

КГКП «Электротехнический колледж»

Специальность: «06120200 – Системы информационной безопасности»
Квалификация: «4S06120202 – Техник информационной безопасности»

ДИПЛОМНАЯ РАБОТА

По предмету: Обеспечение информационной безопасности предприятия

Тема: Защита папок и файлов от несанкционированного доступа.

Разработал: Серикжан Аян.

Группа: СИБ-413

Руководитель: Павлова О.В.

Оценка: _____

**Семей 2023г.
ЗАДАНИЕ**

Для курсового проектирования по предмету: Обеспечение
информационной безопасности предприятия

Студент(ка): Серикжан Аян.

Курс: 1

Группа: СИБ-413

Тема задания: Защита папок и файлов от несанкционированного доступа.
При выполнении проекта на указанную тему должны быть представлены:

Введение;

Раздел 1 Теоретическая часть;

Раздел 2 Проектная часть;

Охрана труда;

Заключение;

Перечень условных обозначений и сокращений;

Список литературы (прил. В);

Приложения (объем не ограничивается)

Дата выдачи: 17.02.23

Срок окончания: 9.06.23

Преподаватель, руководитель:

курсового проекта: Павлова О.В.

Введение

Глава I. Архиватор и его назначение. **6**

1.1. Архиватор **6**

1.2. Назначение и основные функции программ архиваторов **7**

1.3. Программа архиватора и их цели

9

1.4. Актуальность архиватора на сегодняшний день

11

Глава II. Python.

12

2.1. О языках программирования.

12

2.2. История языка Python.

13

2.3. Язык Python. Цели и актуальность

15

Практическая часть

Охрана труда

Заключение

Список литературы

Приложение

Введение

Появившиеся в начале 80-ых персональные ЭВМ (ПЭВМ или ПК) прочно вошли во все сферы человеческой деятельности. Вместе с ними у эксплуатирующих ПЭВМ организаций и ведомств возникли и многочисленные проблемы. Одна из них — защита информации. Согласно статистическим данным более 80% компаний и агентств несут финансовые убытки из-за нарушения безопасности данных.

Проблема защиты информации представляет собой совокупность тесно связанных проблем в областях права, организации управления, разработки технических средств, программирования и математики. Одна из центральных задач проектирования систем защиты состоит в рациональном распределении имеющихся ресурсов.

Характерная особенность использования ПЭВМ в нашей стране заключается в том, что доступ к ним имеют многие пользователи. В связи с таким "многопользовательским" режимом работы возникает целый набор взаимосвязанных вопросов по защите информации, хранящейся в ПЭВМ.

При создании и использовании ПЭВМ возникает целый ряд взаимосвязанных теоретических и практических проблем. В коммерческих и военных областях одной из основных является проблема защиты информации. Так можно выделить следующие объективные причины, определяющие важность проблемы защиты информации:

- высокие темпы роста парка ПЭВМ, находящихся в эксплуатации;
- широкое применение ПЭВМ в самых различных сферах человеческой деятельности;
- высокая степень концентрации информации в ПЭВМ;
- совершенствование способов доступа пользователей к ресурсам ПЭВМ;
- усложнение вычислительного процесса в ПЭВМ.

Усложнение методов и средств организации машинной обработки информации приводят к тому, что информация становится все более уязвимой. Этому способствуют такие факторы, как постоянно возрастающие объемы обрабатываемых данных, накопление и хранение данных в ограниченных местах, постоянное расширение круга пользователей, имеющих доступ как к ресурсам ПЭВМ, так и к программам и данным, хранящимся в них, усложнение режимов эксплуатации вычислительных систем и т. п.

Глава I. Архиватор и его значение.

1.1 Архиватор.

Архиватор — программа, предназначенная для упаковки без потерь одного и более файлов в единый файл-архив или в серию архивов для удобства переноса и/или хранения данных. Распаковка архивов выполняется с помощью того же архиватора либо посредством сторонних совместимых утилит. Большинство современных архиваторов также выполняет сжатие упаковываемых в архив данных.

Простейшие архиваторы просто последовательно объединяют (упаковывают) содержимое файлов в архив, который, помимо файловых данных, содержит информацию об именах и размерах исходных файлов, чтобы можно было точно восстанавливать файлы в их первоначальном виде. Поэтому большинство архиваторов также сохраняет метаданные файлов, предоставляемые операционной системой, такие как время создания и права доступа. Такую функциональность реализует tar — стандартный архиватор систем типа UNIX. Если необходимо уменьшить размер tar-архива, к нему применяют сжатие без потерь программами gzip, bzip2 и т. д. Большинство современных прикладных архиваторов использует сжатие при работе с настройками по умолчанию.

Многие архиваторы позволяют указывать дополнительные параметры, наиболее важные из которых влияют на степень и скорость сжатия. Эти характеристики — обратно зависимые величины. То есть, чем выше скорость упаковки, тем меньше степень сжатия, и наоборот. Что касается скорости распаковки, то в большинстве современных архиваторов применяются так называемые асимметричные алгоритмы сжатия, при которых скорость (и степень) упаковки практически не влияет на скорость распаковки, которая обычно гораздо выше.

Большинство архиваторов имеет функцию проверки целостности хранящихся в архиве данных. Для этого в архив при добавлении туда файлов вносится информация об их контрольных суммах. При распаковке (или тестировании) архива обязательно вычисляется контрольная сумма каждого извлекаемого файла, и, если она не совпадает с суммой, хранящейся в архиве, то выводится сообщение об ошибке. Таким образом, архиваторы предоставляют очень важную возможность, о которой многие даже не задумываются: гарантию целостности данных. Кроме того, некоторые архиваторы (например, RAR) имеют функции защиты архивов от физических повреждений или даже полной утери отдельных томов многотомных архивов,

благодаря чему архив можно рассматривать не только как средство для хранения данных, но и для их восстановления в исходном виде в случае повреждений.

Некоторые архиваторы позволяют создавать так называемые многотомные архивы, то есть архивы, состоящие из нескольких частей указанного или разного размера. Такие архивы удобно применять для переноса больших объёмов данных на носителях меньшего размера (например, на флэшках или оптических дисках) и обмена данными через Интернет, когда вместо одного огромного архива практичнее передать несколько файлов меньшего размера. В разных архиваторах многотомность реализована по-разному. Например, в форматах ZIP и 7Z тома — это, по сути, просто разделённый на несколько частей исходный архив, что накладывает определённые ограничения на их использование, тогда как тома многотомных архивов RAR представляют собой практически полноценные архивы.

У ряда архиваторов имеется дополнительная функция создания самораспаковывающихся (SFX) архивов. Такие архивы представляют собой исполняемые файлы, для распаковки которых не требуются никакие другие программы — нужно просто запустить SFX-архив, и он сам извлечёт все содержащиеся в нём данные. Это удобно, когда нужно передать архив кому-то ещё, но нет уверенности, что у него окажется соответствующий архиватор. В действительности SFX-архив — это обычный архив, к которому прикреплён исполняемый модуль распаковки, поэтому SFX-архивы можно обрабатывать внешним архиватором как обычные архивы (например, из-за опасения, что исполняемый модуль может быть заражён вирусом). SFX-архивы, создаваемые некоторыми архиваторами, могут быть многотомными, в этом случае первый том имеет исполняемый формат файла, а все последующие — стандартный для ТОМОВ.

1.2 Назначение и основные функции программ архиваторов.

В зависимости от того, в каком объекте размещены данные, подвергаемые сжатию, различают:

- уплотнение (архивацию) файлов;
- уплотнение (архивацию) папок;
- уплотнение дисков.

Если при сжатии данных происходит изменение их содержания, метод сжатия необратим и при восстановлении данных из сжатого файла не происходит полного восстановления исходной последовательности. Такие методы называют также методами сжатия с регулируемой потерей информации. Они применимы только для тех типов данных, для которых формальная утрата части содержания не приводит к значительному снижению

потребительских свойств. В первую очередь, это относится к мультимедийным данным: видеорядам, музыкальным записям, звукозаписям и рисункам. Методы сжатия с потерей информации обычно обеспечивают гораздо более высокую степень сжатия, чем обратимые методы, но их нельзя применять к текстовым документам, базам данных и, тем более, к программному коду. Характерными форматами сжатия с потерей информации являются:

- .JPG для графических данных;
- .MPG для видеоданных;
- .MP3 для звуковых данных.

Если при сжатии данных происходит только изменение их структуры, то метод сжатия обратим. Из результирующего кода можно восстановить исходный массив путем применения обратного метода. Обратимые методы применяют для сжатия любых типов данных. Характерными форматами сжатия без потери информации являются:

- .GIF, TIF, PCX и многие другие для графических данных;
- .AVI для видеоданных;
- .ZIP, .ARJ, .RAR, .LZH, .LH, .CAB и многие другие для любых типов данных.

«Классическими» форматами сжатия данных, широко используемыми в повседневной работе с компьютером, являются форматы .ZIP и .ARJ. В последнее время к ним добавился популярный формат .RAR. К базовым функциям, которые выполняют большинство современных диспетчеров архивов, относятся:

- извлечение файлов из архивов;
- создание новых архивов;
- добавление файлов в имеющийся архив;
- создание самораспаковывающихся архивов;
- создание распределенных архивов на носителях малой емкости;
- тестирование целостности структуры архивов;
- полное или частичное восстановление поврежденных архивов;
- защита архивов от просмотра и несанкционированной модификации.

Самораспаковывающиеся архивы

Самораспаковывающийся архив готовится на базе обычного архива путем присоединения к нему небольшого программного модуля. Сам архив получает расширение имени .EXE, характерное для исполнимых файлов. Распределенные архивы. Некоторые диспетчеры (например, WinZip) выполняют разбиение сразу на гибкие диски, а некоторые (например, WinRAR и WinArj) позволяют выполнить предварительное разбиение архива на фрагменты заданного размера на жестком диске. Впоследствии их можно перенести на внешние носители путем копирования.

При создании распределенных архивов диспетчер WinZip обладает неприятной особенностью: каждый том несет файлы с одинаковыми именами. В результате этого нет возможности установить номера томов, хранящихся на каждом из гибких дисков, по названию файла. Диспетчеры архивов WinArj и WinRAR маркируют все файлы распределенного архива разными именами и потому не создают подобных проблем.

Защита архивов. В большинстве случаев защиту архивов выполняют с помощью пароля, который запрашивается при попытке просмотреть, распаковать или изменить архив.

К дополнительным функциям диспетчеров архивов относятся сервисные функции, делающие работу более удобной. Они часто реализуются внешним подключением дополнительных служебных программ и обеспечивают:

- просмотр файлов различных форматов без извлечения их из архива;
- поиск файлов и данных внутри архивов;
- установку программ из архивов без предварительной распаковки;
- проверку отсутствия компьютерных вирусов в архиве до его распаковки;
- криптографическую защиту архивной информации;
- декодирование сообщений электронной почты;
- «прозрачное» уплотнение исполнимых файлов .EXE и .DLL;
- создание самораспаковывающихся многотомных архивов;
- выбор или настройку коэффициента сжатия.

1.3 Программа архиватора и их цели.

Программы, осуществляющие упаковку и распаковку файлов, называются программами архиваторами.

Большие по объему архивные файлы могут быть размещены на нескольких дисках (томах). Такие архивы называются многотомными. Том - это составная часть многотомного архива. Создавая архив из нескольких частей, можно записать его части на несколько дискет.

В настоящее время применяется несколько десятков программ - архиваторов, которые отличаются перечнем функций и параметрами работы. Из числа наиболее популярных программ можно выделить:

WinZip, версия 8.0, пожалуй, самый известный архиватор. Это наиболее популярный архиватор, используемый в Интернете. Часто является бесплатным или входит еще в какую-нибудь программу в качестве бесплатного бонуса. Начиная с версии ME (Millenium Edition), WinZip входит даже в систему Windows. Кроме того, существует немало модификаций архиватора WinZip, таких как zip - magic, 7 - zip, g - zip и т.п., различных по эффективности сжатия.

WinRar - автор программы - Евгений Рошал. Однопользовательская система обойдется вам в 29\$. Загрузить испытательскую (пробную) версию можно с сайта www.RaRlab.com. Главный конкурент WinZip на просторах Интернета. Обладая лучшими характеристиками, он постепенно теснит другие форматы, но с авторитетом WinZip пока сладить не может. Удобный (русифицированный) интерфейс и достаточно высокая скорость работы в сочетании с низкими системными требованиями обещают WinRar хорошее будущее.

WinAce 2.0 - свежая версия старого архиватора, пополнившаяся очень интересными функциями и новым алгоритмом сжатия. Этот архиватор использует самый большой размер словаря для архивирования (4 Мб), что во многом объясняет его высокие результаты.

Существуют архиваторы с узкой специализацией, которые могут работать только с одним форматом файлов, как, например, SfArk, который может сжимать только звуковые в формате SF2. Архиватор WavPack отлично «жмет» звуковые файлы в формате wav. Архиватор DJVU специализируется на сжатии сканированных изображений. Также есть целая плеяда универсальных архиваторов таких как, UFA, 777, ACB, IMP, LZOP, UNARC, BOA, Arhangel.

Различных архиваторов существует немало. Эффективность того или иного архиватора зависит от нескольких факторов:

1. Содержимого сжимаемого файла (текстовый, графический, звуковой).
2. От того, какая программа выбрана в качестве архиватора.
3. От настроек программы - архиватора.

Архивация используется как средство для уменьшения размера файла - в архиве файла, как правило, занимает меньше места, чем в обычном состоянии, и применяется в основном для следующих целей:

- Высвобождения места на жестком диске компьютера.
- Сокращения объема файлов, пересылаемых по электронной почте.
- Размещение данных, едином носителе.
- Размещение 1 объемного файла на нескольких носителях, когда он не помещается на 1 носитель.
- Распаковка сжатых файлов.

1.4 Актуальность архиватора на сегодняшний день.

На сегодняшний день имеется много типов архиваторов таких как WinRar, WinZip и т. д., которые работают в среде Windows, имеют больше функций и лучший интерфейс чем их предшественники-DOS, у них выше степень защиты данных.

В настоящее время растет популярность формата RAR и соответствующих программ. Но, хотя технология RAR обеспечивает высокую степень сжатия, стандартом она так и не стала, отчасти из-за не очень гибкого механизма работы с большими архивами.

Рассмотренные программы по большей части ориентированы на работу с архивами в формате ARJ или ZIP, но, как правило, содержат встроенные средства (или допускают подключение внешних модулей) для распаковки и просмотра и архивов других типов.

Все программы обладают удобными инсталляторами и стандартными средствами деинсталляции. Как правило, архиваторы могут выборочно регистрироваться в качестве средства для обработки распознаваемых ими типов файлов.

Лучшие из рассмотренных программ относятся к категории условно-бесплатных, некоммерческие разработки уступают им в разнообразии функций, совместимости и удобстве (хотя и не в эффективности сжатия). Лидером обзора являются Zip-ориентированные утилиты ZipMagic фирмы Mijenix, Zip Explorer Pro компании Aeco Systems и уже упомянутая WinZip фирмы Nico Mak Computing. Все они обеспечивают совместимость с большим числом форматов, удобны в использовании.

Глава 2. Python.

2.1 О языках программирования.

Python — это высокоуровневый язык программирования общего назначения, который используется в том числе и для разработки веб-приложений. Язык ориентирован на повышение производительности разработчика и читаемости кода.

Правильное русское произношение названия языка программирования — Пайтон, но чаще используется искажённое — Питон.

Python поддерживает несколько парадигм программирования: структурное, объектно-ориентированное, функциональное, императивное и аспектно-ориентированное. В языке присутствует динамическая типизация, автоматическое управление памятью, полная интроспекция, механизм обработки исключений, поддержка многопоточных вычислений и удобные высокоуровневые структуры данных. Программный код на Python организовывается в функции и классы, которые могут объединяться в модули, а они в свою очередь могут быть объединены в пакеты. Python обычно используется как интерпретируемый, но может быть скомпилирован в байт-код Java и в MSIL (в рамках платформы .NET).

Разработчики языка Python придерживаются определённой философии программирования, называемой «The Zen of Python» («Дзен Питона» или «Дзен Пайтона»):

Красивое лучше, чем уродливое.

Явное лучше, чем неявное.

Простое лучше, чем сложное.

Сложное лучше, чем запутанное.

Плоское лучше, чем вложенное.

Разреженное лучше, чем плотное.

Читаемость имеет значение.

Особые случаи не настолько особые, чтобы нарушать правила.

При этом практичность важнее безупречности.

Ошибки никогда не должны замалчиваться.

Если не замалчиваются явно.

Встретив двусмысленность, отбрось искушение угадать.

Должен существовать один — и, желательно, только один — очевидный способ сделать это.

Хотя он поначалу может быть и не очевиден, если вы не голландец.
Сейчас лучше, чем никогда.

Хотя никогда зачастую лучше, чем прямо сейчас.

Если реализацию сложно объяснить — идея плоха.

Если реализацию легко объяснить — идея, возможно, хороша.

Пространства имён — отличная штука! Будем делать их побольше!

По производительности интерпретируемый Python похож на все остальные подобные языки, но возможность компиляции в байт-код позволяет добиться большей производительности.

По сравнению с Ruby и некоторыми другими языками, в Python отсутствует возможность модифицировать встроенные классы, такие, как int, str, float, list и другие.

В Python присутствует глобальная блокировка интерпретатора (GIL) — при своей работе основной интерпретатор постоянно использует большое количество потоко-небезопасных данных. В основном это словари, в которых хранятся атрибуты объектов, и обращения к внешнему коду, поэтому, во избежание разрушения этих данных при совместной модификации из разных потоков, перед началом исполнения нескольких инструкций (обычно ста) поток интерпретатора захватывает GIL, а по окончании освобождает.

Существует несколько реализаций языка — CPython (основная), Jython, PyS60, IronPython, Stackless, Python for .NET, PyPy, python-safethread, Unladen Swallow, tinypy.

Разработка на Python

Python — очень универсальный язык.

Он подходит и для веб-разработки, но с прикладной точки зрения обычно это не лучший выбор.

Ruby в веб-разработке обычно эффективнее и экономически выгоднее (с идентичным качеством результата).

Компилируемые языки (Go / Elixir / Java) — значительно эффективнее по производительности.

Разработка проекта на PHP будет заметно дешевле.

Но в сферах анализа данных и машинного обучения Python сейчас несомненно вне конкуренции.

2.2 История языка Python.

В конце восьмидесятых голландский программист Гвидо ван Россум работал в научно-исследовательском центре математики и информатики, в котором создавал язык программирования ABC. Гвидо вкладывал все силы в этот проект и многому научился в работе над ним.

В декабре 1989 года офис закрылся на рождественские каникулы, и Гвидо было нечем заняться. Тогда он просто ради развлечения написал интерпретатор для нового языка программирования. Свой несерьезный проект он назвал «Python» в честь любимого комедийного шоу «Летающий Цирк Монти Пайтона».

Уже в 1991 году вышел первый релиз Python версии 0.9.0. Новый язык вобрал в себя многие идеи из ABC, а также из других языков: например, система модулей взята из языка Modula-3.

Еще немного позже вышла версия 1.0. В ней появились элементы функционального программирования — функции map, filter, reduce. Их мы будем изучать в следующих курсах.

В версии 2.0 появились знаменитые списковые включения, позаимствованные из языков SETL и Haskell. Тогда же появился сборщик мусора — механизм автоматического управления памятью. В той же второй версии в нем появилась поддержка работы с циклическими структурами.

Мы не стали погружаться в эту тему в базовом курсе, но говорим о ней подробно в других курсах по Python.

Python постоянно развивается с самого начала и до сегодняшнего дня. Сейчас почти все используют следующую, третью версию.

Python изначально позиционировался как «язык для каждого», поэтому в нем сделан упор на читаемость кода и лаконичность синтаксиса. Эта философия делает язык более понятным и востребованным для новичков.

Первый релиз третьей версии Python вышел еще в 2008 году, но переход на него идет до сих пор.

Дело в том, что третья версия не полностью обратно совместима со второй. Чтобы перейти на третью версию, нельзя просто заменить интерпретатор — придется менять код. Разработчики идут на такой шаг, если проект все еще развивается.

Но есть программы, работа над которыми закончена: они находятся в состоянии «только поддержка» и называются legacy. Такие проекты часто не переходят на третью версию — обычно просто некому заниматься адаптацией.

Если программа становится legacy-проектом, никаких новых трудностей не возникнет. Сложнее ситуация с библиотеками. Некоторые важные библиотеки все еще написаны на второй версии. Обновлять их некому, а заменить — пока нечем.

Именно поэтому переход на третью версию Python растянулся так надолго.

Более того, с 2020 года вторая версия больше не поддерживается — в ней перестали устранять уязвимости и критические ошибки. Теперь устаревшие библиотеки могут нести в себе угрозу для всего проекта.

К счастью, большинство популярных и востребованных библиотек уже переведены на третью версию, а все современные рекомендовано писать на Python 3.7.x.

Python распространен не только в виде языка, на котором пишутся конечные проекты. Этот язык часто используется для автоматизации различных задач — например, на нем написан Ansible, ПО для системного администрирования. Поэтому Python часто уже установлен в операционных системах.

Особенно велика вероятность обнаружить его в операционных системах Linux. Кроме того, какая-то версия Linux будет установлена на сервер, с которым будет работать ваше web-приложение или многопользовательская игра.

Тут-то и кроется проблема: в ОС может быть установлен Python второй версии — такое до сих пор встречается. И заменить его будет нельзя, ведь замена может привести к выходу из строя всей ОС.

2.3 Язык Python. Цели и актуальность.

Главная цель, которую ставил перед собой автор — это упрощение процесса программирования. Чтобы писать код было проще, он должен стать более читабельным и понятным для человека. У Python открытый исходный код. Одно из преимуществ этого языка — возможность запускать программы на нём как на ОС Windows, так и на macOS и Linux.

Сейчас язык находится на пике своей популярности, и будет на нём ещё не менее пяти-десяти лет. Вот пять причин, почему питон всё ещё актуален.

Питон перечеркнул миф о сложности разработки. У языка понятный синтаксис, который базируется на английском языке. На питоне легко писать и его легко читать.

Большое количество справочной литературы. Вы не будете испытывать недостатка актуальной информации, потому что её много в открытом доступе — книги, сайты, форумы, видеоролики, платные и бесплатные курсы.

Множество инструментов. Для питона создано множество инструментов, фреймворков и сред разработки, которые позволяют упростить решение многих задач. Можно воспользоваться готовым решением и не тратить время.

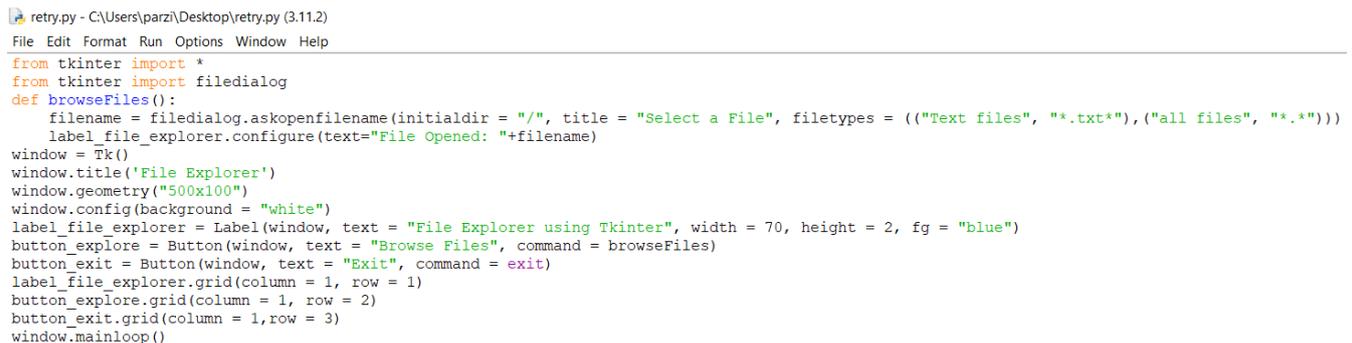
Минимализм. Не нужно писать полотна лишнего кода. Динамическая типизация и другие функции языка дают возможность меньше заморачиваться над шаблонностью кода и упрощать его.

Востребованность специалистов. Если вы сейчас начнёте изучать питон, то у вас не будет проблемы, что через пять лет вы останетесь без работы. И став питон-разработчиком легко освоить любой другой язык.

Практическая часть.

Код защиты папок и файлов.

1. Начнём с кодами.



```
retry.py - C:\Users\parzi\Desktop\retry.py (3.11.2)
File Edit Format Run Options Window Help
from tkinter import *
from tkinter import filedialog
def browseFiles():
    filename = filedialog.askopenfilename(initialdir = "/", title = "Select a File", filetypes = (("Text files", "*.txt*"), ("all files", "*.*")))
    label_file_explorer.configure(text="File Opened: "+filename)
window = Tk()
window.title('File Explorer')
window.geometry("500x100")
window.config(background = "white")
label_file_explorer = Label(window, text = "File Explorer using Tkinter", width = 70, height = 2, fg = "blue")
button_explore = Button(window, text = "Browse Files", command = browseFiles)
button_exit = Button(window, text = "Exit", command = exit)
label_file_explorer.grid(column = 1, row = 1)
button_explore.grid(column = 1, row = 2)
button_exit.grid(column = 1, row = 3)
window.mainloop()
```

Рисунок 1. Коды Python.

Пояснение:

- Tkinter - кроссплатформенная библиотека для разработки графического интерфейса на языке Python (начиная с Python 3.0 переименована в tkinter). Tkinter расшифровывается как Tk interface, и является интерфейсом к Tcl/Tk. Tkinter входит в стандартный дистрибутив Python.
- import - применяется для того, чтобы сделать код в одном модуле доступным для работы в другом.
- filedialog – диалоговые окна открытия и сохранения файлов.
- def – позволяет создавать функцию.
- browseFiles - поиск пути к файлу.
- str – строковый тип данных.
- file - путь к zip-файлу.
- w - применяется для записи нового файла.
- write - записать в файл одну строку.
- close() - закрывает zip-файл.
- as - используется для оборачивания выполнения блока инструкций менеджером контекста.
- extractall() - извлекает все элементы из архива.
- for - данный цикл применяется в том случае, если нужно выполнять тот или иной фрагмент вплоть до момента, когда будет исполнено заданное условие.
- walk - возвращает объект-генератор, из которого получают кортежи.

- path - путь к файлу.
- join - отвечает за объединение списка строк с помощью определенного указателя.
- relpath - возвращает относительный путь к файлу path.
- compress_type - тип сжатия.
- ZIP_DEFLATED - стандартный тип сжатия при архивации в zip.
- while - используется, когда точное число повторений неизвестно и может изменяться в зависимости от поведения переменной в теле цикла.
- input - позволяет обеспечить ввод пользовательских данных из консоли.
- elif - позволяет проверить истинность нескольких выражений и в зависимости от результата проверки, выполнить нужный блок кода.
- else - является специальным, оно указывает нам на то, что если ни одно из вышестоящих условий не сработало, то будет выполнено действие, стоящее после else.
- print - выводит заданные объекты на стандартное устройство вывода (экран) или отправляет их текстовым потоком в файл.

Рисунок 2. Используемые библиотеки.

2. Позволяем создавать функцию архивирование для одного файла.

Рисунок 3. Функция сжатия файла.

3. Позволяем создавать функцию извлечение для одного архива.

Рисунок 4. Функция извлечение файла.

4. Позволяем создавать функцию архивирование для одной папки.

Рисунок 5.

5. Создаём ввод пользовательских данных из консоли.

Рисунок 6. Цикл для работы программы без перезапусков.

6. Итог работы.

Рисунок 7. Запуск архиватора.

Рисунок 8. Архивация файла Серикжан Аян СИБ-413.docx.

Рисунок 9. Извлечение файла python-3.11.2-amd64.

Рисунок 10. Архивация папки «Новая папка».

Рисунок 11. Ошибка, ненайденных файлов или папок.

Охрана труда.

1. Общие требования охраны труда

1.1. На работу в качестве архивариуса назначается служащий (технический исполнитель) соответствующей квалификации, прошедший вводный и первичный на рабочем месте инструктажи по охране труда.

1.2. Архивариус должен пройти специальный инструктаж и получить группу I по электробезопасности.

1.3. Архивариус независимо от квалификации и стажа работы не реже одного раза в шесть месяцев должен проходить повторный инструктаж по охране труда; в случае нарушения требований безопасности труда, при перерыве в работе более чем на 60 календарных дней он должен пройти внеплановый инструктаж.

1.4. Архивариус, допущенный к самостоятельной работе, должен знать: нормативные документы по вопросам ведения архивного дела в организации. Порядок приема и сдачи документов в архив, их хранения и пользования ими. Единую государственную систему делопроизводства. Порядок составления описаний документов постоянного и временного хранения и актов об уничтожении документов. Порядок оформления дел и их подготовки к хранению и использованию. Порядок ведения учета и составления отчетности. Структуру

организации. Основы организации труда. Правила эксплуатации технических средств. Правила, нормы и инструкции по охране труда. Способы оказания первой помощи при несчастных случаях:

Правила внутреннего трудового распорядка;

Правила пожарной безопасности.

1.5. Архивариус, показавший неудовлетворительные знания требований охраны труда при ведении архивного дела, к самостоятельной работе не допускается.

1.6. Архивариус, направленный для участия в несвойственных его должности работах, обязан пройти целевой инструктаж по безопасному проведению предстоящих работ.

1.7. Архивариусу запрещается выполнять работы, к которым он не допущен в установленном порядке, а также пользоваться инструментом и оборудованием, с которыми он не имеет навыков безопасного обращения.

1.8. Во время работы на архивариуса могут оказывать неблагоприятное воздействие в основном следующие опасные и вредные производственные факторы:

- падающие архивные документы при их размещении для хранения, например, на
- стеллаже;

- физические перегрузки, например, при переноске документов;
- нахождение на высоте, например, на стремянке;
- недостаточная освещенность рабочей зоны;
- острые кромки, шероховатости на поверхности лестниц, стремянок, стеллажей, тележек.

1.9. Для предупреждения неблагоприятного воздействия на здоровье архивариусу следует пользоваться средствами индивидуальной защиты.

1.10. Для предупреждения возможности возникновения пожара в помещениях архива, архивариус должен соблюдать требования пожарной безопасности сам и не допускать нарушения этих требований другими работниками; курить разрешается только в специально отведенных для этого местах.

1.11. Архивариус обязан соблюдать трудовую и производственную дисциплину, правила внутреннего трудового распорядка.

1.12. Если с кем-либо из работников произошел несчастный случай, то пострадавшему необходимо оказать первую помощь, сообщить о случившемся руководителю и сохранить обстановку происшествия, если это не создает опасности для окружающих.

1.13. Архивариус при необходимости должен уметь оказать первую помощь, пользоваться аптечкой.

1.14. Для предупреждения возможности заболеваний архивариусу следует соблюдать правила личной гигиены, в том числе перед приемом пищи необходимо тщательно мыть руки с мылом.

1.15. Архивариус, допустивший нарушение или невыполнение требований инструкции по охране труда, несет ответственность в соответствии с действующим законодательством.

2. Требования охраны труда перед началом работы

2.1. Перед началом работы архивариусу следует надеть спецодежду, проверить наличие средств индивидуальной защиты, медицинской аптечки для оказания первой помощи, а также средств пожаротушения.

2.2. Спецодежда должна быть соответствующего размера, чистой и не стеснять движений.

2.3. Прежде чем приступать к работе, следует проверить состояние помещений архива; при необходимости следует навести чистоту, порядок и обеспечить наличие свободных проходов.

2.4. Архивариусу следует проверить исправность лестниц и стремянок, а также средств малой механизации, применяемых для перемещения архивных документов.

2.5. Архивариус должен лично убедиться в том, что все меры, необходимые для обеспечения безопасности, выполнены.

2.6. Архивариус не должен приступать к работе, если у него имеются сомнения в обеспечении безопасности при выполнении предстоящей работы.

2.7. Перед началом работы нужно убедиться в достаточности освещения рабочей зоны, особенно в темное время суток.

2.8. Перед началом работы архивариусу следует обратить внимание на рациональную организацию рабочего места, подготовить необходимые технические средства и проверить их работу.

3. Требования охраны труда во время работы

3.1. Архивариус, находящийся в болезненном или переутомленном состоянии, а также под воздействием алкоголя, наркотических веществ или лекарств, притупляющих внимание и реакцию, не должен приступать к работе, так как это может привести к несчастному случаю.

3.2. Во время работы архивариус должен быть вежливым, вести себя спокойно и

выдержанно, избегать конфликтных ситуаций, которые могут вызвать нервно-эмоциональное напряжение и отразиться на безопасности труда.

3.3. Во время работы архивариусу следует быть внимательным, не отвлекаться от выполнения своих обязанностей.

3.4. Во время работы архивариусу не разрешается оставлять свое рабочее место; это допускается только с разрешения непосредственного руководителя.

3.5. При размещении архивных документов на местах хранения архивариус должен учитывать следующие общие правила:

- при размещении документов следует принять меры против их самопроизвольного смещения, просадки, падения;
- между стеллажами должны быть предусмотрены проходы шириной не менее 1 м;
- документы следует размещать с учетом их массы и способности деформироваться под воздействием вышерасположенных материалов.

3.6. При хранении документов на стеллажах рекомендуется их размещение примерно одинакового размера.

3.7. В случае обнаружения неправильно размещенных для хранения документов необходимо принять меры к их разборке и укладке вновь с устранением замеченных недостатков.

3.8. Архивариус должен использовать исправные стремянки и лестницы, имеющие специальные устройства от скольжения.

3.9. Нельзя использовать вместо стремянки или лестницы случайные предметы (например, ящики, стулья и т. п.).

3.10. Нельзя становиться на верхние ступени стремянок и лестниц во избежание потери устойчивости и падения.

3.11. Не допускается бросать любые материалы и документы с верхних полок и

стеллажей.

- 3.12. Архивариус должен следить за исправностью и устойчивостью стеллажей, полок для хранения архивных документов.
 - 3.13. Стеллажи не следует перегружать материалами и документами выше допустимой на них нагрузки.
 - 3.14. Архивные документы нужно размещать на стеллажах таким образом, чтобы они не мешали работе и исключалась возможность их падения.
 - 3.15. Нельзя хранить документы навалом и размещать их вплотную к радиаторам и трубам отопления.
 - 3.16. Архивариусу следует проявлять осторожность при переноске архивных документов, чтобы не споткнуться во время ходьбы о возможные препятствия.
 - 3.17. При переноске архивных документов следует соблюдать установленные нормы перемещения (для мужчин и женщин) тяжестей вручную.
 - 3.18. Помещение для размещения архива должно быть обеспечено первичными средствами пожаротушения, установкой пожарной автоматики, телефоном и аптечкой для оказания первой помощи.
 - 3.19. Рабочее место пользователя персональным компьютером следует оборудовать подставкой для ног, имеющей ширину не менее 300 мм, глубину не менее 400 мм, регулировку по высоте в пределах до 150 мм и по углу наклона опорной поверхности подставки до 20°.
 - 3.20. Для уменьшения напряжения зрения следует установить на экране монитора оптимальный цветовой режим (если такая возможность имеется); при этом рекомендуются ненасыщенные цвета: светло-зеленый, желто-зеленый, желто-оранжевый, желто-коричневый; по возможности следует избегать насыщенных цветов, особенно красного, синего, яркозеленого.
 - 3.21. С целью снижения зрительного и костно-мышечного утомления пользователю следует соблюдать установленный режим труда и отдыха.
4. Требования охраны труда в аварийных ситуациях
 - 4.1. В случае обнаружения нарушений требований охраны труда, которые создают угрозу здоровью или личной безопасности, архивариус должен обратиться к непосредственному руководителю и сообщить ему об этом; до устранения угрозы следует прекратить работу и покинуть опасную зону.
 - 4.2. При несчастном случае, внезапном заболевании необходимо немедленно оказать первую помощь пострадавшему, вызвать врача или помочь доставить пострадавшего к врачу, а затем сообщить руководителю о случившемся.
 - 4.3. При обнаружении пожара или признаков горения (задымление, запах гари, повышение температуры и т. п.) необходимо немедленно уведомить об этом пожарную охрану.
 - 4.4. До прибытия пожарной охраны нужно принять меры по эвакуации людей, имущества и приступить к тушению пожара.

4.5. Следует организовать встречу подразделений пожарной охраны и оказать помощь в выборе кратчайшего пути для подъезда к очагу пожара.

5. Требования охраны труда по окончании работы

По окончании работы архивариусу следует выполнить следующие работы:

5.1. Проверить состояние пожарной безопасности в помещениях, где хранятся архивные документы.

5.2. Выключить электрооборудование и персональный компьютер.

5.3. Привести в порядок рабочее место.

5.4. Снять спецодежду и убрать средства индивидуальной защиты, почистить их, при необходимости сдать в ремонт, стирку или химчистку.

5.5. Архивное помещение закрыть на замок, сдать ключи под охрану в установленном порядке.

5.6. Тщательно вымыть руки теплой водой с мылом, при необходимости принять душ.

Заключение

В конце 80-ых и начала 90-ых годов проблемы связанные с защитой информации беспокоят как специалистов в области компьютерной безопасности, так и многочисленных рядовых пользователей персональных компьютеров. Это связано с глубокими изменениями, вносимыми компьютерной технологией в нашу жизнь. Изменился сам подход к понятию информация. Этот термин сейчас больше используется для обозначения специального товара, который можно купить, продать, обменять на что-то другое и т.д. При этом стоимость подобного товара зачастую превосходит в десятки, а то и в сотни раз стоимость самой вычислительной техники, в рамках которой он функционирует.

Естественно, возникает потребность защитить информацию от несанкционированного доступа, кражи, уничтожения и других преступных действий. Однако, большая часть пользователей не осознает, что постоянно рискует своей безопасностью и личными тайнами. И лишь немногие хоть, каким-либо образом защищают свои данные. Пользователи компьютеров регулярно оставляют полностью незащищенными даже такие данные как налоговая и банковская информация, деловая переписка и электронные таблицы. Проблемы значительно усложняются, когда вы начинаете работать или играть в сети, так как хакеру намного легче в это время заполучить или уничтожить информацию, находящуюся на вашем компьютере.

На данный момент с развитием технологий существует большое количество угроз, направленных на несанкционированный доступ к информации, на ее искажение, удаление, например, вирусы которые успешно внедрились в повседневную компьютерную жизнь и покидать ее в обозримом будущем не собираются. Нужно четко представлять себе, что никакие аппаратные, программные и любые другие решения не смогут гарантировать абсолютную надежность и безопасность данных в информационных системах. Но следует помнить, что большая концентрация защитных средств в информационной системе может привести не только к тому, что система окажется очень дорогостоящей, но и к тому, что у нее произойдет существенное снижение коэффициента готовности. Например, если такие ресурсы системы, как время центрального процессора будут постоянно тратиться на работу антивирусных программ, шифрование, резервное архивирование и тому подобное, скорость работы пользователей в такой системе может упасть до нуля.

Поэтому главное при определении мер и принципов защиты информации это квалифицированно определить границы разумной безопасности и затрат на средства защиты с одной стороны и поддержания системы в работоспособном состоянии и приемлемого риска с другой.

Список литературы.

- <https://ronl.org/kontrolnyye-raboty/informatika/410942/>
- <https://ru.wikipedia.org/wiki/Архиватор>
- <https://studfile.net/preview/7674241/page:10/>
- <http://allrefs.net/c3/3x15n/p11/>
- <https://web-creator.ru/articles/python>
- <https://dzen.ru/a/YryFCQDh8FNoaxda>
- <https://gb.ru/blog/yazyk-programmirovaniya-python/>
- https://esalmaty.kz/images/docs/2020/dp-rk-714-01_process_upr_tech_bezopasnosti_i_okhranoi_truda.pdf
- <https://studbooks.net/2238374/informatika/zaklyuchenie>

Приложение.

```
from tkinter import *
from tkinter import filedialog
def browseFiles():
    filename = filedialog.askopenfilename(initialdir = "/", title = "Select a
File", filetypes = (("Text files", "*.txt*"),("all files", "*.*")))
    label_file_explorer.configure(text="File Opened: "+filename)
window = Tk()
window.title('File Explorer')
window.geometry("500x100")
window.config(background = "white")
label_file_explorer = Label(window, text = "File Explorer using Tkinter",
width = 70, height = 2, fg = "blue")
button_explore = Button(window, text = "Browse Files", command =
browseFiles)
button_exit = Button(window, text = "Exit", command = exit)
label_file_explorer.grid(column = 1, row = 1)
button_explore.grid(column = 1, row = 2)
button_exit.grid(column = 1,row = 3)
window.mainloop()
```