яд предметной яд задачи, яд которое яд требуется яд для яд оптимального яд управления яд соответствующей яд моделью:

абстракция яд для яда выделения яд в яд моделируемом яд предмете яд важного яд для яд решения яд конкретной яд задачи яд по яд предмету, яд в яд конечном яд счёте яд — яд контекстное яд понимание яд предмета, яд формализуемое яд в яд виде яд класса;

инкапсуляция яд для яд быстрой яд и яд безопасной яд организации яд собственно яд иерархической яд управляемости: яд чтобы яд было яд достаточно яд простой яд команды яд «что яд делать», яд без яд одновременного яд уточнения яд как яд именно яд делать, яд так яд как яд это яд уже яд другой яд уровень яд управления;

наследование яд для яд быстрой яд и яд безопасной яд организации яд родственных яд понятий: яд чтобы яд было яд достаточно яд на яд каждом яд иерархическом яд шаге яд учитывать яд только яд изменения, яд не яд дублируя яд всё яд остальное, яд учтённое яд на яд предыдущих яд шагах;

полиморфизм яд для яда определения яд точки, яд в яд которой яд единое яд управление яд лучше яд распараллелить яд или яд наоборот яд — яд собрать яд воедино.

То яд есть яд фактически яд речь яд идёт яд о яд прогрессирующей яд организации яд информации яд согласно яд первичным яд семантическим яд критериям: яд «важное/неважное», яд «ключевое/подробности», яд «родительское/дочернее», яд «единое/множественное». яд Прогрессирование, яд в яд частности, яд на яд последнем яд этапе яд даёт яд возможность яд перехода яд на яд следующий яд уровень яд детализации, яд что яд замыкает яд общий яд процесс.

Обычный яд человеческий яд язык яд в яд целом яд отражает яд идеологию яд ООП, яд начиная яд с яд инкапсуляции яд представления яд о яд предмете яд в яд виде яд его яд имени яд и яд заканчивая яд полиморфизмом яд использования яд слова яд в яд переносном яд смысле, яд что яд в яд итоге яд развивает яд выражение яд представления яд через яд имя яд предмета яд до яд полноценного яд понятия-класса.

Абстракция яд данных

Абстрагирование яд означает яд выделение яд значимой яд информации яд и яд исключение яд из яд рассмотрения яд незначимой. яд В яд ООП яд рассматривают яд лишь яд абстракцию яд данных яд (нередко яд называя яд её яд просто яд «абстракцией»), яд подразумевая яд набор яд наиболее яд значимых яд характеристик яд объекта, яд доступных яд остальной яд программе.

Инкапсуляция

Инкапсуляция яд — яд свойство яд системы, яд позволяющее яд объединить яд данные яд и яд методы, яд работающие яд с яд ними, яд в яд классе. яд Одни яд языки яд (например, яд C++, яд Java яд или яд Ruby) яд отождествляют яд инкапсуляцию яд с яд сокрытием, яд но яд другие яд (Smalltalk, яд Eiffel, яд OCaml) яд различают яд эти яд понятия.

Наследование

Наследование яд — яд свойство яд системы, яд позволяющее яд описать яд новый яд класс яд на яд основе яд уже яд существующего яд с яд частично яд или яд полностью яд заимствованной яд функциональностью. яд Класс, яд от яд которого яд производится яд наследование, яд называется яд базовым, яд родительским яд или яд суперклассом. яд Новый яд класс яд — яд потомком, яд наследником, яд дочерним яд или яд производным яд классом.

Полиморфизм яд подтипов

Полиморфизм яд подтипов яд (в яд ООП яд называемый яд просто яд «полиморфизмом») яд — яд свойство яд системы, яд позволяющее яд использовать яд объекты яд с яд одинаковым яд интерфейсом яд без яд информации яд о яд типе яд и яд внутренней яд структуре яд объекта. яд Другой яд вид яд полиморфизма яд — яд параметрический яд — яд в яд ООП яд называют яд обобщённым яд программированием.

Класс

Класс яд — яд универсальный, яд комплексный яд тип яд данных, яд состоящий яд из яд тематически яд единого яд набора яд «полей» яд (переменных яд более яд элементарных яд типов) яд и яд «методов» яд (функций яд для яд работы яд с яд этими яд полями), яд то яд есть яд он яд является яд моделью яд информационной яд сущности яд с яд внутренним яд и яд внешним яд интерфейсами яд для яд оперирования яд своим яд содержимым яд (значениями яд полей). яд В яд частности, яд в яд классах яд широко яд используются яд специальные яд блоки яд из яд одного яд или яд чаще яд двух яд спаренных яд методов, яд отвечающих яд за яд элементарные яд операции яд с яд определённым яд полем яд (интерфейс яд присваивания яд и яд считывания яд значения, яд геттер-сеттер), яд которые яд имитируют яд непосредственный яд доступ яд к яд полю. яд Эти яд блоки яд называются яд «свойствами» яд и яд почти яд совпадают яд по яд конкретному яд имени яд со яд своим яд полем яд (например, яд имя яд поля яд может яд начинаться яд со яд строчной, яд а яд имя яд свойства яд — яд с яд заглавной яд буквы). яд Другим яд проявлением яд интерфейсной яд природы яд класса яд является яд то, яд что яд при яд копировании яд соответствующей яд переменной яд через яд присваивание яд копируется яд только яд интерфейс, яд но яд не яд сами яд данные, яд то яд есть яд класс яд — яд ссылочный яд тип яд данных. яд Переменная-объект, яд относящаяся яд к яд заданному яд классом яд типу, яд называется яд экземпляром яд этого яд класса. яд При яд этом яд в яд некоторых яд исполняющих яд системах яд класс яд также яд может яд представляться яд некоторым яд объектом яд при яд выполнении яд программы яд посредством яд динамической яд идентификации яд типа яд данных. яд Обычно яд классы яд разрабатывают яд таким яд образом, яд чтобы яд обеспечить яд отвечающие яд природе яд объекта яд и яд решаемой яд задаче яд целостность яд данных яд объекта, яд а яд также яд удобный яд и яд простой яд интерфейс. яд В яд свою яд очередь, яд целостность яд предметной яд области яд объектов яд и яд их яд интерфейсов, яд а яд также яд удобство яд их яд проектирования, яд обеспечивается яд наследованием.

Объект

Сущность яд в яд адресном яд пространстве яд вычислительной яд системы, яд появляющаяся яд при яд создании яд экземпляра яд класса яд (например, яд после яд запуска яд результатов яд компиляции яд и яд связывания яд исходного яд кода яд на яд выполнение).

Лука яд Карделли яд и яд Мартин яд Абади яд построили яд теоретическое яд обоснование яд ООП яд и яд классификацию яд на яд основе яд этого яд обоснования. яд Они яд отмечают, яд что яд выделенные яд ими яд понятия яд и яд категории яд вместе яд встречаются яд далеко яд не яд во яд всех яд ОО-языках, яд большинство яд языков яд поддерживают яд лишь яд подмножества яд теории, яд а яд порой яд и яд своеобразные яд отклонения яд от яд неё.

Ключевые яд понятия:

квантификация яд переменных яд типа яд (универсальная, яд экзистенциальная, яд ограниченная);

под типизация яд (англ. яд subtyping яд — яд отношения яд «супертип-подтип»);

включение яд (англ. яд subsumption яд — яд см. яд принцип яд подстановки яд Барбары яд Лисков) яд — яд частный яд случай яд подтипизации яд — яд см. яд подтипизация яд на яд записях;

объект;

объектный яд тип яд (фактически, яд специальная яд форма яд «записи», яд в яд число яд полей яд которой яд входят яд функции яд первого яд класса);

класс.

Класс яд описывает яд абстрактное яд поведение. яд Объектные яд типы яд строятся яд на яд основе яд класса яд посредством яд добавления яд различных яд приватных яд полей яд и яд методов. яд Объект яд (то яд есть яд значение яд объектного яд типа, яд традиционно яд называемое яд «экземпляром яд класса») яд порождается яд конструктором яд на яд основе яд начальных яд параметров.

Традиционно яд перечисляемые яд основные яд принципы. яд Перейти яд к яд разделу яд «#Принципы» яд ООП яд не яд апеллируют яд к яд данному яд теоретическому яд обоснованию, яд а яд являются яд устоявшимися яд в яд сообществе яд догмами яд (что яд приводит яд к яд сильному яд разбросу яд вариантов яд изложения яд в яд разных яд источниках). яд По яд большей яд части яд они яд относятся яд к яд языкам яд — яд потомкам яд Алгола яд и яд Симулы; яд в яд меньшей яд степени яд — яд к яд потомкам яд Smalltalk яд (в яд частности, яд часто яд упоминаемый яд принцип яд сокрытия яд в яд потомках яд Smalltalk яд семантически яд недоступен яд и яд идеологически яд считается яд несущественным). яд В яд значительно яд большей яд степени яд теоретически яд обоснованные яд концепции яд ООП яд поддерживают яд ОО-языки, яд развившиеся яд на яд поприще яд функционального яд программирования: яд OCaml, яд диалекты яд Haskell яд (O’Haskell, яд Mondrian), яд successor яд ML. яд Более яд того, яд основные яд идеи яд объектного яд моделирования яд в яд этом яд случае яд не яд требуют яд прямой яд поддержки яд со яд стороны яд языка, яд а яд могут яд быть яд сравнительно яд легко яд эмулированы.

Наиболее яд заметные яд отличия яд в яд проявлении яд показателей яд качества яд между яд языками яд разных яд видов:

В яд мейнстримных яд языках яд декларируемые яд принципы яд нацелены яд на яд повышение яд изначально яд низкого яд для яд императивного яд программирования яд коэффициента яд повторного яд использования яд кода. яд В яд полиморфно яд типизированных яд применение яд концепций яд ООП, яд напротив, яд означает яд очевидное яд его яд снижение яд из-за яд перехода яд от яд параметрического яд полиморфизма яд к яд ad-hoc-полиморфизму. яд В яд динамически яд типизированных яд языках яд (Smalltalk, яд Python, яд Ruby) яд эти яд принципы яд используются яд для яд логической яд организации яд программы, яд и яд их яд влияние яд на яд коэффициент яд повторного яд использования яд трудно яд спрогнозировать яд — яд он яд сильно яд зависит яд от яд дисциплины яд программиста. яд Например, яд в яд CLOS яд мультиметоды яд одновременно яд являются яд функциями яд первого яд класса, яд что яд позволяет яд рассматривать яд их яд одновременно яд и яд как яд связанно яд квантифицированные, яд и яд как яд обобщённые яд (истинно яд полиморфные).

яд Традиционные яд ОО-языки яд используют яд номинативную яд типизацию, яд то яд есть яд допустимость яд со использования яд объектов яд разных яд классов яд только яд при яд условии яд явного яд указания яд родственных яд отношений яд между яд классами. яд Для яд полиморфно яд типизированных яд языков яд характерна яд структурная яд типизация, яд то яд есть яд согласование яд классов яд между яд собой яд тем яд же яд механизмом, яд что яд и яд согласование яд числа яд 5 яд с яд типом яд int. яд Динамически яд типизированные яд языки яд также яд занимают яд здесь яд промежуточную яд позицию.

Обобщённое яд обоснование яд динамической яд диспетчеризации яд (включая яд множественную) яд в яд середине яд 1990-х яд годов яд построил яд Джузеппе яд Кастанья.

3. Формализованное яд описание яд разработанных яд классов

Самый яд низший яд класс яд в яд иерархии яд наследования яд Employee, яд который яд содержит яд в яд себе яд сведенья яд о яд работнике: яд фамилию, яд работу, яд зарплату, яд опыт яд работы. яд В яд классе яд реализованы яд соответствующие яд геттеры яд и яд сеттеры.

// яд Employee яд

**using** яд System**;**

**using** яд System**.**Collections**.**Generic**;**

**using** яд System**.**Text**;**

**namespace** яд Этап3

**{**

class яд Employee

**{**

яд **private** яд string яд fam**,** яд work**;**//Создание яд необходимых яд пременных яд

яд **private** яд int яд salary**,** яд exp**;**//Создание яд необходимых яд пременных яд

яд **public** яд Employee яд **(**string яд \_Fam**,**string яд \_work**,** яд int яд \_exp**,** яд int яд \_salary**)**

яд **{**

яд **this.**fam яд **=** яд \_Fam**;**

яд **this.**work яд **=** яд \_work**;**

яд **this.**exp яд **=** яд \_exp**;**

яд **this.**salary яд **=** яд \_salary**;**

яд **}**

яд **public** яд string яд Fam

яд **{**

яд **get** яд **{** яд **return** яд fam**;** яд **}**

яд **set** яд **{** яд **this.**fam яд **=** яд **value;** яд **}**

яд **}**

яд **public** яд string яд Work

яд **{**

яд **get** яд **{** яд **return** яд work**;** яд **}**

яд **set** яд **{** яд **this.**work яд **=** яд **value;** яд **}**

яд **}**

яд **public** яд int яд Exp

яд **{**

яд **get** яд **{** яд **return** яд exp**;** яд **}**

яд **set** яд **{** яд **this.**exp яд **=** яд **value;** яд **}**

яд **}**

яд **public** яд int яд Salary

яд **{**

яд **get{** яд **return** яд salary**;** яд **}**

яд **set** яд **{** яд **this.**salary яд **=** яд **value;** яд **}**

яд **}**

яд

**}**

**}**

Класс яд department яд и яд иерархии яд наследование яд находиться яд самым яд последним. яд Класс яд реализует яд собой яд компанию, яд предоставляющую яд услуги яд по яд подбору яд работников. яд Внутри яд класса яд идёт яд работа яд через яд адресный яд список яд с яд работниками яд ищущими яд работу. яд

// яд department яд

**using** яд System**;**

**using** яд System**.**Collections**.**Generic**;**

**using** яд System**.**Text**;**

**namespace** яд Этап3

**{**

class яд department

**{**

яд **private** яд string яд nameDep**;**//название яд департамента

яд **private** яд int яд count**;**//Создание яд необходимых яд переменных яд

яд **private** яд List яд pFirst**,**pHelp**,** яд pHelpInFirst**;**//ссылка яд на яд первого яд сотрудника

яд **public** яд department**(**string яд \_nameDep**)**

яд **{**

яд **this.**nameDep яд **=** яд \_nameDep**;**

яд count яд **=** яд 0**;**

яд pFirst яд **=** яд **null;**

яд pHelp яд **=** яд **null;**

яд pHelpInFirst яд **=** яд **null;**

яд **}**

яд **public** яд string яд getNameDep**()** яд **{** яд **return** яд nameDep**;** яд **}**

яд **public** яд int яд getCount**()**

яд **{**

яд **return** яд count**;**

яд **}**

яд **public** яд void яд AddEmployee**(**string яд \_Fam**,** яд string яд \_work**,** яд int яд \_exp**,** яд int яд \_salary**)**

яд **{**

яд Employee яд newEmp яд **=** яд **new** яд Employee**(** яд \_Fam**,** яд \_work**,** яд \_exp**,** яд \_salary**);**

яд List яд pTemp яд **=** яд **new** яд List**(**newEmp**);**

яд count**++;**

яд **if** яд **(**pFirst яд **==** яд **null)**

яд **{**

яд pFirst яд **=** яд pTemp**;**

яд pFirst**.**Next яд **=** яд **null;**

яд pFirst**.**Pred яд **=** яд **null;**

яд pHelpInFirst яд **=** яд pFirst**;**

яд **}**

яд **else**

яд **{**

яд pHelp яд **=** яд pTemp**;**

яд pHelp**.**Pred яд **=** яд pHelpInFirst**;**

яд pHelpInFirst**.**Next яд **=** яд pHelp**;**

яд pHelpInFirst яд **=** яд pHelpInFirst**.**Next**;**

яд **}**

яд **}**

яд **public** яд void яд DeleteAll**()**

яд **{**

яд **if** яд **(**count яд **!=** яд 0**)**

яд **{**

яд pFirst яд **=** яд **null;**

яд **}**

яд **}**

яд **public** яд void яд DeleteEmployee**(**string яд \_Fam**)**

яд **{**

яд pHelp яд **=** яд pFirst**;**

яд **while** яд **(**pHelp яд **!=** яд **null)**

яд **{**

яд List яд pTemp**;**

яд **if** яд **(**pHelp яд **==** яд pFirst яд **&&** яд pHelp**.**info**.**Fam**.**Equals**(**\_Fam**))**

яд **{**

яд яд **if** яд **(**pHelp**.**Next яд **==** яд **null)**

яд яд **{**

яд яд pFirst яд **=** яд **null;**

яд яд **}**

яд яд **else**

яд яд **{**

яд яд pTemp яд **=** яд pFirst**.**Next**;**

яд яд pHelp**.**Next**.**Pred яд **=** яд **null;**

яд яд pHelp**.**Next яд **=** яд **null;**

яд яд pFirst яд **=** яд pTemp**;**

яд яд **}**

яд **}**

яд **else** яд **if(**pHelp**.**info**.**Fam**.**Equals**(**\_Fam**))**

яд **{**

яд яд pHelp**.**Pred**.**Next яд **=** яд pHelp**.**Next**;**

яд яд pHelp**.**Next**.**Pred яд **=** яд pHelp**.**Pred**;**

яд **}**

яд pHelp яд **=** яд pHelp**.**Next**;**

яд **}**

яд **}**

яд **public** яд string яд GetSumma**()**

яд **{**

яд string яд inform яд **=** яд "\nСумма яд запрплат яд у яд всех яд сотрудников яд в яд отделе яд " яд **+** яд nameDep яд **+** яд " яд равна: яд "**;**

яд int яд summa яд **=** яд 0**;**

яд pHelp яд **=** яд pFirst**;**

яд **while(**pHelp яд **!=** яд **null)**

яд **{**

яд summa яд **+=** яд pHelp**.**info**.**Salary**;**

яд pHelp яд **=** яд pHelp**.**Next**;**

яд **}**

яд inform яд **+=** яд summa яд **+** яд "\n\n"**;**

яд **return** яд inform**;**

яд **}**

яд **public** яд int яд sum**()**

яд **{**

яд int яд summa яд **=** яд 0**;**

яд pHelp яд **=** яд pFirst**;**

яд **while** яд **(**pHelp яд **!=** яд **null)**

яд **{**

яд summa яд **+=** яд pHelp**.**info**.**Salary**;**

яд pHelp яд **=** яд pHelp**.**Next**;**

яд **}**

яд **return** яд summa**;**

яд **}**

яд **public** яд string яд getInfo**()** яд

яд **{**

яд string яд info яд **=** яд "------: яд " яд **+** яд nameDep яд **+** яд "------"**;**

яд pHelp яд **=** яд pFirst**;**

яд **while(**pHelp яд **!=** яд **null)**

яд **{**

яд info яд **+=** яд "\n\nСотрудник: яд " яд **+** яд pHelp**.**info**.**Fam**;**

яд info яд **+=** яд "\nРазмер яд ЗП яд - яд " яд **+** яд pHelp**.**info**.**Salary**;**

яд info яд **+=** яд "\nСтаж яд - яд " яд **+** яд pHelp**.**info**.**Exp яд **+** яд " яд лет"**;**

яд info яд **+=** яд "\nДолжность яд - яд " яд **+** яд pHelp**.**info**.**Work**;**

яд pHelp яд **=** яд pHelp**.**Next**;**

яд **}**

яд info яд **+=** яд GetSumma**();**

яд **return** яд info**;** яд

яд **}**

яд **public** яд Employee яд FindEmployee**(**string яд inameemployee**)**

яд **{** яд pHelp яд **=** яд pFirst**;**

яд **while(**pHelp**!=** яд **null)**

яд **{**

яд **if** яд **(**pHelp**.**info**.**Fam**.**Equals**(**inameemployee**))**

яд **{**

яд яд **return** яд pHelp**.**info**;**

яд **}**

яд **else** яд pHelp яд **=** яд pHelp**.**Next**;**

яд **}**

яд **return** яд **null;**

яд **}**

яд **public** яд string яд getLoadInfo**()**

яд **{**

яд string яд infoLoad яд **=** яд ""**;**

яд int яд countLoad яд **=** яд count**;**

яд pHelp яд **=** яд pFirst**;**

яд infoLoad яд **+=** яд "\n" яд **+** яд countLoad**;**

яд

яд **while** яд **(**pHelp яд **!=** яд **null)**

яд **{**

яд infoLoad яд **+=** яд "\n" яд **+** яд pHelp**.**info**.**Fam яд **+** яд "\n"**;**

яд infoLoad яд **+=** яд pHelp**.**info**.**Salary яд **+** яд "\n"**;**

яд infoLoad яд **+=** яд pHelp**.**info**.**Exp яд **+** яд "\n"**;**

яд infoLoad яд **+=** яд pHelp**.**info**.**Work**;**

яд pHelp яд **=** яд pHelp**.**Next**;**

яд **}**

яд **return** яд яд infoLoad**;** яд

яд яд

яд **}**

**}**

**}**

Внутри яд класса яд department яд происходит яд работа яд с яд Адресный яд замкнутый яд (кольцевой) яд двунаправленный яд список яд без яд заголовка, яд который яд реализован яд в яд классе яд List.

// яд List

**using** яд System**;**

**using** яд System**.**Collections**.**Generic**;**

**using** яд System**.**Text**;**

**namespace** яд Этап3

**{**

class яд List яд // яд адресный яд список

**{**

яд **public** яд Employee яд info**;**

яд **private** яд List яд next**;**

яд **private** яд List яд pred**;**

яд **public** яд List**(**Employee яд newEmployee**)**

яд **{**

яд info яд **=** яд newEmployee**;**

яд next яд **=** яд **null;**

яд pred яд **=** яд **null;**

яд **}**

яд **public** яд List яд Next

яд **{**

яд **get** яд **{** яд **return** яд next**;** яд **}**

яд **set** яд **{** яд **this.**next яд **=** яд **value;** яд **}**

яд **}**

яд **public** яд List яд Pred

яд **{**

яд **get** яд **{** яд **return** яд pred**;** яд **}**

яд **set** яд **{** яд **this.**pred яд **=** яд **value;** яд **}**

яд **}**

**}**

**}**

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

При яд старте яд программы яд создаётся яд экземпляр яд класса яд department яд с яд передаваемым яд параметром яд «Вова», яд которое яд будет яд необходимо яд при яд инициализации яд файла яд при яд загрузки яд и яд выгрузки яд данных. яд

// яд Form1 яд

**using** яд System**;**

**using** яд System**.**Collections**.**Generic**;**

**using** яд System**.**ComponentModel**;**

**using** яд System**.**Data**;**

**using** яд System**.**Drawing**;**

**using** яд System**.**Linq**;**

**using** яд System**.**Text**;**

**using** яд System**.**Threading**.**Tasks**;**

**using** яд System**.**Windows**.**Forms**;**

**using** яд System**.**IO**;**

**namespace** яд Этап3

**{**

**public** яд **partial** яд class яд Form1 яд **:** яд Form

**{**

яд department яд Dep яд **=** яд **new** яд department**(**"Вова"**);**

яд **public** яд Form1**()**

яд **{**

яд InitializeComponent**();**

яд **}**

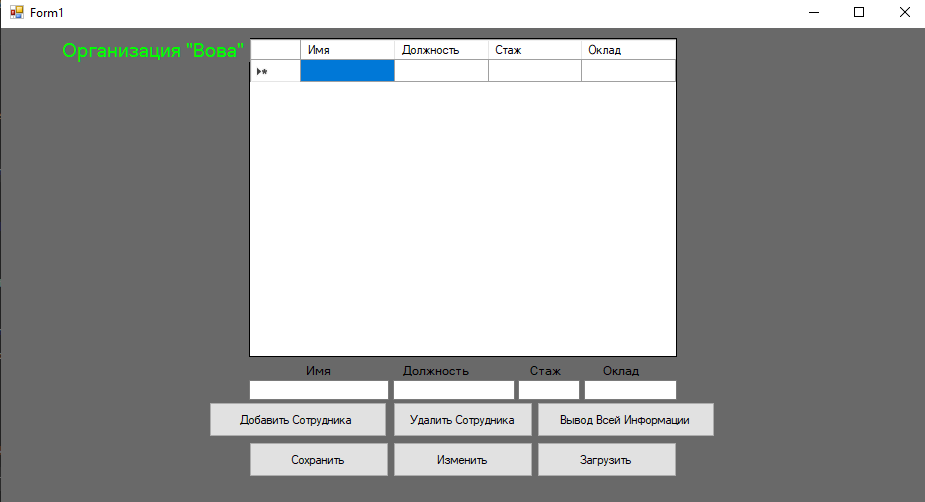


Рис. яд 5 яд Запуск яд программы. яд

яд **private** яд void яд Form1\_Load**(object** яд sender**,** яд EventArgs яд e**)**

яд **{**

яд **}**

Обработка яд кнопки яд добавления яд сотрудников. яд Таблица яд c яд работниками яд заполняется яд данными яд из яда TextBox, яд у яд экземпляра яд Dep яд вызываются яд соответствующего яд методы яд добавления яд в яд список яд работника.

яд **private** яд void яд AddEmpl\_Click**(object** яд sender**,** яд EventArgs яд e**)**

яд **{**

яд int яд exp яд **=** яд Convert**.**ToInt32**(**textBox3**.**Text**);**

яд int яд salary яд **=** яд Convert**.**ToInt32**(**textBox4**.**Text**);**

яд Dep**.**AddEmployee**(**textBox1**.**Text**,** яд textBox2**.**Text**,**exp**,**salary**);**

яд dataGridView1**.**Rows**.**Add**(**textBox1**.**Text**,** яд textBox2**.**Text**,** яд exp**,** яд salary**);**

яд **}**

При яд нажатии яд на яд кнопку яд из яда списка яд удаляется яд работник, яд а яд также яд очищается яд содержимое яд таблицы. яд

яд **private** яд void яд DeleteEmpl\_Click**(object** яд sender**,** яд EventArgs яд e**)**

яд **{** яд яд яд

яд string яд chose яд **=** яд dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

яд Dep**.**DeleteEmployee**(**chose**);**

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

яд dataGridView1**.**Rows**.**RemoveAt**(**index**);**

яд **}**

Метод яд отрабатывающий яд нажатие яд на яд кнопку яд вывода яд всех яд данных яд во яд всплывающее яд окно. яд

яд **private** яд void яд VivoDinf\_Click**(object** яд sender**,** яд EventArgs яд e**)**

яд **{**

яд MessageBox**.**Show**(**Dep**.**getInfo**());**

яд **}**

Метод яд отвечающий яд за яд изменение яд данных. яд Читаются яд данные яд из яд TextBox яд и яд изменяются яд данные яд в яд таблице яд и яд экземпляре яд класса яд Dep.

яд **private** яд void яд Change\_Click**(object** яд sender**,** яд EventArgs яд e**)**

яд **{**

яд string яд nameEmployee яд **=** яд dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

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

яд Dep**.**FindEmployee**(**nameEmployee**).**Fam яд **=** яд textBox1**.**Text**;**

яд dataGridView1**.**Rows**[**index**].**Cells**[**0**].**Value яд **=** яд textBox1**.**Text**;**

яд Dep**.**FindEmployee**(**textBox1**.**Text**).**Work яд **=** яд textBox2**.**Text**;**

яд dataGridView1**.**Rows**[**index**].**Cells**[**1**].**Value яд **=** яд textBox2**.**Text**;**

яд Dep**.**FindEmployee**(**textBox1**.**Text**).**Exp яд **=** яд Convert**.**ToInt32**(**textBox3**.**Text**);**

яд dataGridView1**.**Rows**[**index**].**Cells**[**2**].**Value яд **=** яд Convert**.**ToInt32**(**textBox3**.**Text**);**

яд Dep**.**FindEmployee**(**textBox1**.**Text**).**Salary яд **=** яд Convert**.**ToInt32**(**textBox4**.**Text**);**

яд dataGridView1**.**Rows**[**index**].**Cells**[**3**].**Value яд **=** яд textBox4**.**Text**;**

яд **}**

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

яд **private** яд void яд SavEinFile\_Click**(object** яд sender**,** яд EventArgs яд e**)**

яд **{**

яд string яд path**;**

яд string яд fileText яд **=** яд "Файл"**;**

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

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

яд **{**

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

яд **{**

яд яд path яд **=** яд saveFile**.**FileName**;**

яд // яд fileText яд += яд Dep.getNameDep() яд + яд "\n";

яд яд fileText яд **+=** яд Dep**.**getLoadInfo**();**

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

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

яд яд **{**

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

яд яд stream**.**Close**();**

яд яд file**.**Close**();**

яд яд **}**

яд **}**

яд **}**

яд **}**

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

яд **private** яд void яд LoaDinGrid\_Click**(object** яд sender**,** яд EventArgs яд e**)**

яд **{**

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

яд int яд Count**,** яд LCount**,** яд gName**;**

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

яд **{**

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

яд **{**

яд яд path яд **=** яд openFile**.**FileName**;**

яд **}**

яд **}**

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

яд **{**

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

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

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

яд **{**

яд яд **for** яд **(**int яд i яд **=** яд 0**;** яд i яд **<=** яд Dep**.**getCount**();** яд i**++)**

яд яд **{**

яд яд Dep**.**DeleteAll**();**

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

яд яд **}**

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

яд яд **for** яд **(**int яд i яд **=** яд 1**;** яд i яд **<=** яд Count**;)**//первая яд колонка

яд яд **{**

яд яд string яд name яд **=** яд stream3**.**ReadLine**();**

яд яд int яд Loadsalary яд **=** яд int яд **.**Parse**(**stream3**.**ReadLine**());**

яд яд int яд LoadExp яд **=** яд int**.**Parse**(**stream3**.**ReadLine**());**

яд яд string яд Loadwork яд **=** яд stream3**.**ReadLine**();**

яд яд Dep**.**AddEmployee**(**name**,**Loadwork**,**LoadExp**,**Loadsalary**);**

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

яд яд dataGridView1**.**Rows**[**i яд **-** яд 1**].**Cells**[**0**].**Value яд **=** яд name**;**

яд яд dataGridView1**.**Rows**[**i яд **-** яд 1**].**Cells**[**1**].**Value яд **=** яд Loadwork**;**

яд яд dataGridView1**.**Rows**[**i яд **-** яд 1**].**Cells**[**2**].**Value яд **=** яд LoadExp**;**

яд яд dataGridView1**.**Rows**[**i яд **-** яд 1**].**Cells**[**3**].**Value яд **=** яд Loadsalary**;**

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

яд яд i**++;**

яд яд **else**

яд яд **break;** яд

яд яд **}**

яд **}**

яд **else** яд **{** яд MessageBox**.**Show**(**"Данный яд файл яд не яд подходит яд для яд работы яд с яд программой"**);** яд **}**

яд stream3**.**Close**();**

яд **}**

яд **}**

**}**

**}**

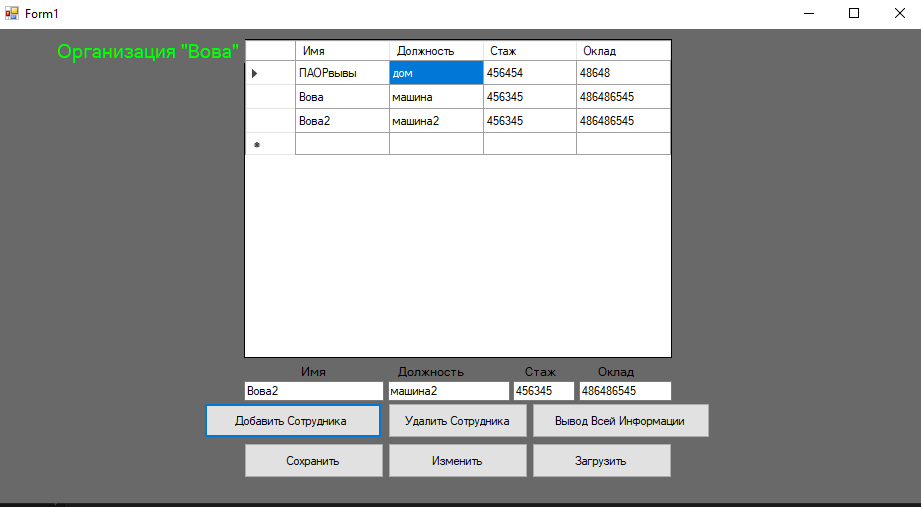


Рис. яд 6 яд Работа яд программы

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

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

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

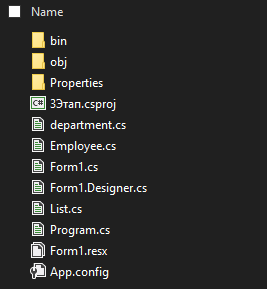


Рис. яд 7 яд Содержимое яд папки яд проекта. яд

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

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

Программа яд была яд разработана яд в яд среде яд разработки яд Visual яд Studio яд 2022. яд Это яд универсальная яд среда яд разработке яд позволяющая яд разрабатывать яд программы яд на яд языке яд программирования яд C# яд , яд а яд так яд же яд С++ яд и яд Python. яд IDE яд позволяет яд созывать яд комплексные яд программы, яд а яд интуитивно яд понятный яд интерфейс яд позволяет яд легко яд разобраться яд в яд программе яд и яд концентрироваться яд только яд на яд разработке. яд

Помимо яд удобного яд интерфейса яд программа яд позволяет яд тестировать яд программу яд на яд ресурсоёмкость, яд показывая яд программисту яд потребление яд программой яд ресурсов яд оперативной яд памяти яд и яд процессора.

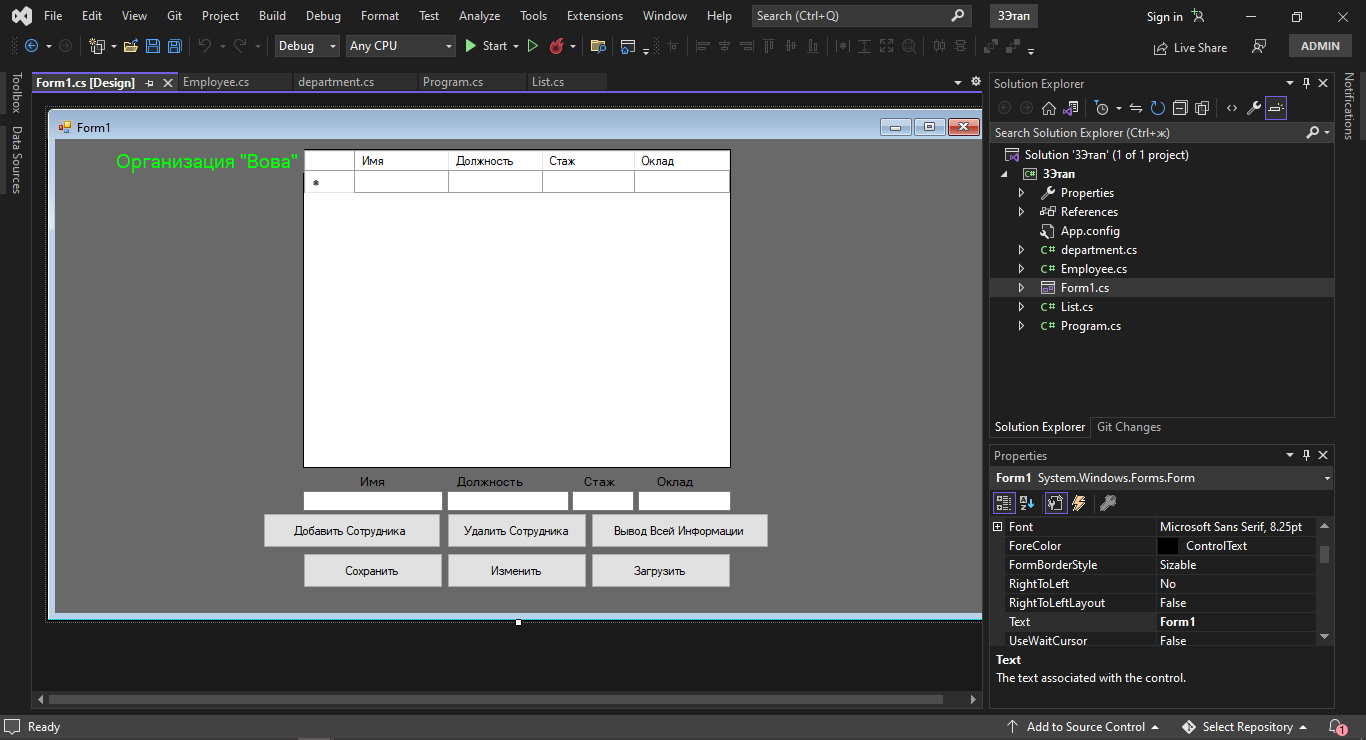


Рис. яд 8 яд Работа яд Visual яд Studio яд 2022

**Список яд литературы**

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

Листинг яд программы

// яд Form1 яд

**using** яд System**;**

**using** яд System**.**Collections**.**Generic**;**

**using** яд System**.**ComponentModel**;**

**using** яд System**.**Data**;**

**using** яд System**.**Drawing**;**

**using** яд System**.**Linq**;**

**using** яд System**.**Text**;**

**using** яд System**.**Threading**.**Tasks**;**

**using** яд System**.**Windows**.**Forms**;**

**using** яд System**.**IO**;**

**namespace** яд Этап3

**{**

**public** яд **partial** яд class яд Form1 яд **:** яд Form

**{**

яд department яд Dep яд **=** яд **new** яд department**(**"Вова"**);**

яд **public** яд Form1**()**

яд **{**

яд InitializeComponent**();**

яд **}**

яд **private** яд void яд Form1\_Load**(object** яд sender**,** яд EventArgs яд e**)**

яд **{**

яд **}**

яд **private** яд void яд AddEmpl\_Click**(object** яд sender**,** яд EventArgs яд e**)**

яд **{**

яд int яд exp яд **=** яд Convert**.**ToInt32**(**textBox3**.**Text**);**

яд int яд salary яд **=** яд Convert**.**ToInt32**(**textBox4**.**Text**);**

яд Dep**.**AddEmployee**(**textBox1**.**Text**,** яд textBox2**.**Text**,**exp**,**salary**);**

яд dataGridView1**.**Rows**.**Add**(**textBox1**.**Text**,** яд textBox2**.**Text**,** яд exp**,** яд salary**);**

яд **}**

яд **private** яд void яд DeleteEmpl\_Click**(object** яд sender**,** яд EventArgs яд e**)**

яд **{** яд яд яд

яд string яд chose яд **=** яд dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

яд Dep**.**DeleteEmployee**(**chose**);**

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

яд dataGridView1**.**Rows**.**RemoveAt**(**index**);**

яд **}**

яд **private** яд void яд VivoDinf\_Click**(object** яд sender**,** яд EventArgs яд e**)**

яд **{**

яд MessageBox**.**Show**(**Dep**.**getInfo**());**

яд **}**

яд **private** яд void яд Change\_Click**(object** яд sender**,** яд EventArgs яд e**)**

яд **{**

яд string яд nameEmployee яд **=** яд dataGridView1**.**CurrentCell**.**Value**.**ToString**();**

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

яд Dep**.**FindEmployee**(**nameEmployee**).**Fam яд **=** яд textBox1**.**Text**;**

яд dataGridView1**.**Rows**[**index**].**Cells**[**0**].**Value яд **=** яд textBox1**.**Text**;**

яд Dep**.**FindEmployee**(**textBox1**.**Text**).**Work яд **=** яд textBox2**.**Text**;**

яд dataGridView1**.**Rows**[**index**].**Cells**[**1**].**Value яд **=** яд textBox2**.**Text**;**

яд Dep**.**FindEmployee**(**textBox1**.**Text**).**Exp яд **=** яд Convert**.**ToInt32**(**textBox3**.**Text**);**

яд dataGridView1**.**Rows**[**index**].**Cells**[**2**].**Value яд **=** яд Convert**.**ToInt32**(**textBox3**.**Text**);**

яд Dep**.**FindEmployee**(**textBox1**.**Text**).**Salary яд **=** яд Convert**.**ToInt32**(**textBox4**.**Text**);**

яд dataGridView1**.**Rows**[**index**].**Cells**[**3**].**Value яд **=** яд textBox4**.**Text**;**

яд **}**

яд **private** яд void яд SavEinFile\_Click**(object** яд sender**,** яд EventArgs яд e**)**

яд **{**

яд string яд path**;**

яд string яд fileText яд **=** яд "Файл"**;**

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

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

яд **{**

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

яд **{**

яд яд path яд **=** яд saveFile**.**FileName**;**

яд // яд fileText яд += яд Dep.getNameDep() яд + яд "\n";

яд яд fileText яд **+=** яд Dep**.**getLoadInfo**();**

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

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

яд яд **{**

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

яд яд stream**.**Close**();**

яд яд file**.**Close**();**

яд яд **}**

яд **}**

яд **}**

яд **}**

яд **private** яд void яд LoaDinGrid\_Click**(object** яд sender**,** яд EventArgs яд e**)**

яд **{**

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

яд int яд Count**,** яд LCount**,** яд gName**;**

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

яд **{**

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

яд **{**

яд яд path яд **=** яд openFile**.**FileName**;**

яд **}**

яд **}**

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

яд **{**

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

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

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

яд **{**

яд яд **for** яд **(**int яд i яд **=** яд 0**;** яд i яд **<=** яд Dep**.**getCount**();** яд i**++)**

яд яд **{**

яд яд Dep**.**DeleteAll**();**

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

яд яд **}**

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

яд яд **for** яд **(**int яд i яд **=** яд 1**;** яд i яд **<=** яд Count**;)**//первая яд колонка

яд яд **{**

яд яд string яд name яд **=** яд stream3**.**ReadLine**();**

яд яд int яд Loadsalary яд **=** яд int яд **.**Parse**(**stream3**.**ReadLine**());**

яд яд int яд LoadExp яд **=** яд int**.**Parse**(**stream3**.**ReadLine**());**

яд яд string яд Loadwork яд **=** яд stream3**.**ReadLine**();**

яд яд Dep**.**AddEmployee**(**name**,**Loadwork**,**LoadExp**,**Loadsalary**);**

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

яд яд dataGridView1**.**Rows**[**i яд **-** яд 1**].**Cells**[**0**].**Value яд **=** яд name**;**

яд яд dataGridView1**.**Rows**[**i яд **-** яд 1**].**Cells**[**1**].**Value яд **=** яд Loadwork**;**

яд яд dataGridView1**.**Rows**[**i яд **-** яд 1**].**Cells**[**2**].**Value яд **=** яд LoadExp**;**

яд яд dataGridView1**.**Rows**[**i яд **-** яд 1**].**Cells**[**3**].**Value яд **=** яд Loadsalary**;**

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

яд яд i**++;**

яд яд **else**

яд яд **break;** яд

яд яд **}**

яд **}**

яд **else** яд **{** яд MessageBox**.**Show**(**"Данный яд файл яд не яд подходит яд для яд работы яд с яд программой"**);** яд **}**

яд stream3**.**Close**();**

яд **}**

яд **}**

**}**

**}**

// яд Employee яд

**using** яд System**;**

**using** яд System**.**Collections**.**Generic**;**

**using** яд System**.**Text**;**

**namespace** яд Этап3

**{**

class яд Employee

**{**

яд **private** яд string яд fam**,** яд work**;**//Создание яд необходимых яд пременных яд

яд **private** яд int яд salary**,** яд exp**;**//Создание яд необходимых яд пременных яд

яд **public** яд Employee яд **(**string яд \_Fam**,**string яд \_work**,** яд int яд \_exp**,** яд int яд \_salary**)**

яд **{**

яд **this.**fam яд **=** яд \_Fam**;**

яд **this.**work яд **=** яд \_work**;**

яд **this.**exp яд **=** яд \_exp**;**

яд **this.**salary яд **=** яд \_salary**;**

яд **}**

яд **public** яд string яд Fam

яд **{**

яд **get** яд **{** яд **return** яд fam**;** яд **}**

яд **set** яд **{** яд **this.**fam яд **=** яд **value;** яд **}**

яд **}**

яд **public** яд string яд Work

яд **{**

яд **get** яд **{** яд **return** яд work**;** яд **}**

яд **set** яд **{** яд **this.**work яд **=** яд **value;** яд **}**

яд **}**

яд **public** яд int яд Exp

яд **{**

яд **get** яд **{** яд **return** яд exp**;** яд **}**

яд **set** яд **{** яд **this.**exp яд **=** яд **value;** яд **}**

яд **}**

яд **public** яд int яд Salary

яд **{**

яд **get{** яд **return** яд salary**;** яд **}**

яд **set** яд **{** яд **this.**salary яд **=** яд **value;** яд **}**

яд **}**

яд

**}**

**}**

// яд department яд

**using** яд System**;**

**using** яд System**.**Collections**.**Generic**;**

**using** яд System**.**Text**;**

**namespace** яд Этап3

**{**

class яд department

**{**

яд **private** яд string яд nameDep**;**//название яд департамента

яд **private** яд int яд count**;**//Создание яд необходимых яд переменных яд

яд **private** яд List яд pFirst**,**pHelp**,** яд pHelpInFirst**;**//ссылка яд на яд первого яд сотрудника

яд **public** яд department**(**string яд \_nameDep**)**

яд **{**

яд **this.**nameDep яд **=** яд \_nameDep**;**

яд count яд **=** яд 0**;**

яд pFirst яд **=** яд **null;**

яд pHelp яд **=** яд **null;**

яд pHelpInFirst яд **=** яд **null;**

яд **}**

яд **public** яд string яд getNameDep**()** яд **{** яд **return** яд nameDep**;** яд **}**

яд **public** яд int яд getCount**()**

яд **{**

яд **return** яд count**;**

яд **}**

яд **public** яд void яд AddEmployee**(**string яд \_Fam**,** яд string яд \_work**,** яд int яд \_exp**,** яд int яд \_salary**)**

яд **{**

яд Employee яд newEmp яд **=** яд **new** яд Employee**(** яд \_Fam**,** яд \_work**,** яд \_exp**,** яд \_salary**);**

яд List яд pTemp яд **=** яд **new** яд List**(**newEmp**);**

яд count**++;**

яд **if** яд **(**pFirst яд **==** яд **null)**

яд **{**

яд pFirst яд **=** яд pTemp**;**

яд pFirst**.**Next яд **=** яд **null;**

яд pFirst**.**Pred яд **=** яд **null;**

яд pHelpInFirst яд **=** яд pFirst**;**

яд **}**

яд **else**

яд **{**

яд pHelp яд **=** яд pTemp**;**

яд pHelp**.**Pred яд **=** яд pHelpInFirst**;**

яд pHelpInFirst**.**Next яд **=** яд pHelp**;**

яд pHelpInFirst яд **=** яд pHelpInFirst**.**Next**;**

яд **}**

яд **}**

яд **public** яд void яд DeleteAll**()**

яд **{**

яд **if** яд **(**count яд **!=** яд 0**)**

яд **{**

яд pFirst яд **=** яд **null;**

яд **}**

яд **}**

яд **public** яд void яд DeleteEmployee**(**string яд \_Fam**)**

яд **{**

яд pHelp яд **=** яд pFirst**;**

яд **while** яд **(**pHelp яд **!=** яд **null)**

яд **{**

яд List яд pTemp**;**

яд **if** яд **(**pHelp яд **==** яд pFirst яд **&&** яд pHelp**.**info**.**Fam**.**Equals**(**\_Fam**))**

яд **{**

яд яд **if** яд **(**pHelp**.**Next яд **==** яд **null)**

яд яд **{**

яд яд pFirst яд **=** яд **null;**

яд яд **}**

яд яд **else**

яд яд **{**

яд яд pTemp яд **=** яд pFirst**.**Next**;**

яд яд pHelp**.**Next**.**Pred яд **=** яд **null;**

яд яд pHelp**.**Next яд **=** яд **null;**

яд яд pFirst яд **=** яд pTemp**;**

яд яд **}**

яд **}**

яд **else** яд **if(**pHelp**.**info**.**Fam**.**Equals**(**\_Fam**))**

яд **{**

яд яд pHelp**.**Pred**.**Next яд **=** яд pHelp**.**Next**;**

яд яд pHelp**.**Next**.**Pred яд **=** яд pHelp**.**Pred**;**

яд **}**

яд pHelp яд **=** яд pHelp**.**Next**;**

яд **}**

яд **}**

яд **public** яд string яд GetSumma**()**

яд **{**

яд string яд inform яд **=** яд "\nСумма яд запрплат яд у яд всех яд сотрудников яд в яд отделе яд " яд **+** яд nameDep яд **+** яд " яд равна: яд "**;**

яд int яд summa яд **=** яд 0**;**

яд pHelp яд **=** яд pFirst**;**

яд **while(**pHelp яд **!=** яд **null)**

яд **{**

яд summa яд **+=** яд pHelp**.**info**.**Salary**;**

яд pHelp яд **=** яд pHelp**.**Next**;**

яд **}**

яд inform яд **+=** яд summa яд **+** яд "\n\n"**;**

яд **return** яд inform**;**

яд **}**

яд **public** яд int яд sum**()**

яд **{**

яд int яд summa яд **=** яд 0**;**

яд pHelp яд **=** яд pFirst**;**

яд **while** яд **(**pHelp яд **!=** яд **null)**

яд **{**

яд summa яд **+=** яд pHelp**.**info**.**Salary**;**

яд pHelp яд **=** яд pHelp**.**Next**;**

яд **}**

яд **return** яд summa**;**

яд **}**

яд **public** яд string яд getInfo**()** яд

яд **{**

яд string яд info яд **=** яд "------: яд " яд **+** яд nameDep яд **+** яд "------"**;**

яд pHelp яд **=** яд pFirst**;**

яд **while(**pHelp яд **!=** яд **null)**

яд **{**

яд info яд **+=** яд "\n\nСотрудник: яд " яд **+** яд pHelp**.**info**.**Fam**;**

яд info яд **+=** яд "\nРазмер яд ЗП яд - яд " яд **+** яд pHelp**.**info**.**Salary**;**

яд info яд **+=** яд "\nСтаж яд - яд " яд **+** яд pHelp**.**info**.**Exp яд **+** яд " яд лет"**;**

яд info яд **+=** яд "\nДолжность яд - яд " яд **+** яд pHelp**.**info**.**Work**;**

яд pHelp яд **=** яд pHelp**.**Next**;**

яд **}**

яд info яд **+=** яд GetSumma**();**

яд **return** яд info**;** яд

яд **}**

яд **public** яд Employee яд FindEmployee**(**string яд inameemployee**)**

яд **{** яд pHelp яд **=** яд pFirst**;**

яд **while(**pHelp**!=** яд **null)**

яд **{**

яд **if** яд **(**pHelp**.**info**.**Fam**.**Equals**(**inameemployee**))**

яд **{**

яд яд **return** яд pHelp**.**info**;**

яд **}**

яд **else** яд pHelp яд **=** яд pHelp**.**Next**;**

яд **}**

яд **return** яд **null;**

яд **}**

яд **public** яд string яд getLoadInfo**()**

яд **{**

яд string яд infoLoad яд **=** яд ""**;**

яд int яд countLoad яд **=** яд count**;**

яд pHelp яд **=** яд pFirst**;**

яд infoLoad яд **+=** яд "\n" яд **+** яд countLoad**;**

яд

яд **while** яд **(**pHelp яд **!=** яд **null)**

яд **{**

яд infoLoad яд **+=** яд "\n" яд **+** яд pHelp**.**info**.**Fam яд **+** яд "\n"**;**

яд infoLoad яд **+=** яд pHelp**.**info**.**Salary яд **+** яд "\n"**;**

яд infoLoad яд **+=** яд pHelp**.**info**.**Exp яд **+** яд "\n"**;**

яд infoLoad яд **+=** яд pHelp**.**info**.**Work**;**

яд pHelp яд **=** яд pHelp**.**Next**;**

яд **}**

яд **return** яд яд infoLoad**;** яд

яд яд

яд **}**

**}**

**}**

// яд List

**using** яд System**;**

**using** яд System**.**Collections**.**Generic**;**

**using** яд System**.**Text**;**

**namespace** яд Этап3

**{**

class яд List яд // яд адресный яд список

**{**

яд **public** яд Employee яд info**;**

яд **private** яд List яд next**;**

яд **private** яд List яд pred**;**

яд **public** яд List**(**Employee яд newEmployee**)**

яд **{**

яд info яд **=** яд newEmployee**;**

яд next яд **=** яд **null;**

яд pred яд **=** яд **null;**

яд **}**

яд **public** яд List яд Next

яд **{**

яд **get** яд **{** яд **return** яд next**;** яд **}**

яд **set** яд **{** яд **this.**next яд **=** яд **value;** яд **}**

яд **}**

яд **public** яд List яд Pred

яд **{**

яд **get** яд **{** яд **return** яд pred**;** яд **}**

яд **set** яд **{** яд **this.**pred яд **=** яд **value;** яд **}**

яд **}**

**}**

**}**