



# Постановка задачи- Файловый Обзоратель

Лабораторная работа №3

Предлагается реализовать приложение **«Файловый обозреватель»**, в котором основная работа ведется с файловой системой компьютера.

Основное назначение приложение - **отображение размера в процентах:**

## 1. Текущей папки

C:\Program Files:

Microsoft	445,123 KB	43 %
Adobe	234,744 KB	25 %
Symantec	98,906 KB	10 %
...		

## 2. Файлов одинакового типа

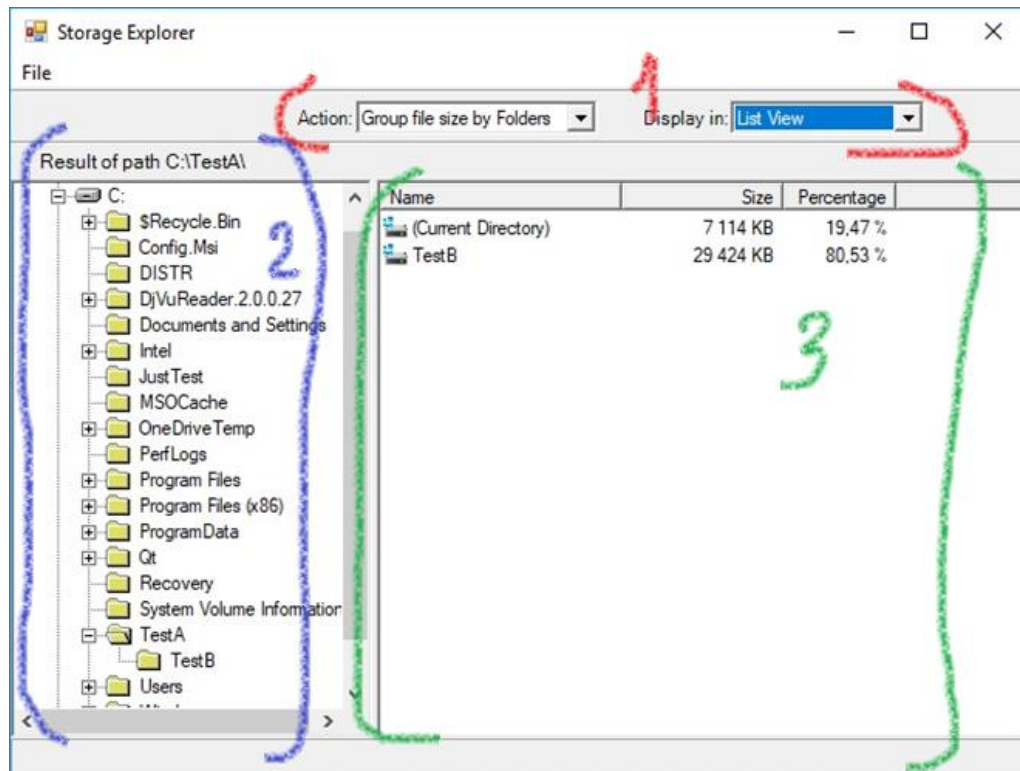
D:\

*.mp3	602,456 KB	47 %
*.jpg	305,830 KB	30 %
*.doc	245,355 KB	20 %
...		

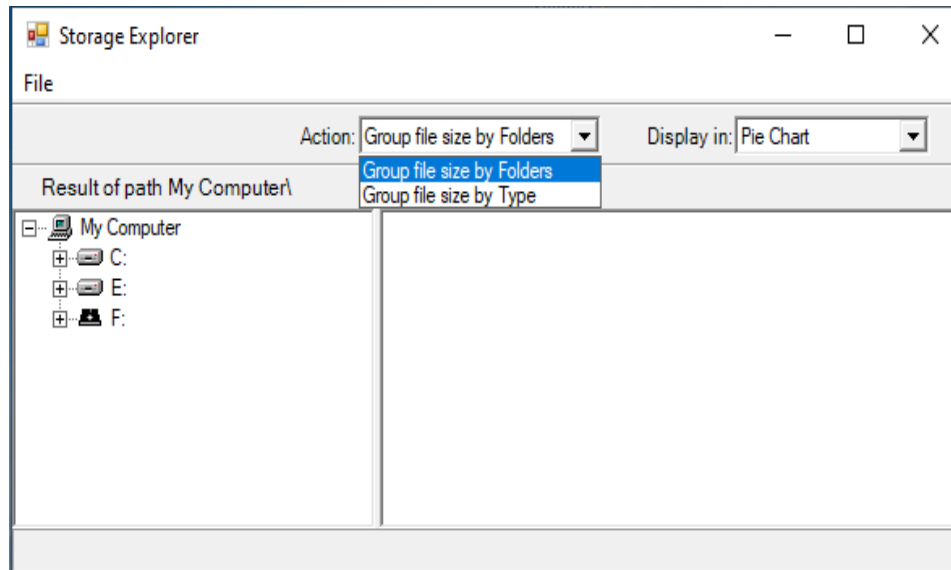
# Графический интерфейс(1)

Рассмотрим примерный графический интерфейс, которому будем следовать при разработке собственного приложения с помощью библиотек QT. Интерфейс включает в себя:

1. Меню(1), в котором происходит выбор типа просмотра размера группы файлов *по папкам* либо *по типам*. Выбор способа отображения(списком, с помощью диаграммы).
2. Двух панелей(2)и(3). В первой отображаем файловую систему. Здесь осуществляется выбор папки для отображения ее размера в соответствии с выбранными условиями. На второй панели– непосредственное отображение размера в соответствии с выбранными условиями.



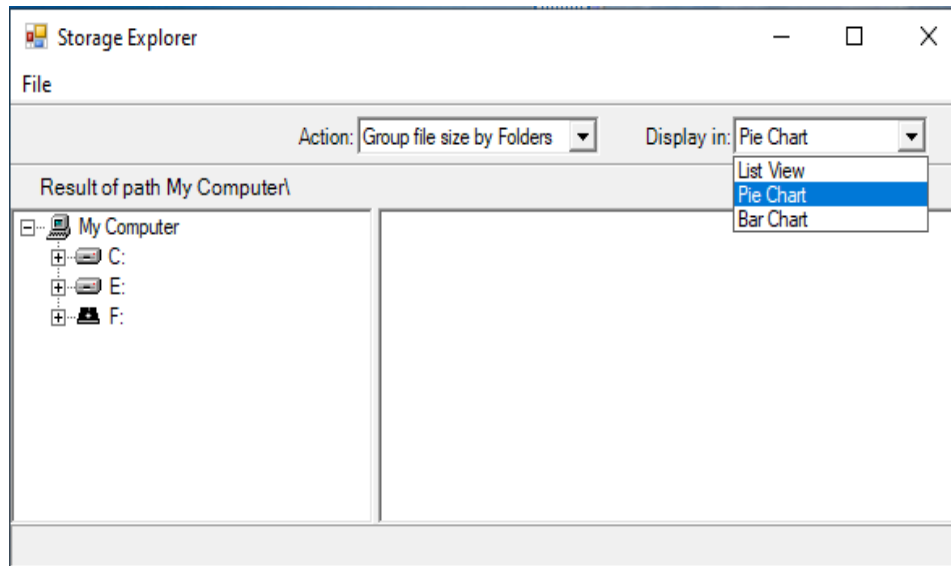
# Графический интерфейс(2)



Здесь, демонстрируется состав выпадающих списков.

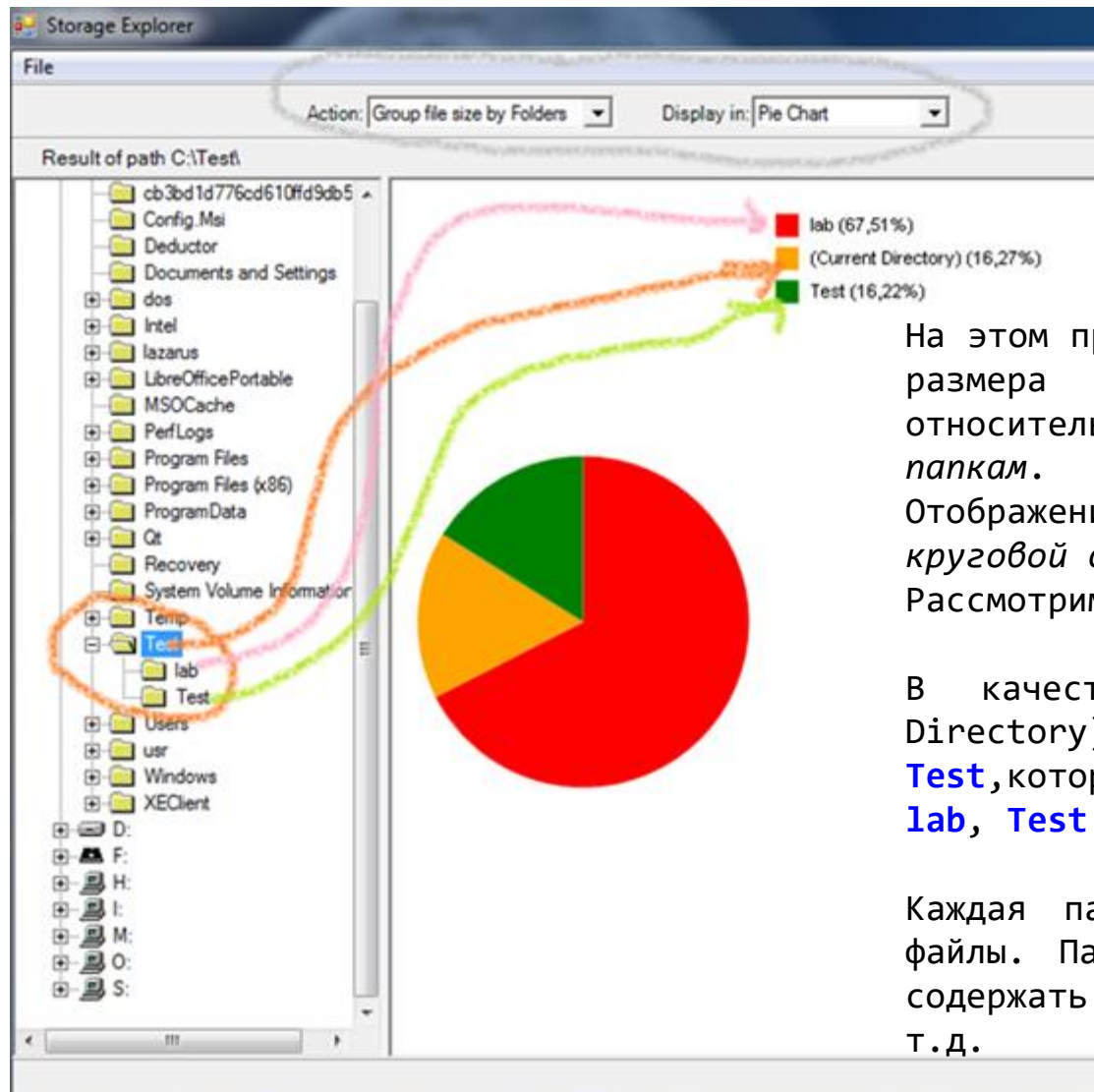
С помощью первого списка(**Action**) определяем типа просмотра размера группы файлов  
*1. по папкам;*  
*2. по типам.*

С помощью второго выпадающего списка осуществляется выбор способа отображения(**Display in**):



- 1.Отображение с помощью списка(**компонента ListView**)
2. Отображение с помощью круговой диаграммы (**Pie Chart**)
- 3.Отображение с помощью столбчатой диаграммы (**Bar Chart**)

# Графический интерфейс(3.1)



На этом примере, показано отображение размера группы файлов, в % относительно текущей папки, *по папкам*.

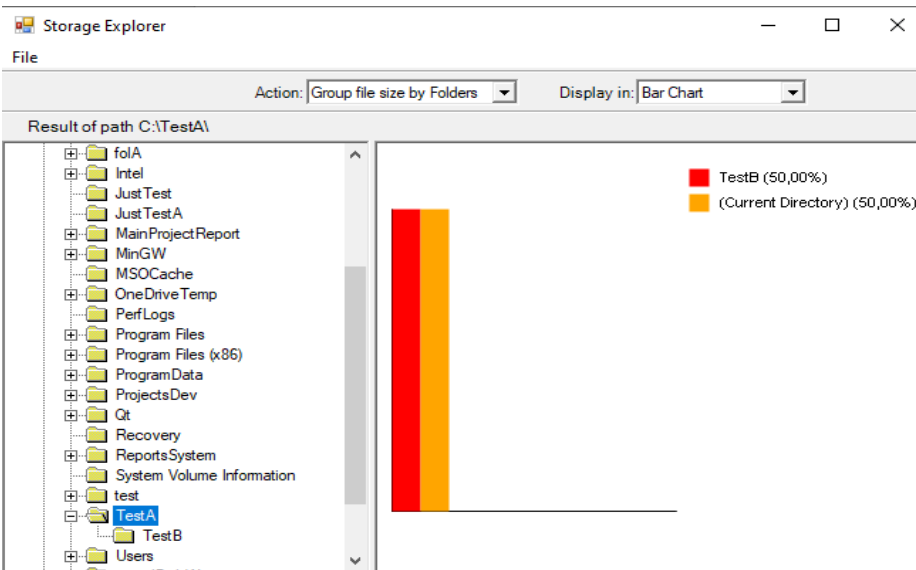
Отображение происходит с помощью *круговой диаграммы*.

Рассмотрим данный пример.

В качестве текущей папки(Current Directory), здесь выбрана папка **Test**, которая, в свою очередь содержит **lab**, **Test**.

Каждая папка здесь может содержать файлы. Папки **lab**, **Test**, могут, также содержать файлы и вложенные папки и т.д.

# Графический интерфейс(3.2)



На этом примере, показано отображение размера **группы файлов**, в % относительно текущей папки, *по папкам*.

Отображение происходит с помощью *Столбчатой диаграммы*.

В качестве текущей папки(Current Directory), здесь выбрана папка **TestA**, которая, в свою очередь содержит **TestB**.

Из примера, видно, что текущая папка и вложенная занимают одинаковое пространство на диске по объему.

The screenshot shows the Storage Explorer application window. The left pane displays a tree view of folders under the path C:\TestA. The right pane shows a list view titled "Result of path C:\TestA\". The list contains two entries: "(Current Directory)" and "TestB". Both entries have a size of 29 424 KB and a percentage of 50,00 %.

Name	Size	Percentage
(Current Directory)	29 424 KB	50,00 %
TestB	29 424 KB	50,00 %

На этом примере, показано отображение размера **группы файлов**, в % относительно текущей папки, с помощью *списка*. Информация представлена в виде таблицы с атрибутами(Имя, Размер, Проценты )

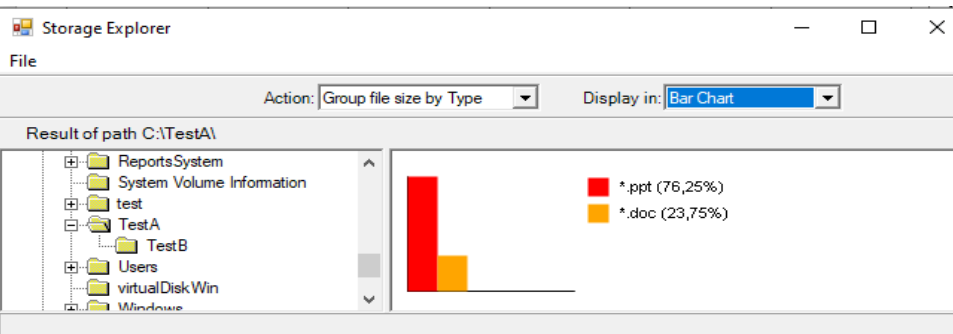
В качестве текущей папки(Current Directory), здесь выбрана папка **TestA**, которая, в свою очередь содержит **TestB**. В случае отображения с помощью списка, здесь отображается еще и размер занимаемой области относительно текущей папки.

# Графический интерфейс(3.3)



На этом примере, показано отображение размера **группы файлов по типам**, в % относительно текущей папки

Отображение происходит с помощью **Круговой диаграммы**.



На этом примере, показано отображение размера **группы файлов по типам**, в % относительно текущей папки

Отображение происходит с помощью **Столбчатой диаграммы**.

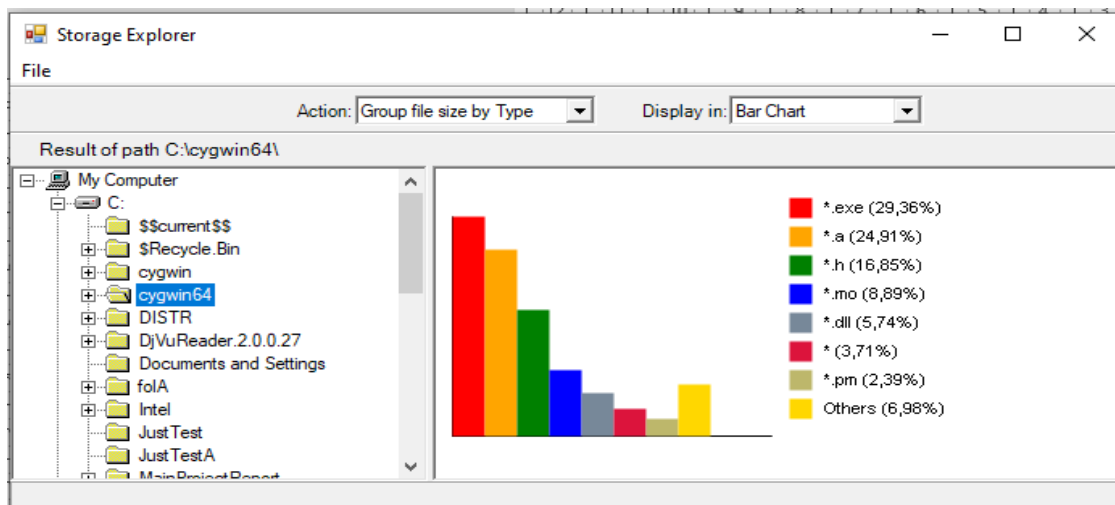
The screenshot shows the Storage Explorer window with the 'File' tab selected. The 'Action' dropdown is set to 'Group file size by Type' and the 'Display in' dropdown is set to 'List View'. The left pane shows the directory structure with 'TestA' selected. The right pane displays a list view of the files, showing their names, sizes, and percentages.

Name	Size	Percentage
*.ppt	44 332 KB	76,25 %
*.doc	13 810 KB	23,75 %

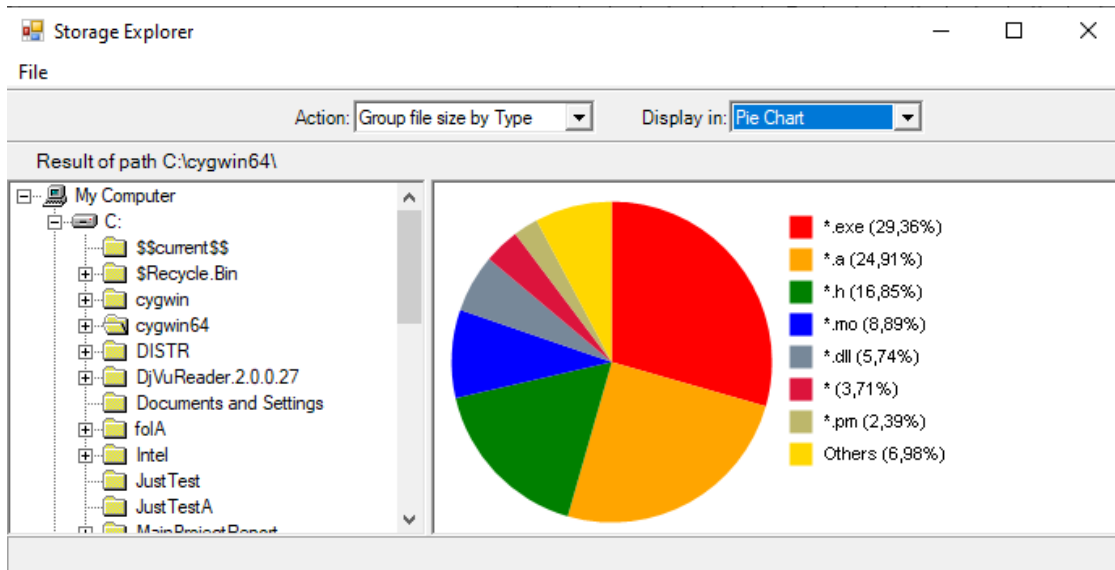
На этом примере, показано отображение размера **группы файлов по типам**, в % относительно текущей папки, с помощью **списка**. Информация также, представлена в виде таблицы с атрибутами(**Имя, Размер, Проценты** ).

# Графический интерфейс(3.4)

Замечание: При отображении размера *группы файлов по типам*, может возникнуть ситуация, когда количество файлов одного типа минимально и они занимают очень, очень мало места относительно других, то эти файлы выносятся в категорию «Другие». Например.



При решении задачи, нужно будет задать порог в %, при котором файлы будут относиться к категории «Другие»





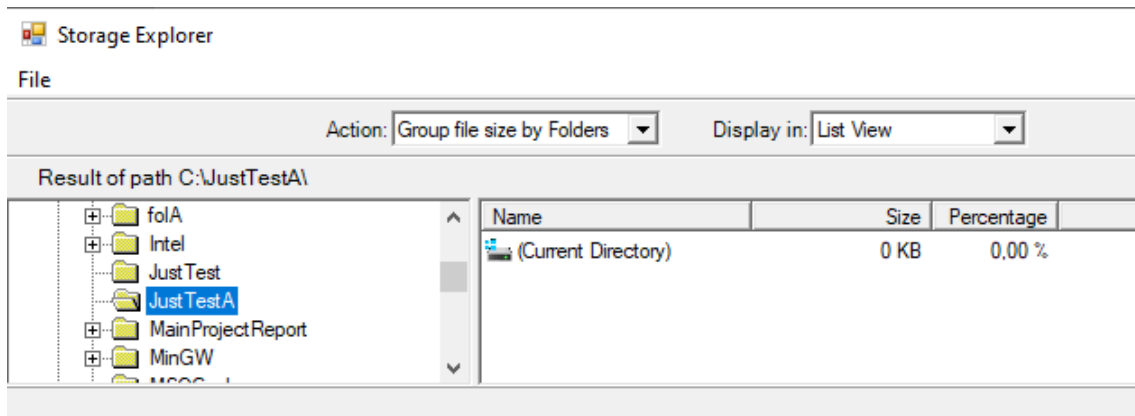
# Графический интерфейс(4.1)

Рассмотрим все возможные ситуации которые могут возникнуть при выборе некоторой папки.

Здесь рассматривается папка JustTestA, которая является пустой.  
Итак, первый случай.

## 1.Текущая папка является пустой.

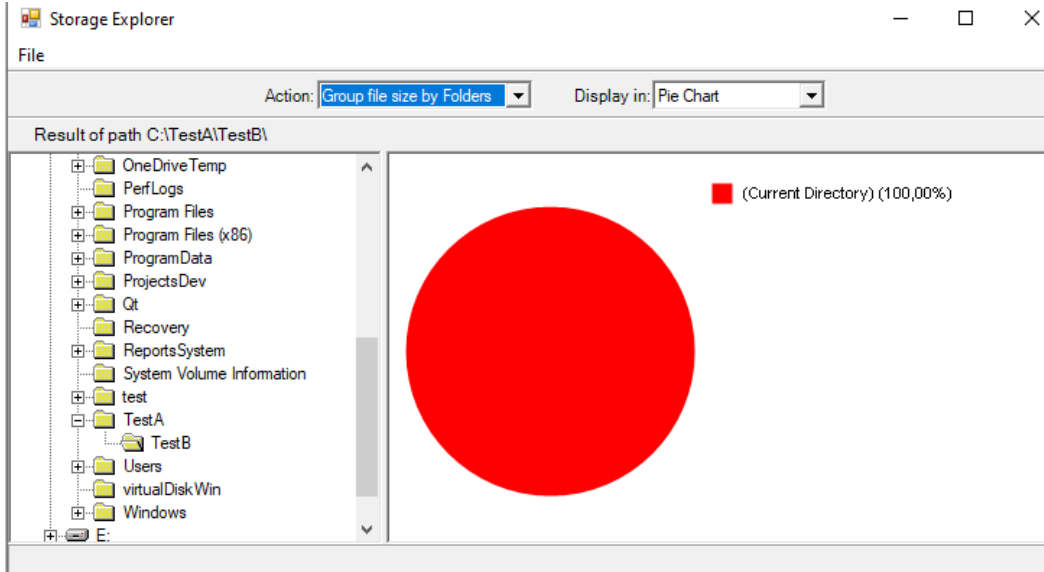
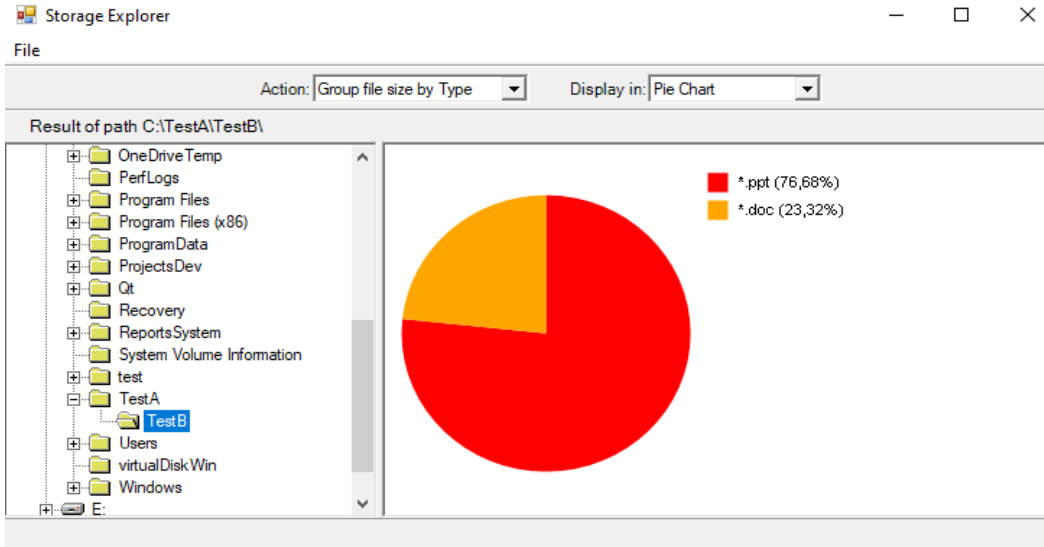
В этом случае, графики не отображаем. В области отображения нужно указать , что папка пустая, процент занимаемой области равен 0.(Также можно предложить свой способ отображения)



# Графический интерфейс(4.2)

Здесь рассматривается папка TestB, которая не является пустой.

2. Текущая папка не является пустой и у нее нет вложений  
Здесь варианты группировки файлов– по типам и по папкам.

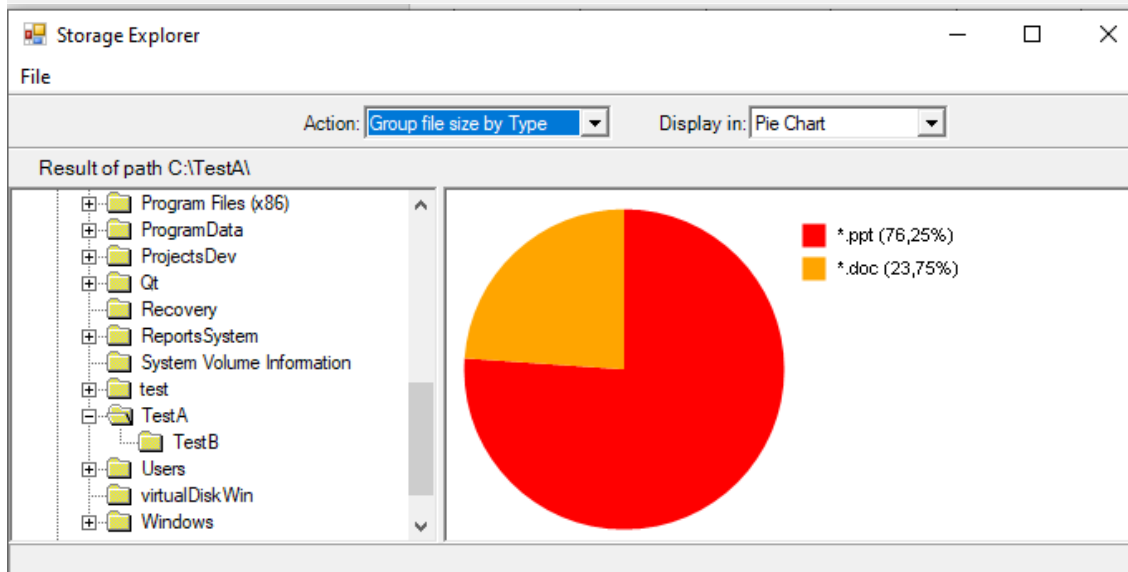
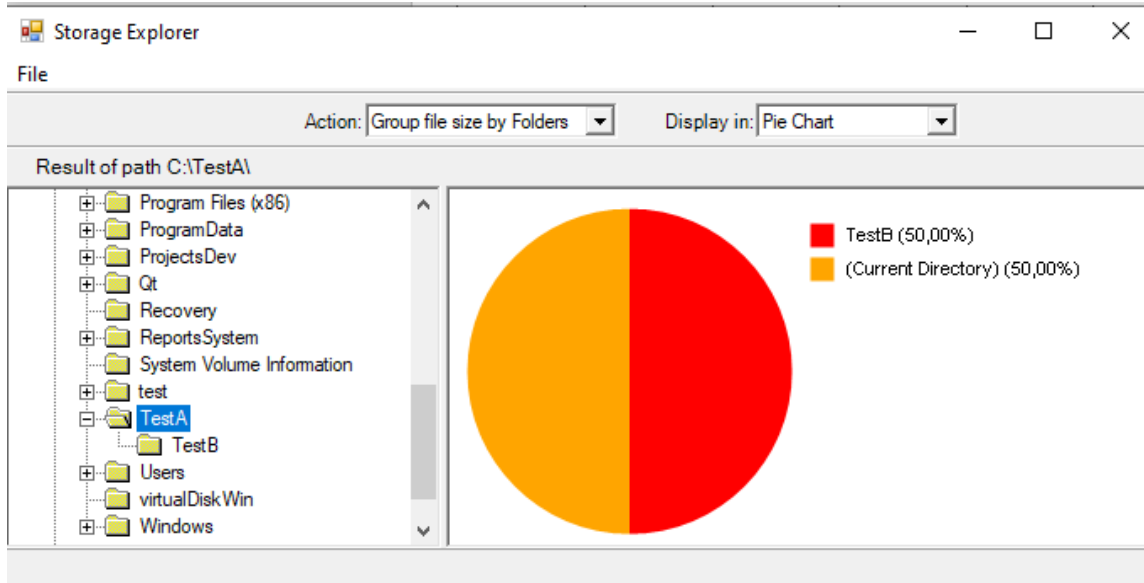


# Графический интерфейс(4.3)

Здесь рассматривается папка TestA, которая не является пустой.

3. Текущая папка не является пустой и у нее есть вложения.

Здесь варианты группировки файлов – по типам и по папкам.

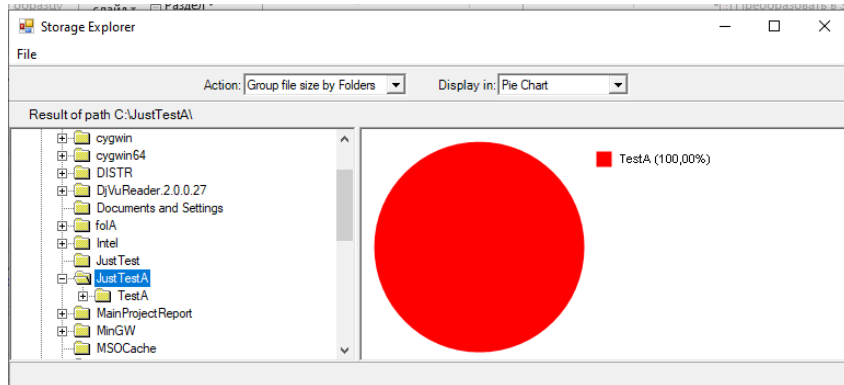


# Графический интерфейс(4.4)

Здесь рассматривается папка JustTestA, которая не является пустой.

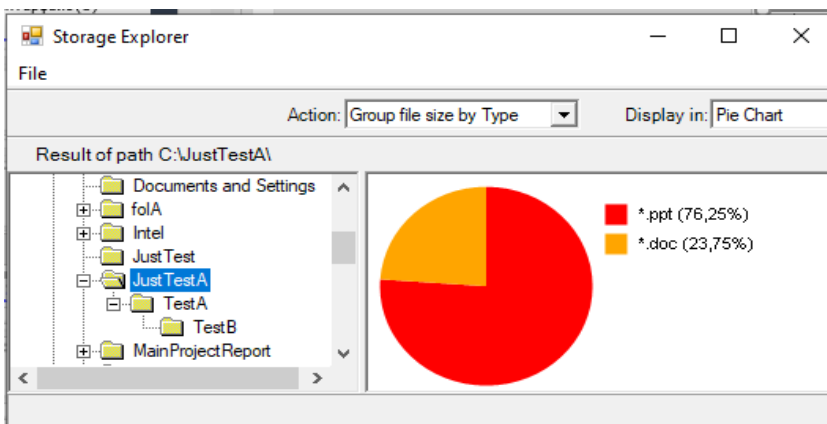
4.Текущая папка является пустой и у нее есть вложения.

Здесь варианты группировки файлов – по типам и по папкам. Отображение – графики и списки.



Storage Explorer window showing the path C:\JustTestA\. The 'Display in' dropdown is set to 'List View'. The table shows the contents of the directory.

Name	Size	Percentage
TestA	58 848 KB	100,00 %
(Current Directory)	0 KB	0,00 %



Storage Explorer window showing the path C:\JustTestA\. The 'Display in' dropdown is set to 'List View'. The table shows the contents of the directory.

Name	Size	Percentage
*.ppt	44 332 KB	76,25 %
*.doc	13 810 KB	23,75 %

# Графический интерфейс(4.5)

На этом примере демонстрируется общая ситуация. Когда текущая папка **TestA** – не является пустой и имеет уровень вложенности больше 1. Причем, для демонстрации, каждая папка содержит одинаковый объем. Пример показывает, как выполняется вычисление размера папки. Во первых, в качестве результата всегда отображаем первый уровень вложенности (на примере (**Current Dirrectiory**) и **TestB**). Размер для папок вычисляется рекурсивно. Т.е. Для **TestB** выполняется подсчет общего размера файлов(в текущей папке и в **TestC**)

The screenshot shows the Storage Explorer application window. The title bar reads "Storage Explorer". Below the title bar is a "File" menu. The main area is divided into two panes. The left pane shows a directory tree with the following structure:

- cygwin
- cygwin64
- DISTR
- DjVuReader.2.0.0.27
- Documents and Settings
- folA
- Intel
- JustTest
- JustTestA
  - TestA
    - TestB
      - TestC
- MainProjectReport
- MinGW
- MSOCache
- OneDriveTemp
- PerfLogs
- Program Files
- Program Files (x86)

The right pane shows the "Result of path C:\JustTestA\TestA\". It contains a table with the following data:

Name	Size	Percentage
(Current Directory)	29 424 KB	33,33 %
TestB	58 848 KB	66,67 %

Hand-drawn annotations include an orange arrow pointing from the "TestA" folder in the tree to the "(Current Directory)" row in the table, and a green arrow pointing from the "TestB" folder in the tree to the "TestB" row in the table.

## Задание.

Проанализировать предложенные случаи и  
ответить на вопрос - Все ли случаи  
здесь рассмотрены?

Что будем использовать для реализации этого задания?

### Паттерны:

1. Стратегия,
2. Наблюдатель,
3. Шаблонный метод,
4. Адаптер,
5. Синглтон,
7. MVC(Модель – Представление – Контроллер)
8. Виджеты.

Целью данного задания является разработка архитектуры ПО с использованием изученных паттернов.

Но, с чего нужно начать?

Мы с вами, последовательно, обсудим каждый шаг проектирования **Файлового Обзорателя**.

Начнем с функции, являющаяся основой нашего ПО, функция которая проводит главную вычислительную работу – вычисление общего размера файлов *сгруппированных по папкам* и *сгруппированных по типам*.

Функция, которая не зависит от архитектуры ПО.

Теперь нужно подумать, как будет выглядеть функция, что у нее буде на вход, и что будет у нее на выходе.

На вход будем подавать *путь к текущей папки*, а на выходе должны получить пару значений:

1. В случае «*сгруппированных по папкам* »

<«Имя Папки», размер в кб>

2. В случае «*сгруппированных по типам*»

<«\*.тип файла», размер в кб>

Теперь вопрос вам, как нужно представить эту функцию?

**??????? Function(path, flag) - ????**



Требования.

??????? **Function(path, flag) - ????**

Третья лабораторная работа, будет состоять из нескольких частей.(мы все обсуждаем)

В первой части, сдаете, отдельным консольным приложением, реализацию основной функции – вычисление размера.

Цель, показать, что всевозможные условия обработаны. Также подготавливаете тесты(примеры), на которых будете демонстрировать состоятельность вашей функции.

Далее представлены, небольшие примеры, которые помогут понять как можно выполнить обход по файловой системе.

В нашей работе придется выполнять работу с файловой системой. Рассмотрим основные классы, а также подходы к работе.

**QDir** - Класс QDir обеспечивает доступ к структурам каталогов и их содержимому.

**QFileInfo** - QFileInfo предоставляет информацию об имени файла и позиции (пути) в файловой системе, его правах доступа и о том, является ли он каталогом или символической ссылкой и т.д. QFileInfo также может использоваться для получения информации о ресурсе Qt.

Также доступны размер файла и время последнего *изменения/чтения*.

## Пример обхода папок в файловой системе

```
QDir myDir;
QString info;
myDir.setPath("E:/TestA");
info = myDir.path();
/* Рассмотрим способы обхода содержимого папок на диске.
Предлагается вариант решения, который может быть применен для более сложных задач.
Итак, если требуется выполнить анализ содержимого папки, то необходимо организовать обход
содержимого.
Например:*/
if (fileInfo.isDir()) {
/*
Если fileInfo папка, то заходим в нее, чтобы просмотреть находящиеся в ней файлы.
Если нужно просмотреть все файлы, включая все вложенные папки, то нужно организовать
рекурсивный обход.
*/
QDir dir = fileInfo.dir();

if (dir.cd(fileInfo.fileName())) {
/*
Если зашли в папку, то пройдемся по контейнеру QFileInfoList, полученного методом
entryInfoList,
*/
foreach (QFileInfo inf, dir.entryInfoList(QDir::Files | QDir::NoDotAndDotDot, QDir::Type)) {
qDebug() << inf.fileName() << "---" << inf.size();
}
dir.cdUp();//выходим из папки
}}
```

## Пример вывода информации о файлах

```
QDir dir; //объявляем объект работы с папками
dir.setFilter(QDir::Files | QDir::Hidden | QDir::NoSymLinks);
///устанавливаем фильтр выводимых файлов
dir.setSorting(QDir::Size | QDir::Reversed); //устанавливаем сортировку "от
меньшего к большему"
QFileInfoList list = dir.entryInfoList(); //получаем список файлов
директории
std::cout << "      Bytes Filename" << std::endl; //выводим заголовок
/* в цикле выводим сведения о файлах */
for (int i = 0; i < list.size(); ++i) {
QFileInfo fileInfo = list.at(i);
std::cout << qPrintable(QString("%1 %2").arg(fileInfo.size(),
10).arg(fileInfo.fileName())); //выводим в формате "размер имя"
std::cout << std::endl; //переводим строку
```