

Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ  
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)  
Кафедра комплексной информационной безопасности электронно-  
вычислительных систем (КИБЭВС)

ПРОГРАММА ДЛЯ УЧЕТА УСТАНОВЛЕННОГО ПО  
Курсовая работа по дисциплине «Программно-аппаратные средства  
обеспечения информационной безопасности»

Студент гр. 724

\_\_\_\_\_ А.А. Крупина

\_\_\_\_\_

Руководитель

Аспирант кафедры БИС, к.т.н.

\_\_\_\_\_ И.А. Рахманенко

\_\_\_\_\_

## Реферат

Курсовая работа \_\_ страниц, \_\_ таблицы, \_\_ рисунков, \_\_ источников, \_\_ приложение.

ПРОГРАММА ДЛЯ УЧЕТА УСТАНОВЛЕННОГО ПО.

Ключевые слова: программа, установленное ПО, реестр.

Объектом исследования является процесс получения данных об установленном ПО из реестра.

Цель работы – создание программы, которая должна выводить информацию (предусмотренную техническим заданием) об установленном ПО. Для разработки пояснительной записки использовались документы:

- ОС ТУСУР 01-2013 [1];
- ГОСТ 34.602-89 Техническое задание на создание автоматизированной системы.

Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования

ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ  
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)

Кафедра комплексной информационной безопасности электронно-  
вычислительных систем (КИБЭВС)

УТВЕРЖДАЮ

Аспирант кафедры БИС, к.т.н.

Рахманенко И.А.

\_\_\_\_\_

Дата: \_\_\_\_\_

УТВЕРЖДАЮ

Студент группы 724

Крупина А.А.

\_\_\_\_\_

Дата: \_\_\_\_\_

ПРОГРАММА ДЛЯ УЧЕТА УСТАНОВЛЕННОГО ПО

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

На \_\_\_\_\_ листах

Действует с

СОГЛАСОВАНО

Аспирант кафедры БИС, к.т.н.

И.А. Рахманенко

\_\_\_\_\_

Дата: \_\_\_\_\_

## 1 Общие сведения

### 1.1 Полное наименование системы и ее условное обозначение

Программа для учета установленного программного обеспечения (далее - ПО).

Условное обозначение – программа.

### 1.2 Шифр темы или шифр (номер) договора

Отсутствуют.

### 1.3 Наименование предприятий (объединений) разработчика и заказчика (пользователя) системы и их реквизит

Заказчик – кафедра комплексной информационной безопасности электронно-вычислительных систем (КИБЭВС).

Адрес: г. Томск, ул. Красноармейская 146 УЛК, 407.

Разработчик – Крупина А.А.

Е – mail: [krupina.aa@mail.ru](mailto:krupina.aa@mail.ru).

### 1.4 Перечень документов, на основании которых создается система, кем и когда утверждены эти документы

Перечень документов:

– методические указания для выполнения курсовой работы по дисциплине "Программно-аппаратные средства обеспечения информационной безопасности" / Рахманенко И.А. - 2017. - 5 с.[Электронный ресурс] - Режим доступа: <https://goo.gl/MRZi23>, дата обращения: 31.05.2018.

– рабочая программа дисциплины "Программно-аппаратные средства обеспечения информационной безопасности" / Троян П.Е. – 2018. – 22 с.[Электронный ресурс] - Режим доступа: [https://edu.tusur.ru/work\\_programs/28695](https://edu.tusur.ru/work_programs/28695), дата обращения: 31.05.2018.

#### 1.5 Плановые сроки начала и окончания работы по созданию системы

С 28.02.2018 по 8.06.2018.

#### 1.6 Сведения об источниках и порядке финансирования работ

Сведения об источниках и порядке финансирования работ отсутствуют.

1.7 Порядок оформления и предъявления заказчику результатов работ по созданию системы (ее частей), по изготовлению и наладке отдельных средств (технических, программных, информационных) и программно-технических (программно-методических) комплексов системы

Работы по созданию программы сдаются разработчиком в конце срока окончания работы, указанном в пункте 1.5, где сдаются соответствующие отчетные документы: техническое задание, пояснительная записка (руководство пользователя в составе ПЗ). Более подробные сроки выполнения каждого из этапов указаны в таблице 1.1.

Таблица 1.1 - Порядок оформления и предъявления заказчику результатов работ по созданию программы

Вид работ	Сроки	Выполняют
Техническое задание	С 28.02.18 до 31.03.18	Разработчик
Проектирование	С 1.04.18 до 20.04.18	Разработчик
Реализация	С 21.04.18 до 24.05.18	Разработчик
Пояснительная записка	С 25.05.18 до 3.06.18	Разработчик
Защита работы	С 4.06.18 до 8.06.18	Разработчик

## 2 Назначение и цели создания (развития) системы

### 2.1 Назначение системы

Программа предназначена для считывания из нескольких заранее заданных веток реестра наименований программ и дат их установки. Предполагается использование в личных или учебных целях.

### 2.2 Цели создания системы

Основными целями создания программы являются:

- реализации курсовой работы;
- определении веток реестра для считывания определенных ключей;
- реализация программы с графическим интерфейсом.

### 3 Характеристика объектов автоматизации

#### 3.1 Краткие сведения об объекте автоматизации или ссылки на документы, содержащие такую информацию

Объектами автоматизации являются:

- процесс считывания данных из реестра;
- процесс идентификации пользователя, который выполнил вход в операционную систему Windows.

Процесс идентификации пользователя должен включать в себя:

- идентификацию пользователя;
- вывод имени и автаркии пользователя (если имеется, иначе должна устанавливаться заданная картинка по умолчанию).

Считывание из реестра должно включать в себя:

- определение имеющихся основных заданных веток реестра для создания списка установленных приложений;
- вывод информации реестра: наименования программы, даты установки;
- сортировку по наименованию приложения и по дате;
- фильтрацию приложений по наименованию.

#### 3.2 Сведения об условиях эксплуатации объекта автоматизации и характеристиках окружающей среды.

Сведения:

- температура и влажность, условия окружающей среды должны быть благоприятными для программы, запуска программы и работы компьютера, на котором установлена программа, иначе стабильная работа не гарантируется;



- приложение использовать в соответствии с его прямым назначением;
- необходимый набор минимальных характеристик для работы программы расписан в пункте 4.3.2.

## 4 Требования к системе

### 4.1 Требования к системе в целом

#### 4.1.1 Требования к структуре и функционированию системы

4.1.1.1 Перечень подсистем, их назначение и основные характеристики, требования к числу уровней иерархии и степени централизации системы

Программа должна быть децентрализованной.

Список подсистем программы:

- идентификация пользователя;
- вывод по имени пользователя его аватара;
- определение имеющихся основных заданных веток реестра для создания списка установленных приложений;
- вывод информации из реестра: наименования программы, даты установки;
- сортировка по наименованию приложения и по дате;
- фильтрация приложений по наименованию.

4.1.1.2 Требования к способам и средствам связи для информационного обмена между компонентами системы

Чтение данных из реестра программой могут производиться на основании прав доступа пользователя, вошедшего в систему, к реестру.

4.1.1.3 Требования к характеристикам взаимосвязей создаваемой системы со смежными системами, требования к ее совместимости, в том числе указания о способах обмена информацией (автоматически, пересылкой документов, по телефону и т. п.)

Пользователи программы должны иметь права чтения некоторых веток реестра, определенных на этапе проектирования.

#### 4.1.1.4 Требования к режимам функционирования системы

Может устанавливаться один из следующих режимов функционирования системы:

- нормальный режим функционирования (непрерывный режим работы программы, исправно функционирует системное, базовое и прикладное программное обеспечение системы);
- аварийный режим функционирования (отказ одного или нескольких компонент программного и (или) технического обеспечения).

#### 4.1.1.5 Требования по диагностированию системы

Требования отсутствуют.

#### 4.1.1.6 Перспективы развития, модернизации системы

Требования отсутствуют.

#### 4.1.2 Требования к численности и квалификации персонала системы и режиму его работы

Для пользователей программы достаточно минимальное владение компьютером.

#### 4.1.3 Показатели назначения

##### 4.1.3.1 Степень приспособляемости системы к изменению процессов и методов управления, к отклонениям параметров объекта управления

Требования не предъявляются.

##### 4.1.3.2 Допустимые пределы модернизации и развития системы

Требования не предъявляются.

##### 4.1.3.3 Вероятностно-временные характеристики, при которых сохраняется целевое назначение системы

Время считывания данных из реестра определяется техническими возможностями аппаратного обеспечения, на котором функционирует система.

#### 4.1.4 Требования к надежности

Надежность должна обеспечиваться за счет нормально функционирующего системного программного обеспечения, соблюдения правил эксплуатации программы.

Программа должна обеспечивать восстановление своих функций при сбоях в системе электроснабжения аппаратной части, приводящих к

перезагрузке ОС (восстановление программы должно происходить после повторного запуска исполняемого файла программы). При ошибках, связанных с системным программным обеспечением (реестр), данные по данному ПО выводиться не должны (в случае наименования ПО) или дата должна остаться пустой (если наименование есть, а дата установки отсутствует).

#### 4.1.5 Требования безопасности

Система электропитания должна обеспечивать защитное отключение при перегрузках и коротких замыканиях в цепях нагрузки, а также аварийное ручное отключение.

Общие требования пожарной безопасности должны соответствовать нормам на бытовое электрооборудование. В случае возгорания не должно выделяться ядовитых газов и дымов. После снятия электропитания должно быть допустимо применение любых средств пожаротушения.

Условия эксплуатации объекта автоматизации и характеристики окружающей среды определяются в соответствии с Гигиеническими требованиями к персональным электронно-вычислительным машинам и организации работы (Санитарные правила и нормы. СанПиН 2.2.2/2.4.1340-03).

#### 4.1.6 Требования к эргономике и технической эстетике

Программа должна быть реализована с графическим интерфейсом. При возникновении ошибок пользователь должен быть уведомлен в виде сообщения в диалоговом окне.

#### 4.1.7 Требования к транспортабельности для подвижных АС

Требования не предъявляются.

4.1.8 Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы

Программа должна использоваться для выполнения предписанных данным техническим заданием задач. Для нормальной эксплуатации разрабатываемой программы должно быть обеспечено бесперебойное питание

ПЭВМ. При эксплуатации программы должна быть обеспечена соответствующая стандартам хранения носителей и эксплуатации ПЭВМ температура и влажность воздуха.

4.1.9 Требования к защите информации от несанкционированного доступа

Требования не предъявляются.

4.1.10 Требования по сохранности информации при авариях

Требования не предъявляются.

4.1.11 Требования к защите от влияния внешних воздействий

4.1.11.1 Требования к радиоэлектронной защите средств АС

Требования не предъявляются.

4.1.11.2 Требования по стойкости, устойчивости и прочности к внешним воздействиям (среде применения)

Требования не предъявляются.

4.1.12 Требования к патентной чистоте

Требования не предъявляются.

4.1.13 Требования по стандартизации и унификации

Требования не предъявляются.

4.1.14 Дополнительные требования

Требования не предъявляются.

4.2 Требования к функциям (задачам), выполняемым системой

4.2.1 Идентификация пользователя

Функции:

- идентификация пользователя, вошедшего в систему;
- вывод имени пользователя.

4.2.2 Вывод аватара пользователя

Функции:

- вывод аватара пользователя из каталога «C:\Users\имя пользователя\AppData\Roaming\Microsoft\Windows\AccountPictures» с расширением «accountpicture-ms»;
- если данного файла по заданному пути обнаружено не было или доступ к данному каталогу был ограничен, то нужно вывести картинку по умолчанию.

#### 4.2.3 Определение наличия заранее заданных веток реестра

Функции:

- определение наличия заранее заданных веток реестра;
  - при отсутствии нескольких из веток выводить информацию, исходя из оставшихся веток;
  - если отсутствуют все ветки реестра, то ничего выводить не нужно.
- Ветки реестра будут обозначены на этапе проектирования.

#### 4.2.4 Вывод информации из реестра

Функции:

- вывод наименования ПО (в реестре соответствует параметру DisplayName);
- вывод даты установки ПО (в реестре соответствует параметру InstallDate);
- при отсутствии наименования ПО данный раздел не учитывать;
- при отсутствии даты установки воспользоваться вспомогательными параметрами местонахождения иконки ПО (параметр DisplayIcon) или исполняемого файла (параметр InstallLocation) и вывести дату создания файла/ каталога;
- при отсутствии вспомогательных параметров или возникновения ошибки в дате вывести пустую строку для даты.

#### 4.2.5 Сортировка

Функции:

- сортировка по алфавиту для названий ПО;
- сортировка по дате для дат установки.

#### 4.2.6 Фильтрация

Функция – фильтрация установленного ПО по названию ПО.

### 4.3 Требования к видам обеспечения

#### 4.3.1 Математическое обеспечение системы

Требования не предъявляются.

#### 4.3.2 Информационное обеспечение системы

Доступ к данным должен быть предоставлен только авторизованным пользователям с учетом их служебных полномочий, а также с учетом категории запрашиваемой информации. В состав системы должна входить специализированная подсистема резервного копирования и восстановления данных.

#### 4.3.3 Лингвистическое обеспечение системы

Разработка программы должна вестись с использованием языков высокого уровня. Основными языками взаимодействия пользователей и программы должны быть английский и русский языки.

#### 4.3.4 Программное обеспечение системы

Для нормального функционирования программы необходимо наличие установленного Microsoft .NET Framework 4.5.2 или выше. Программа должна поставляться Заказчику на CD-ROM в следующем составе:

- комплект файлов, необходимых для запуска программы;
- установщик Microsoft .NET Framework 4.5.2 (при наличии у пользователя более новых версий – переустанавливать не надо).

Базовой программной платформой должна являться операционная система MS Windows (Windows 7 Service Pack 1 и новее).

#### 4.3.5 Техническое обеспечение системы

Минимальные характеристики:

- процессор с тактовой частотой 1 ГГц или выше;
- ОЗУ объемом 512 МБ или выше;

- 5 ГБ доступного пространства на жестком диске (x64);
- клавиатура;
- мышь;
- монитор.

#### 4.3.6 Требования к организационному обеспечению

Требования не предъявляются.



## 5 Состав и содержание работ по созданию системы

Перечень стадий и этапов работы отображён в таблице 5.1.

Таблица 5.1 - Перечень стадий и этапов работы

Стадии	Этапы работ	Сроки	Итоги выполненной работы
Подготовительный	Выясняется, что необходимо заказчику, определяются основные ограничения на систему	28.02.18 - 31.03.18	Техническое задание
Проектирование	Разработка структуры программы, выполнение модульной декомпозиции системы	1.04.18 - 20.04.18	Описание алгоритма
Реализация	Разработка программы	21.04.18 - 24.05.18	Программный продукт
Рабочая документация	Сдача итоговых документов с рабочей программой	25.05.18 - 3.06.18	Пояснительная записка
Защита работы	Представление курсовой работы	4.06.18 – 8.06.18	Оценка

Техническое задание должно быть согласовано с Заказчиком на предмет соответствия содержательной части техническим требованиям.

## 6 Порядок контроля и приемки системы

На этапе защиты комиссия, назначенная Заказчиком, проверяет соответствие реализованного функционала программы на данных, предоставленных Заказчиком. Сдача происходит в сроки, описанные в пункте 1.5.

## 7 Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие

Для создания условий функционирования программы, при которых гарантируется соответствие требованиям, содержащимся в настоящем техническом задании, и возможность эффективного её использования, комиссия должна предусмотреть:

- наличие прав доступа (чтения) пользователя к реестру Windows 7 Service Pack 1 или новее;
- наличие установленного Microsoft .NET Framework 4.5.2 (или новее);
- соответствие минимальным техническим характеристикам, описанным в пункте 4.3.5.

## 8 Требования к документированию

Документация должна быть представлена в бумажном виде на русском языке, оформленная в соответствии с ОС ТУСУР 01-2013, основные пункты технического задания которой взяты из ГОСТ 34.602-89 «Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы».

Должен быть подготовлен и передан Заказчику комплект документации в составе:

- техническое задание;
- пояснительная записка (состоящая из введения, теоретической части, основной части, заключения, руководства пользователя).

## 9 Источники разработки

Данная система несет учебную направленность и разрабатывается на основе требований учебного плана специальности «Информационная безопасность автоматизированных систем» на кафедре комплексной информационной безопасности электронно-вычислительных систем (КИБЭВС), а также дополнительными источниками являются:

- санитарные правила и нормы. СанПиН 2.2.2/2.4.1340-03;
- ГОСУДАРСТВЕННЫЙ СТАНДАРТ СОЮЗА ССР «ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ. Комплекс стандартов на автоматизированные системы 34.602-89.ТЕХНИЧЕСКОЕ ЗАДАНИЕ НА СОЗДАНИЕ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ» от 01.01.1990г.;
- Методические указания для выполнения курсовой работы по дисциплине "Программно-аппаратные средства обеспечения информационной безопасности" / Рахманенко И.А. - 2017. - 5 с.[Электронный ресурс] - Режим доступа: <https://goo.gl/MRZi23>, дата обращения: 31.05.2018;
- ОС ТУСУР 01-2013/ УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ приказом ректора ТУСУРа от 03 декабря 2013 г. № 14103.

## СОСТАВИЛ

Наименование организации, предприятия	Должность исполнителя	Фамилия, имя, отчество	Подпись	Дата
ТУСУР	Разработчик	Крупина Алиса Александровна		

## СОГЛАСОВАНО

Наименование организации, предприятия	Должность	Фамилия, имя, отчество	Подпись	Дата

## Оглавление

1 Введение.....	24
2 Теоретическая часть.....	25
3 Основная часть .....	27
4 Заключение .....	32
Список используемых источников.....	33
Приложение А (обязательное) .....	34

## 1 Введение

Цель работы – создание программы для учета установленного ПО. Ввиду целей, поставленных в техническом задании, нужно также:

- определить каким образом получать информацию об установленном ПО;
- какую именно информацию нужно выводить;
- реализация сортировки и фильтрации списка установленного ПО;
- реализация программы с графическим интерфейсом.

Область применения исследуемого объекта – для личного использования, для учебных целей.

Экономическая целесообразность – не рассчитывается.



## 2 Теоретическая часть

Было выявлено несколько способов получения информации об установленном ПО:

- из реестра;
- с помощью Windows PowerShell (оболочка командной строки на основе задач, созданная на базе .NET Framework, с языком сценариев);
- с помощью стороннего ПО (CCleaner, например);
- с помощью команды WMIC (консольная утилита для взаимодействия со структурой WMI на локальном или удаленном компьютере).

Т.к. целью курсовой работы является реализация программы учета установленного ПО, то нет смысла использовать стороннее ПО для этого.

WMIC и PowerShell показывают лишь те программы, которые устанавливались с помощью Windows Installer, к тому же требуются дополнительные права на powershell или командную строку. Сведения об установленных программах содержатся в классе Win32\_Product.

В итоге, самым предпочтительным вариантом стал реестр, так как панель управления берет информацию именно из него, к тому же получение информации можно настроить более гибко.

Основной веткой, где должно храниться установленное ПО является HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall.

Но в дополнении к ней было решено добавить еще следующие ветки:

- HKEY\_LOCAL\_MACHINE\Software\Wow6432Node\Microsoft\Windows\CurrentVersion\Uninstall\;
- HKEY\_CURENT\_USER\Software\Microsoft\Windows\CurrentVersion\Uninstall\.

Из данных веток считывались подразделы, у которых присутствовал параметр `DisplayName`, и считывались наименование и дата (в зависимости от наличия параметров, описано в техническом задании 4.2.4).

Программа рассчитана на ОС Windows. В качестве языка программирования был выбран C# — это простой язык для выбора, который содержит много классов и платформ поддержки, составляющих .NET. Для разработки интерфейса приложения имеются два основных средства — Windows Forms (WinForms) и Windows Presentation Foundation (WPF). Для программы был выбран WinForms, как более простой.

## 3 Основная часть

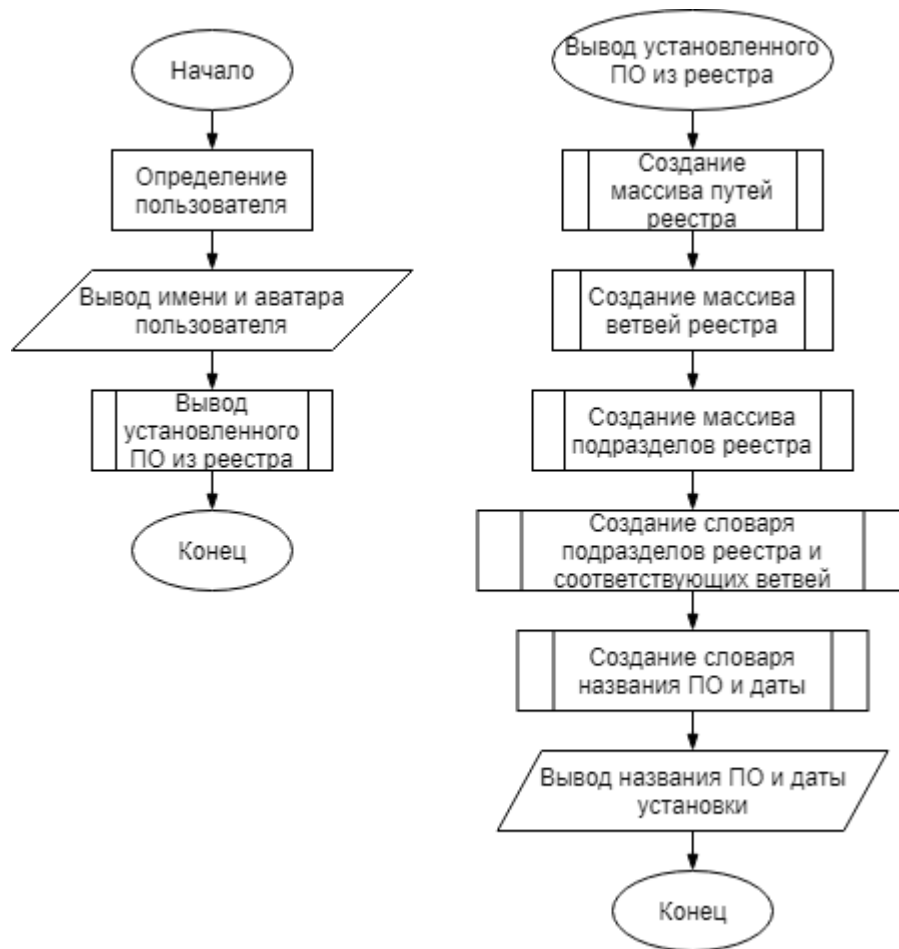


Рисунок 3. 1 - Блок-схема работы программы

На рисунке 3.1 представлена блок-схема работы программы без учета отлавливания событий сортировки и фильтрации. Интерфейс программы представлен на рисунке ниже.

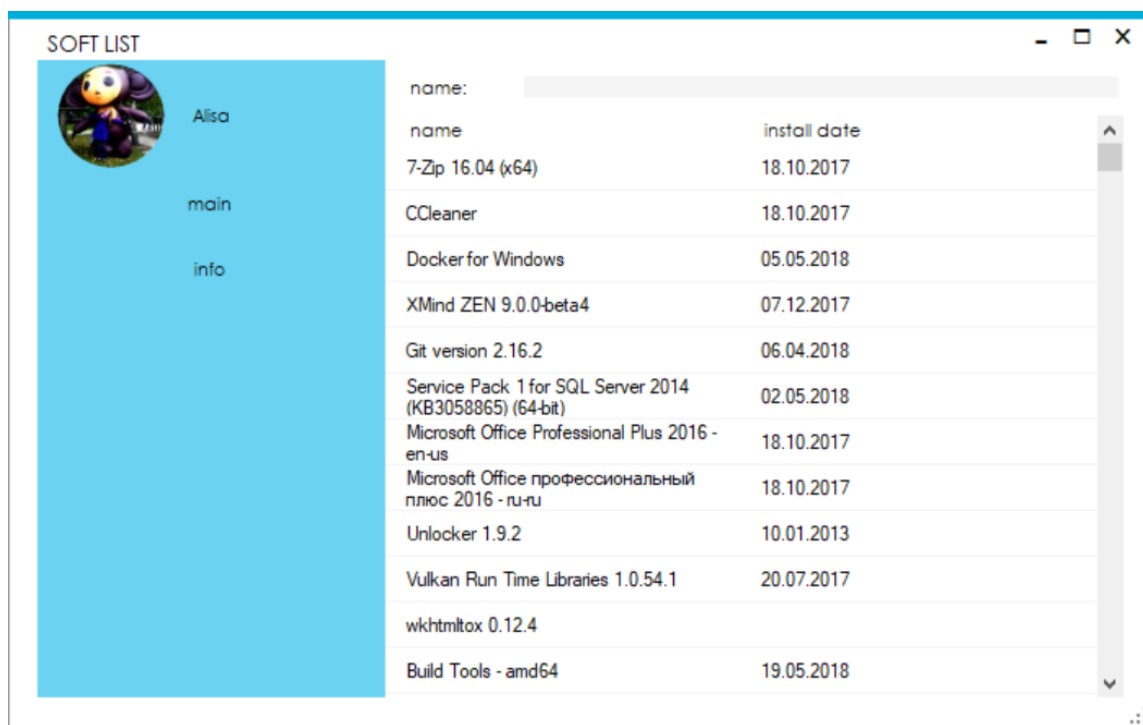


Рисунок 3. 2 - Интерфейс программы

Создание подразделов также включает в себя проверку на существование заданных путей в реестре. На этапе создания словаря подразделов реестра и соответствующих ветвей происходит запись только тех подразделов, у которых имеется параметр `DisplayName`. В функции создания словаря с названием ПО и датой (словарь обеспечивает уникальность названия ПО, поэтому повторения отсутствуют) также находится функция проверки наличия даты (рисунок 3.3).

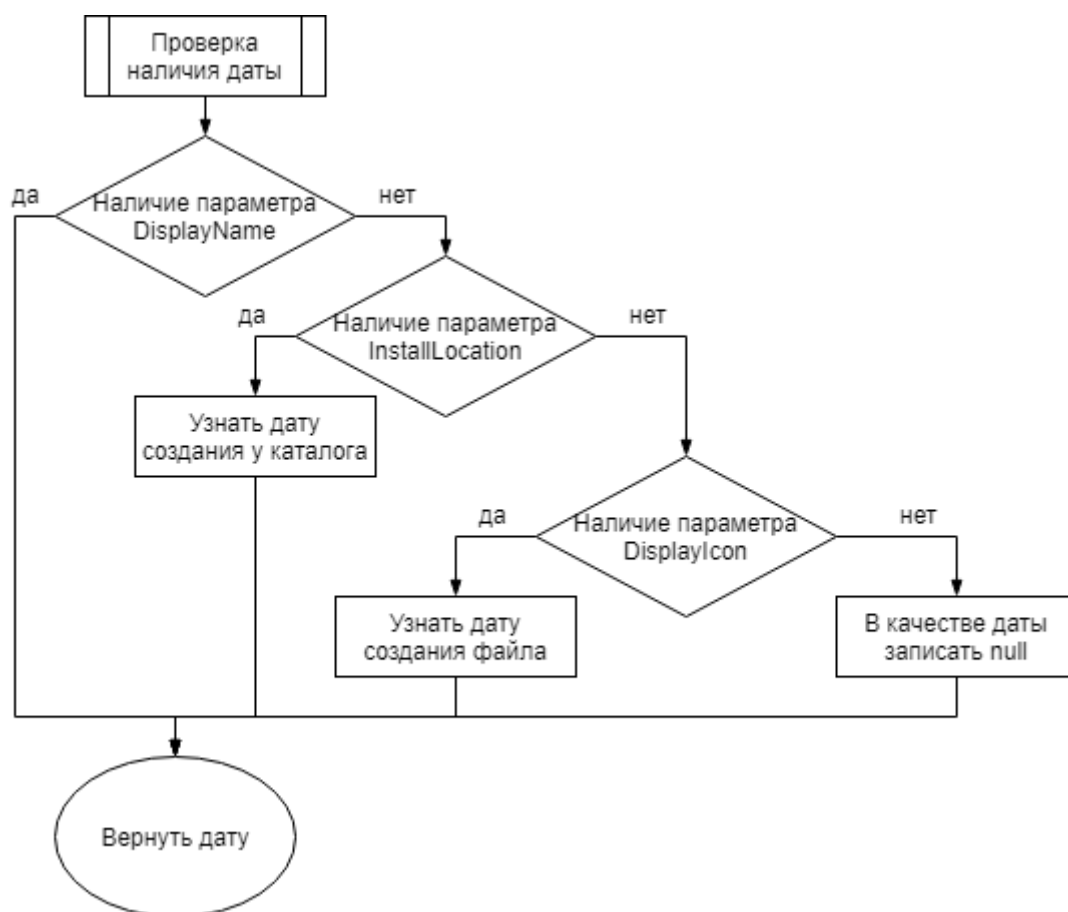


Рисунок 3. 3 - Функция проверки наличия даты

Вывод происходит в таблицу DataGridView, где 1 столбец – название ПО, 2 столбец – дата установки. Сортировка происходит при событии Click на шапку таблицы (сортировка по названию и по дате). Пример сортировки представлен на рисунке 3.4. Фильтрация происходит при возникновении события изменения элемента управления textbox, в котором пользователь может задавать наименование искомого ПО. Основные моменты Код программы представлен на диске.

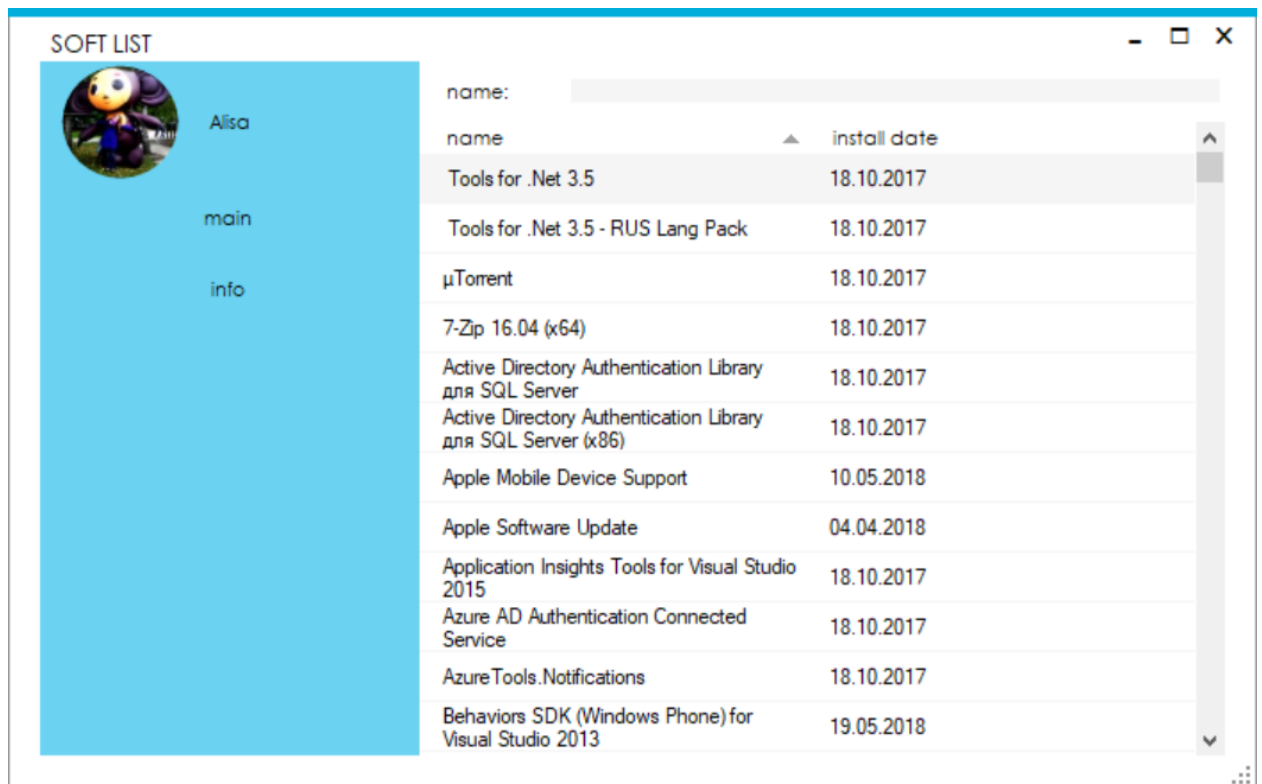


Рисунок 3. 4 - Пример сортировки по наименованию ПО

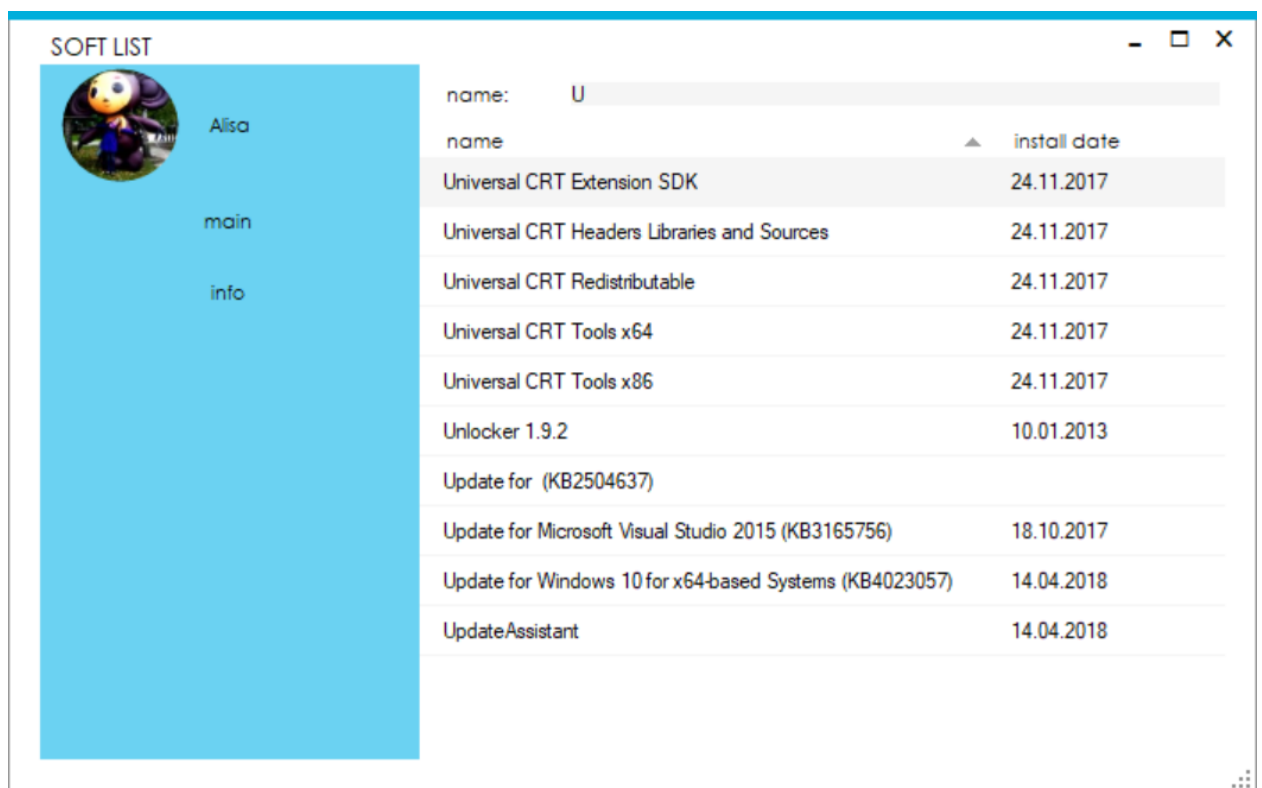


Рисунок 3. 5 - Пример поиска по наименованию ПО

Для нормальной работы приложения требуются права на чтение из реестра, а также установленный Microsoft .NET Framework 4.5.2 или новее. Работоспособность данной программы проверялась на Windows 7, 8.1, 10.

#### 4 Заключение

В ходе выполнения курсовой работы была создана программа для учета установленного ПО, а также:

- определен способ получения информации об установленном ПО;
- определена информация для вывода;
- реализована сортировка, фильтрация установленного ПО;
- реализована программа с графическим интерфейсом.



## Список используемых источников

- 1 ОС ТУСУР 01-2013/ УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ приказом ректора ТУСУРа от 03 декабря 2013 г. № 14103.
- 2 Методические указания для выполнения курсовой работы по дисциплине "Программно-аппаратные средства обеспечения информационной безопасности" / Рахманенко И.А. - 2017. - 5 с.[Электронный ресурс] - Режим доступа: <https://goo.gl/MRZi23>, дата обращения: 31.05.2018.

## Приложение А (обязательное)

### Руководство пользователя

#### 1 Минимальные характеристики

Минимальные характеристики:

- процессор с тактовой частотой 1 ГГц или выше;
- ОЗУ объемом 512 МБ или выше;
- 5 ГБ доступного пространства на жестком диске (x64);
- установленный Microsoft .NET Framework 4.5.2 или выше;
- ОС Windows 7 или новее;

Требуется наличие для нормального взаимодействия с программой:

- клавиатуры;
- мыши;
- монитора.

#### 2 Программа

Программа предназначена для учета установленного ПО. Интерфейс программы представлен на рисунке А.1.

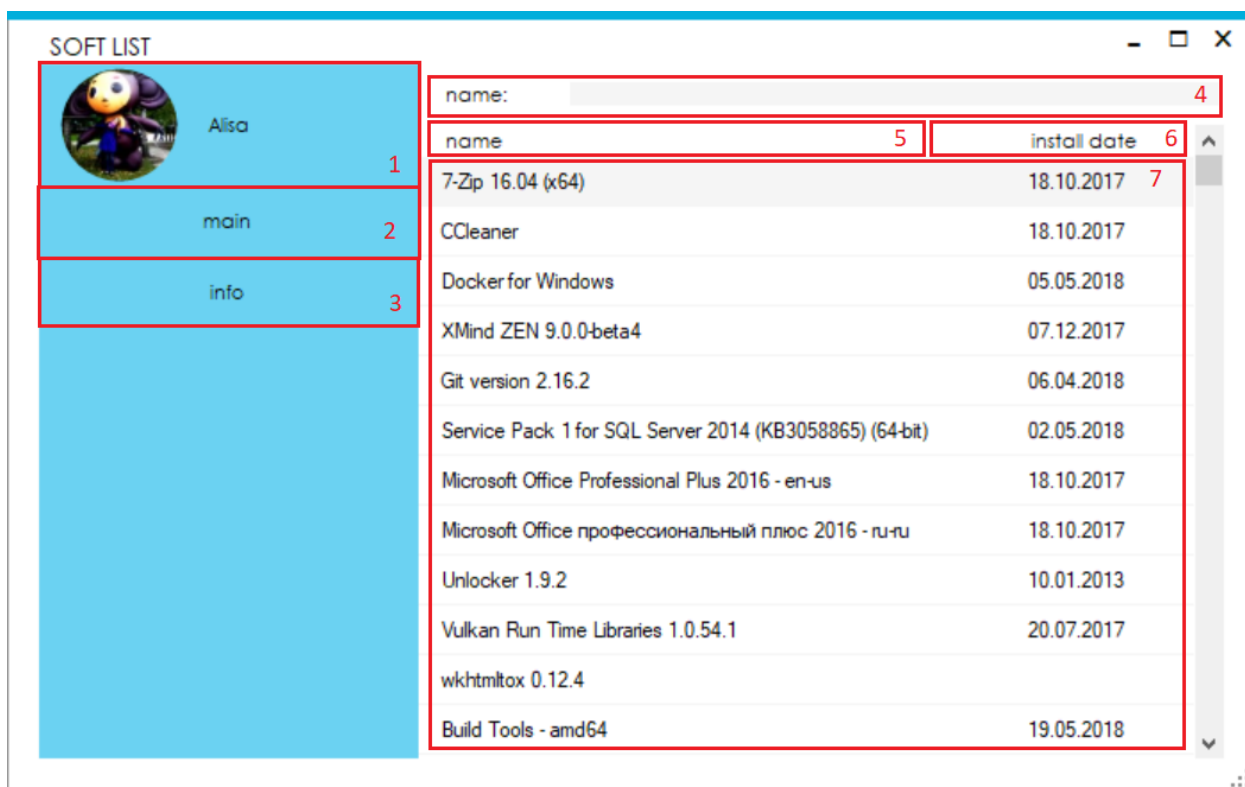


Рисунок А. 1 - Интерфейс программы

Блок 1: отображает имя пользователя и его аватар (если такой есть, если нет, то выводится стандартная картинка, как показано на рисунке А.2).

Блок 2: кнопка, которая перенаправляет на главное окно программы.

Блок 3: кнопка, которая перенаправляет на окно со справочной информацией (рисунок А.3).

Блок 4: область для ввода названия ПО, используется для поиска.

Блок 5: область для сортировки по наименованию ПО.

Блок 6: область для сортировки по дате установки ПО.

Блок 7: область для вывода списка установленного ПО.



Рисунок А. 2 - Учетная запись без аватара

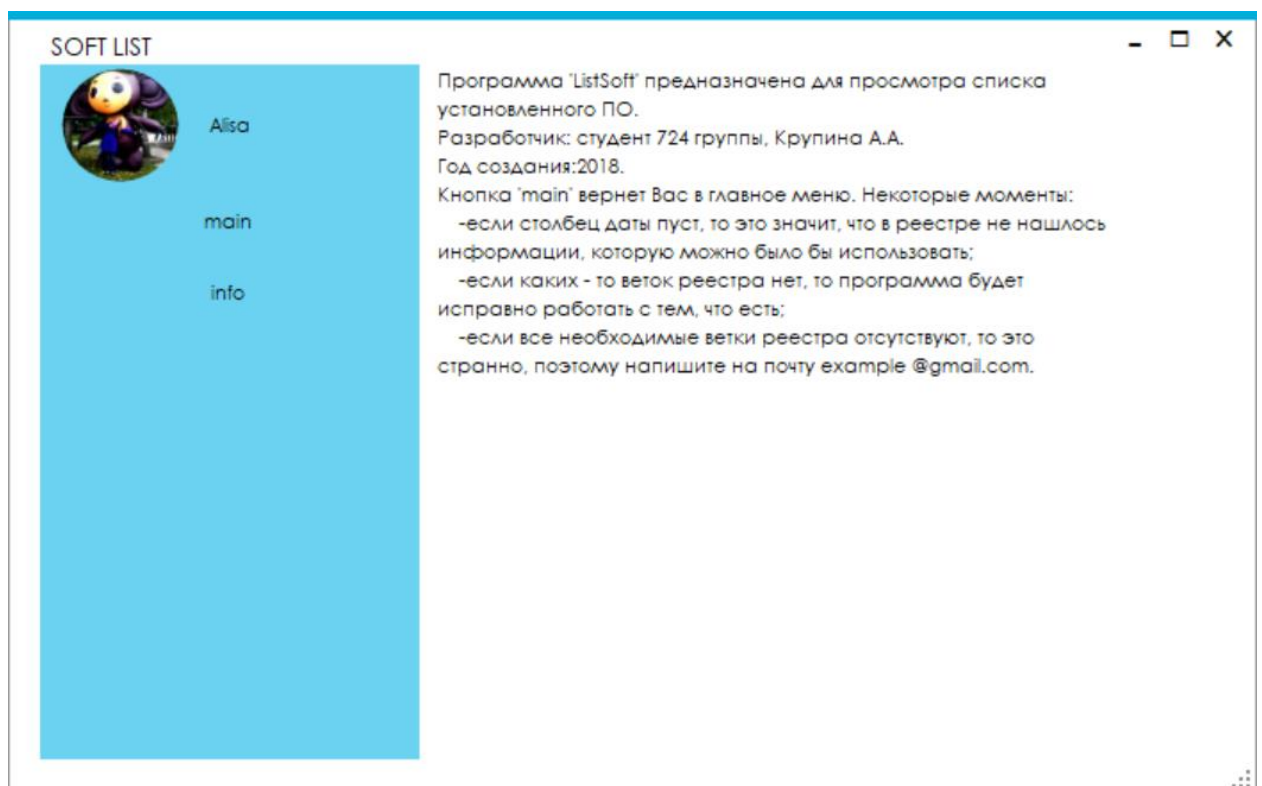


Рисунок А. 3 - Справочное окно

Если в блоке 7 у ПО отсутствует дата, значит ее нет в реестре. Программа не предусматривает запись в реестр. Также программа не предусматривает добавление веток реестра.

### 3 Сообщения об ошибках

Варианты возможных ошибок представлены на рисунках А.4 - А.6.

Ошибка А.4 возникает, когда на компьютере пользователя не установлен Microsoft .NET Framework 4.5.2 или новее.

Ошибка на рисунке А.5 возникает, когда какие-то из заданных путей реестра не обнаружилось в реестре пользователя.

Ошибка на рисунке А.6 возникает, когда ни один путь не обнаружен в реестре пользователя. В следствии этих ошибок программа продолжает работу. При отсутствии всех путей реестра программа выглядит, как показано на рисунке А.7. Если отсутствует один из путей (или несколько), то программа выводит информацию по имеющимся путям.

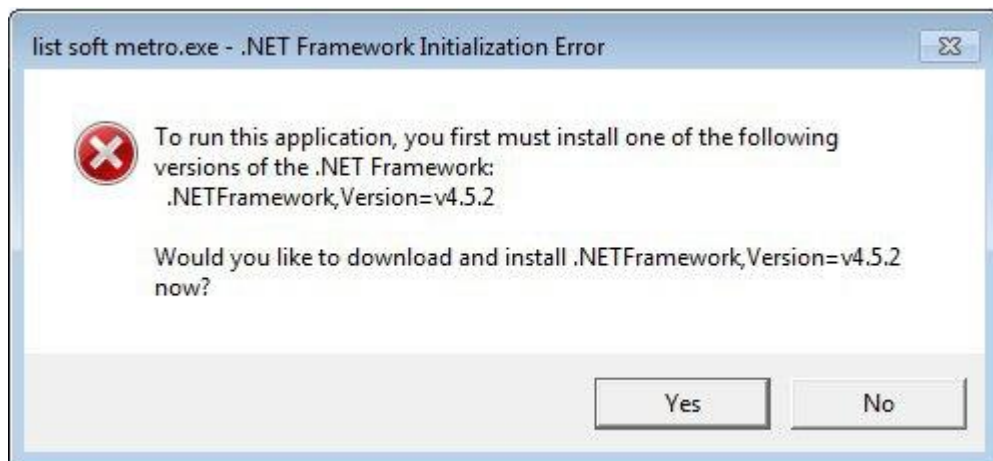


Рисунок А. 4 - Отсутствие Microsoft .NET Framework 4.5.2

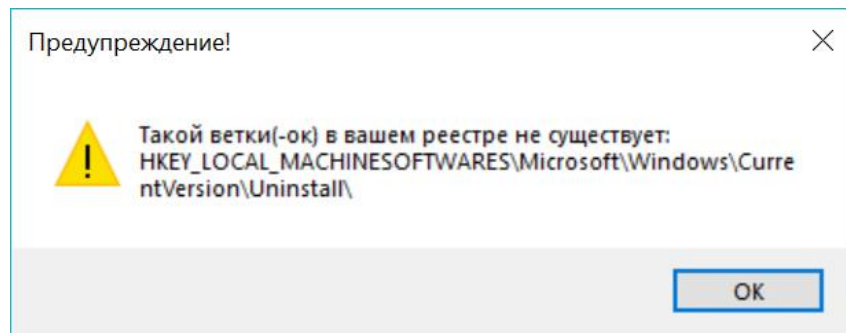


Рисунок А. 5 - Какой-то ветки, заданной в программе, не обнаружено

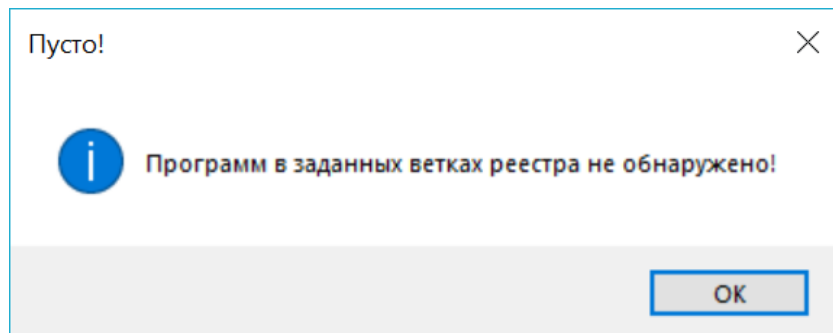


Рисунок А. 6 - Ни одной ветки реестра не обнаружено

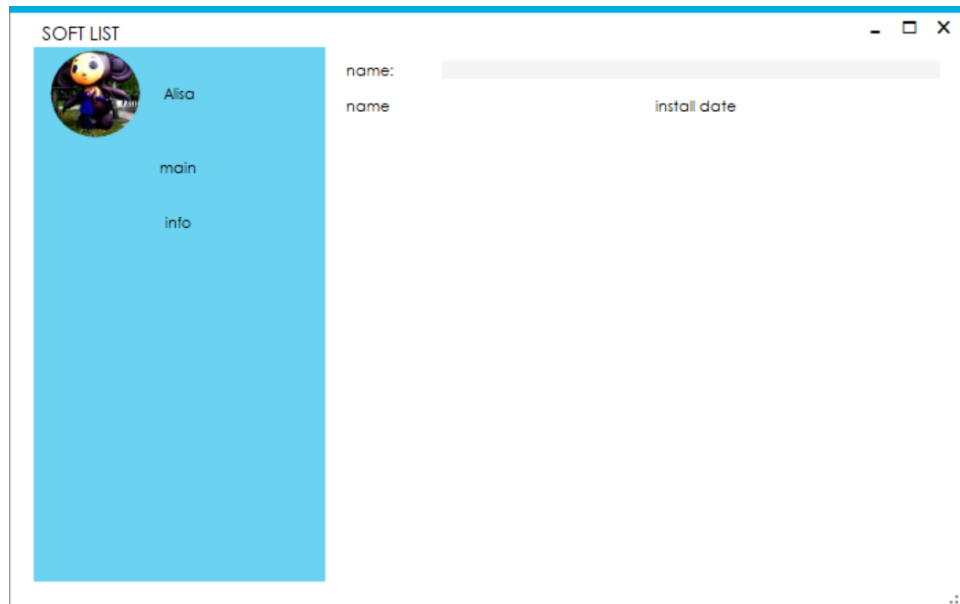


Рисунок А. 7 - Окно программы при отсутствующих ветках