

Министерство цифрового развития, связи и массовых коммуникаций РФ  
Федеральное государственное бюджетное образовательное учреждение высшего  
образования

«Сибирский государственный университет  
телекоммуникаций и информатики» (СибГУТИ)

Кафедра Прикладной Математики и Кибернетики (ПМиК)

**Лабораторная работа №4**  
**по дисциплине «Визуальное Программирование и Человеко-Машинное**  
**Взаимодействие»**  
**CWT-анализ приложения WizTree**

Выполнил: студент 3 курса

ИВТ, гр. ИП-111

Прилепа М. К.

Проверила:

доцент ПМиК, Мерзлякова Е.Ю.

## Оглавление

1. Описание пользователя. Описание программы
2. Задачи интерфейса
3. Задача пользователя. Решение. История
  - 3.1. Задача №1
  - 3.2. Задача №2
4. Заключение

## Описание пользователя. Описание программы

Программа предназначена для обычного пользователя ПК, обладающего навыками пользования проводником. Приложение не русифицировано, потому подробное изучение интерфейса, а также поиск необходимых параметров и функций могут быть затруднены для человека, который не знает английского языка.

Основной мощью программы является то, что она в разы быстрее считывает файлы и папки, чем стандартные методы операционной системы для работы с файловой системой. Это связано с тем, что мы не знаем, где какие stat-данные (метаданные) файла хранятся физически на жёстком диске, по этому для изъятия той или иной информации часто приходится файловой системе танцевать с бубном. Программе же достаточно на прямую считать вообще все stat-данные подряд, ибо целью является собрать их все целиком и сформировать соответствующее двумерное дерево размеров, в чём и суть программы.

### Задачи интерфейса

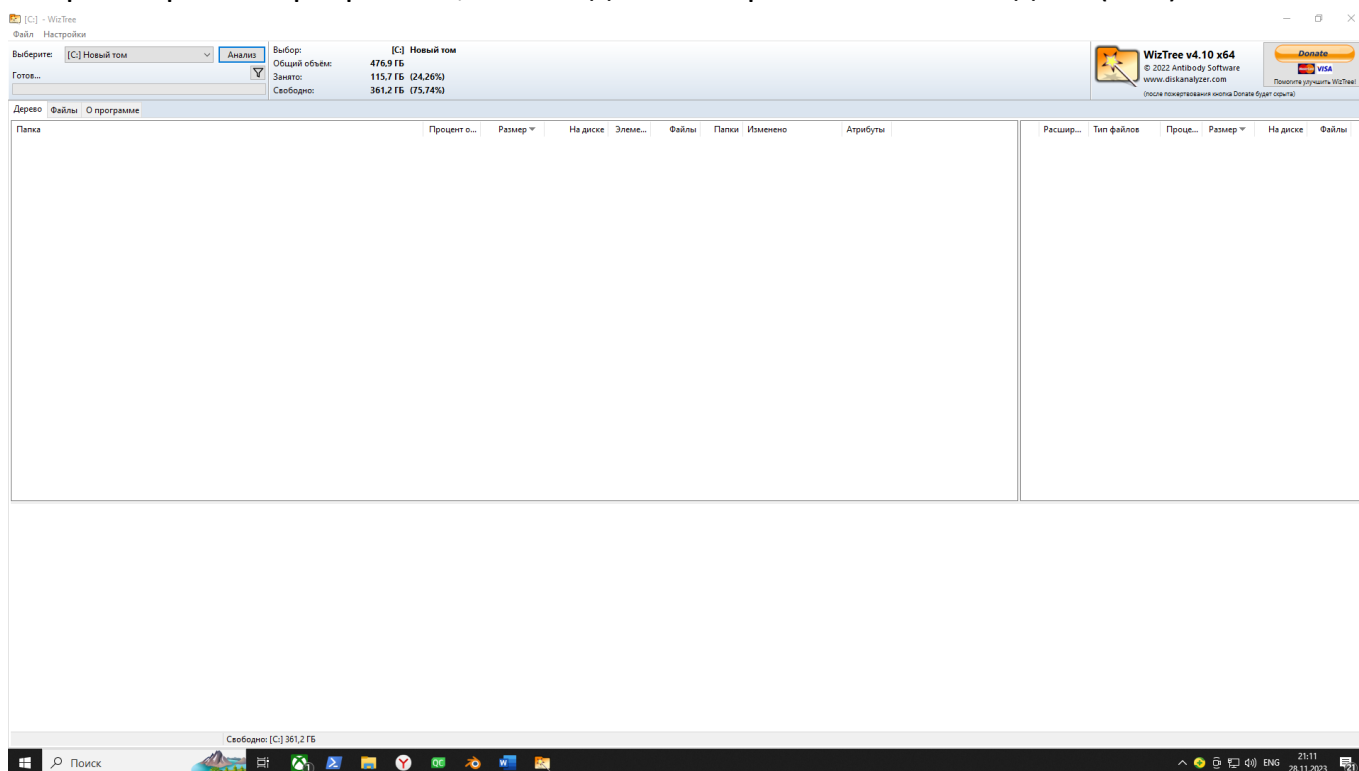
- 1.) Выбор логического диска (тома) для дальнейшего анализа.
- 2.) Считывание вообще всех stat-данных выбранного тома. Выводится графически завершенность действия каждый фрейм, вместо того, чтобы программа просто зависла.
- 3.) Вывод привычного пользователю дерева файлов, а также вывод особого графического 2d-дерева размеров файлов (графическая карта размеров).
- 4.) Возможность сортировать не только по размерам, а ещё и по названиям файлов, особым атрибутам stat-записи, количеству элементов, файлов и папок.
- 5.) Возможность выбора любого файла на графической карте, чтобы отрисовка думала, что это корневой файл, ибо некоторые файлы могут быть столь малы, что неуместятся даже в пиксель.

# Задача пользователя. Решение. История

## Задача №1

Пользователь впервые открыл программу с целью выяснить, чем забился его жёсткий диск. Этой причиной могут послужить особые обновления операционной системы с сохранение старой версии операционной системы и всех сопутствующих файлов Program Files/Data и т.д., а это прямо много. Ещё одним примером может послужить Яндекс Браузер, который постоянно пишет журнал в формате SQLite. Судя по содержимому базы данных, я не совсем понял, что там за данные, но весят они много. В общем-то причин пассивного переполнения жёсткого диска существует сколько угодно, а задача программы – показать эту самую причину. У меня, конечно, проблем с переполнением нет, но программу можно использовать чисто ради того, чтобы поглазеть на графическую карту.

При открытии программы, необходимо выбрать логический диск (том):



По умолчанию выбран логический том с операционной системой. Даже если таких томов на деле несколько, всё равно в независимости от того, какая из них будет запущена, системный том автоматически будет обозван буквой С, т.к. каждая операционная система отдельно задаёт в себе настраиваемый пользователем перечень букв дисков.

Если нас устраивает системный том, то достаточно провести курсор до кнопки “Анализ” и ровно один раз кликнуть левой кнопкой мыши (ЛКМ) по ней. В противном же случае, проводим курсор мыши к надписи буквы тома и его названия, которое, кстати, тоже настраивается, как и буква диска, так что пользователь должен знать, как он же раньше назвал диски и их буквы. По умолчанию это “[C:] Новый том”. После единственного клика ЛКМ, появляется список томов.

Здесь была бы “ваша реклама”, но... Именно в этот момент узнаётся страшный недостаток программы – она перебиндывает кнопку PrintScreen и я в принципе не могу сделать скриншот, т.к. если убрать фокус с программы, список закроется. Сама по себе кнопка PrintScreen ничего не делается, т.е. становится бесполезна. Перебинд работает вообще на всю клавиатуру. Системные бинды (и только они) по типу Ctrl+Shift+Esc (открытие диспетчера задач) ещё функционируют.

После думательных процессов, тянем курсор к нужному тому и давим ровно один раз ЛКМ. Обратите внимание, что соседняя кнопка “Анализ”, в случае выбора через список, не требуется.

А вот так выглядит уже думательный процесс программы в независимости от того, была ли задействована именно кнопка “Анализ” или нет, а все дальнейшие события аналогичны:

ФайлНастройки

Выберите: [C:] Новый том

Отмена

Сканирование... (Папки: 81 258 Файлы: 221 479)

Выбор: [C:] Новый том

Общий объем: 476,9 ГБ

Занято: 115,7 ГБ (24,25%)

Свободно: 361,2 ГБ (75,75%)

Дерево

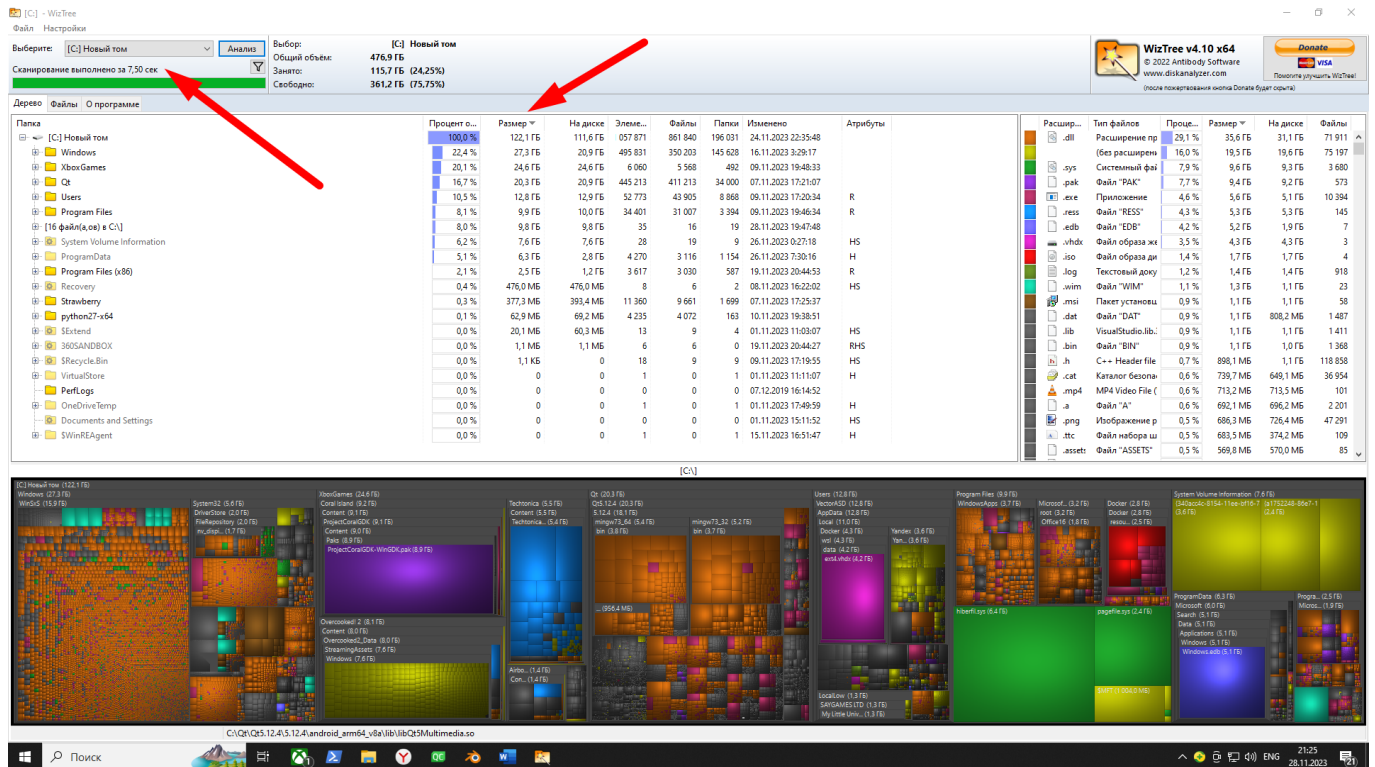
Файлы

О программе

Папка

Процент о...

По окончании думательного процесса, наслаждаемся деревом элементов и графической картой размеров файлов:



По умолчанию сортировка работает на физический размер элемента, т.е. без учёта накладных на фрагментацию всех вложенных файлов. Сортировка идёт по невозрастающей. Также, выписывается время, затраченное для считывания 1057858 элементов – всего 7.5 секунд. Довольно-таки непостоянный результат для стандартных методов операционной системы для работы с файловой системой, ибо ушло бы в 100 раз больше времени, а если бы ещё была затронута групповая политика и прочие штуки по безопасности, то и вовсе улетело бы в 10000 раз больше времени. Знаю это по своему опыту, когда удалял бывшую операционную систему на несистемном жёстком диске, ибо в каждой подпапке ещё нужно было получить соответствующее разрешение - пришлось шаманить скрипт на питоне, который буквально съел всю операционную, оставив пару папок, которые уже ничего не требовали лишнего для удаления обычным проводником.

Давайте посмотрим, что находится внутри QtCreator'a (с расчётом того, что я его уже ранее установил, указав вообще все флажки опций установки, какие только можно было вообразить интерфейсу установщика).

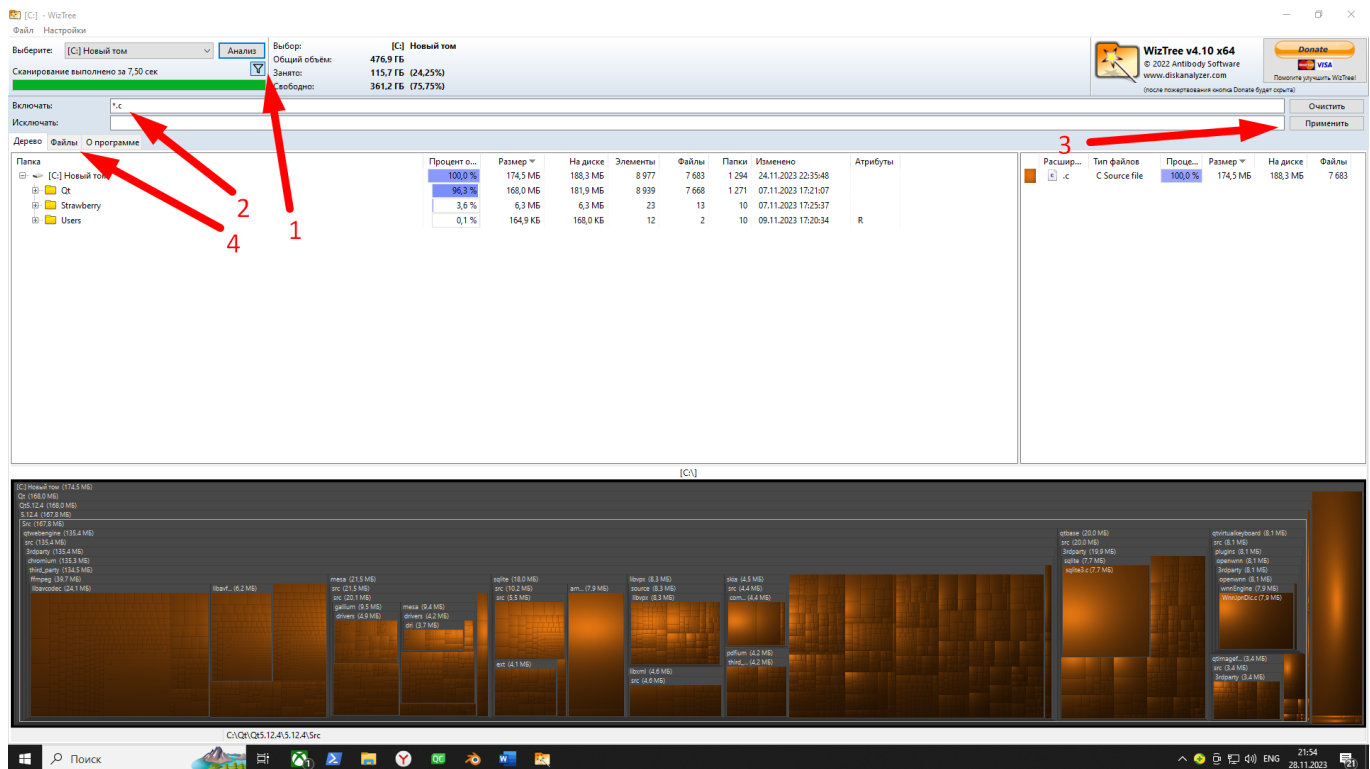
Выбор фильтра для дерева элементов не влияет на тот факт, что графическая карта всегда показывает файлы и папки относительно их размеров.

Задача выполнена, т.к. пользователь теперь знает, что на диске миллион файлов, из которых лишь один Qt занимает 445213 элементов, следовательно внезапное увеличение жёсткого диска после его установки можно проигнорировать, либо как-то уже через сам QtCreator удалять все ненужные компиляторы, что, кстати, занимают ровно половину всего пространства папки Qt, чем сам QtCreator.

## Задача №2

Пользователю требуется узнать, какой самый большой .с файл находится на его логическом томе.

Для этого нужно снова открыть приложение, пододвинуть курсор на кнопку “Анализ” и нажать один раз ЛКМ, либо проделать альтернативные действия, что уже описаны выше. Теперь, вместо того, чтобы лезть в папку Qt, нужно пододвинуть курсор к значку фильтра и нажать один раз ЛКМ:



Здесь уже показаны действия 1, 2 и 3.

- 1.) Пододвинули курсор туда, нажали 1 раз ЛКМ.
- 2.) Пододвинули курсор к полю ввода с меткой “Включать:”, нажали один раз ЛКМ, подумали очень сильно, ещё сильнее, а после очень сложных математических вычислений ядерного взрыва несуществующего острова, написали надпись “\*.с”, предварительно нажав Alt+Shift (или что там забиндено у пользователя) для смены языка на английский, если ещё этого не сделано.
- 3.) Нацелили курсор на кнопку “Применить” и нажали 1 раз ЛКМ. Это действие можно опустить, если к надписи “\*.с” добавить нажатие Qt::Key\_Enter, точнее, клавиши Enter.
- 4.) Осталось нажать на заветную кнопочку “Файлы” и увидеть:



WizTree v4.10 x64  
© 2022 Antibody Software  
www.diskanalyzer.com  
(после повторения имени Директория будет скрыта)

Выбор: [C:] Новый том  
Общий объем: 476,9 ТБ  
Занято: 115,7 ТБ (24,25%)  
Свободно: 361,2 ТБ (75,75%)

Включать: \*.\*  
Исключать:

Дерево Файлы Программы

Фильтр поиска:

Очистить Учет только имени Учет всего пути Максимум файлов: 1 000 Папки

Имя файла	Процент на диске	Размер	На диске	Изменено	Атрибуты
C:\Qt\Qt5.12.4\5.12.4\Src\qtvirtualkeyboard\src\plugins\openwnn\3rdparty\openwnn\wnnEngine\WnnJpnDic.c	4,5 %	7,9 MB	7,9 MB	23.05.2019 19:20:40	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\sqlite\sqlite3.c	4,4 %	7,7 MB	7,7 MB	13.06.2019 3:59:14	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\sqlite\amalgamation\sqlite3.c	4,2 %	7,4 MB	7,4 MB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\sqlite\amalgamation\sqlite3.c	2,6 %	6,2 MB	6,2 MB	07.03.2016 7:49:20	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/networks_sorting.c	1,3 %	2,2 MB	2,2 MB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/networks_merging.c	1,0 %	1,7 MB	1,7 MB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\libxml\src\testapi.c	0,8 %	1,4 MB	1,4 MB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\ffmpeg\libavcodec\metasound_data.c	0,6 %	986,5 KB	986,5 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\ffmpeg\libavcodec\on2avcdata.c	0,3 %	586,5 KB	586,5 KB	13.06.2019 3:59:14	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\ffmpeg\libavcodec\on2avcdata.c	0,3 %	529,1 KB	529,1 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\ffmpeg\libavcodec\h264_data.c	0,3 %	514,3 KB	514,3 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\sqlite\amalgamation\shell\shell.c	0,3 %	493,2 KB	493,2 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\sqlite\amalgamation\shell\shell.c	0,3 %	485,7 KB	485,7 KB	13.06.2019 3:59:14	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,3 %	466,7 KB	466,7 KB	10.08.2018 18:35:38	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,3 %	460,6 KB	460,6 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,3 %	454,9 KB	454,9 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,3 %	453,1 KB	453,1 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	443,1 KB	443,1 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	438,4 KB	438,4 KB	13.06.2019 3:59:14	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	428,7 KB	428,7 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	421,7 KB	421,7 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	410,8 KB	410,8 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	396,7 KB	396,7 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	376,3 KB	376,3 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	351,1 KB	351,1 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	342,2 KB	342,2 KB	13.06.2019 3:59:14	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	335,1 KB	335,1 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	331,2 KB	331,2 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	309,4 KB	309,4 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	305,5 KB	305,5 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	296,9 KB	296,9 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	294,0 KB	294,0 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	293,5 KB	293,5 KB	13.06.2019 3:59:14	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	289,7 KB	289,7 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	283,4 KB	283,4 KB	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwbengine\src\3rdparty\chromium\third_party\skia\src\compute\hl/gen/ducat.c	0,2 %	270,7 KB	270,7 KB	06.06.2019 19:53:16	A

C:\Qt\Qt5.12.4\5.12.4\Src\qtvirtualkeyboard\src\plugins\openwnn\3rdparty\openwnn\wnnEngine\WnnJpnDic.c (1 000 файлов(ов) Общий размер: 113,4 MB)

Теперь задача выполнена сполна – самый жирный файл Си-синтаксиса составляет

C:\Qt\Qt5.12.4\5.12.4\Src\qtvirtualkeyboard\src\plugins\openwnn\3rdparty\openwnn\wnnEngine\WnnJpnDic.c

## Заключение

Разумеется это ещё не предел возможностей данной программы, т.к. поверх фильтра, что мы/пользователь указали ранее, можно ещё натянуть один фильтр, выбрать максимальное количество выводимых файлов запроса, ну и т.д.

Кстати, некоторые даты создания и изменения файлов могут быть не совсем правдивы, т.к. данные файлы появились ещё до появления моего компьютера. Это результат распаковщика установщика, который часть stat-записей напрямую подделывает. Конечно в рамках компьютера, где эти файлы родились, это актуальная информация, но некоторые неопытные пользователи могут почувствовать себя, как на Рен-TV. Данное замечание никак не касается WizTree.

Первый Си-шный файл, не входящий в папку C:\Qt, даже не вошёл в первые 1000 файлов, зато в режиме Максимум файлов “ВСЕ” он такой:

C:\Qt\Qt5.12.4\5.12.4\Src\qtwebengine\src\3rdparty\chromium\third_party\ffmpeg\libavcodec\mips\vp9_intra_msa.c	0,0 %	21,4 КБ	24,0 КБ	06.06.2019 19:53:16	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtbase\src\3rdparty\freetype\src\cff\cf2ft.c	0,0 %	21,4 КБ	24,0 КБ	13.06.2019 3:59:14	A
C:\Users\VectorASD\AppData\Local\Programs\Python\Python312\c\nmake\nmakehlp.c	0,0 %	21,4 КБ	24,0 КБ	02.10.2023 13:27:52	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtimageformats\src\3rdparty\libtiff\libtiff\tif_print.c	0,0 %	21,4 КБ	24,0 КБ	24.05.2019 16:52:56	A
C:\Qt\Qt5.12.4\5.12.4\Src\qtwebengine\src\3rdparty\chromium\third_party\ffmpeg\libavfilter\af_afir.c	0,0 %	21,4 КБ	24,0 КБ	06.06.2019 19:53:16	A

Сразу замечен минус, что файлы не пронумерованы в таблице для лидер-статистики файлов по их размеру. Зато программа работает быстро даже при миллионе элементов на жёстком диске.

# Спасибо за внимание!