

プログラミング実践 1

最終レポート

TrashAndArchives(ゴミ箱)

工学部 電気電子・情報工学科 情報コース 2 年

1183033122 (x3033122)

長谷川智大

Trash And Archives(ゴミ箱)

「TrashAndArhives」という、オリジナルの Java プログラムを作成した。

作成した経緯

普段、データを削除するために「ゴミ箱」を使っている。しかし、このゴミ箱はただデータを処理して削除しているだけで、あまり面白みがなかった。「実用性」という意味では完璧だが、私はこの「削除する行為」をより体験的にすることで、削除行為について考えさせるプログラムを作りたいと考えた。

「削除をしている」という感覚が感じられるものなら、誤った行動を抑制でき、よりよい回答が見つかるのではと思い、演出を追加することでユーザーに一度、「削除していること」を考えさせることができればと考えた。

また一方で、とある作品の「データを削除する」様子を、自分で実現してみたいとも考えていたため、このプログラムを作った。

プログラム概要

プログラムファイルの絶対パス

/home/students/x3033122/Lecture/19_ProgJissen1/FinMyApplication/Trash_Archive/TrashAndArchives.java

上記ファイルにコメントで、変数と具体的処理を記述している。

「即削除」と「アーカイブ」の 2 つのモードを持つ。

「即削除」は、ファイルをドロップするだけでデータを削除できるようにしている。また、ドロップされたファイル名を表示し、少しずつ消えていく演出がある。

「アーカイブ」は、ファイルをドロップすると、ディレクトリ内まで調べ、最大で 100 ファイルまでを表示できる。そして、そのファイル名をクリックすると、テキスト・画像ファイルなら左のプレビュー画面で閲覧することができる。また、削除するときは 4 回確認を行い、削除する際も演出の後、再度確認させてからデータを実際に削除するようなプログラムになっている。

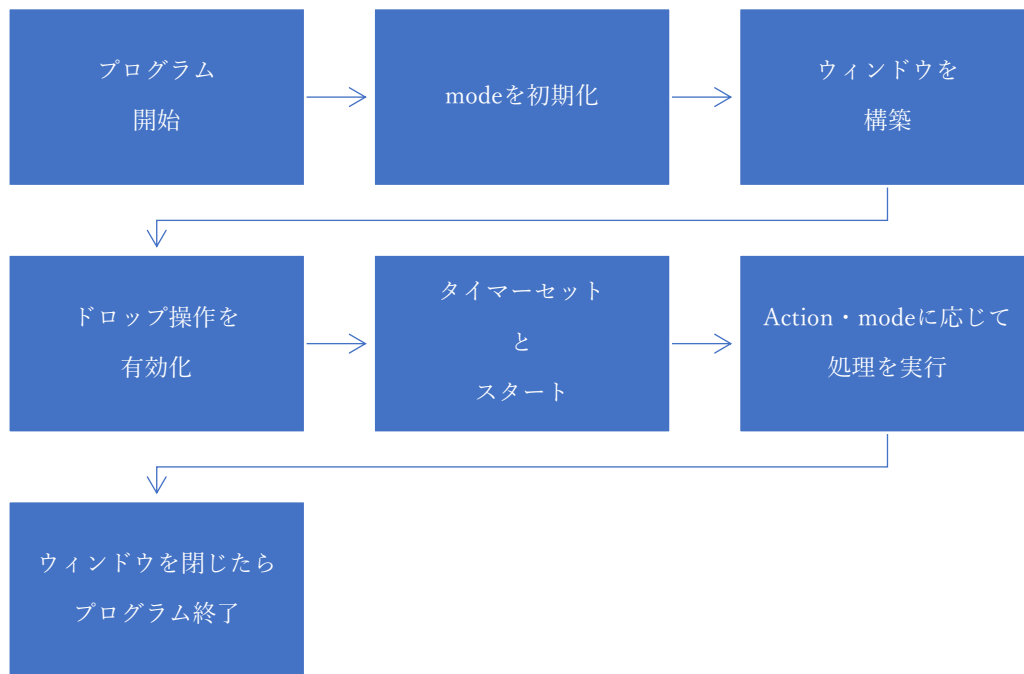
いずれの場合も、「演出だけ見る」ことができる。このボタンをクリックすることで、モードが切り替わり、実際に削除しなくなる。

プログラム記述内容のうち、ドロップ操作とファイルインポートに関するプログラムは、参考サイトを大いに参考にした。一部改変しただけなので、その部分が完全なオリジナルではない。

工夫点と苦労点はフローチャート後に記載してある。

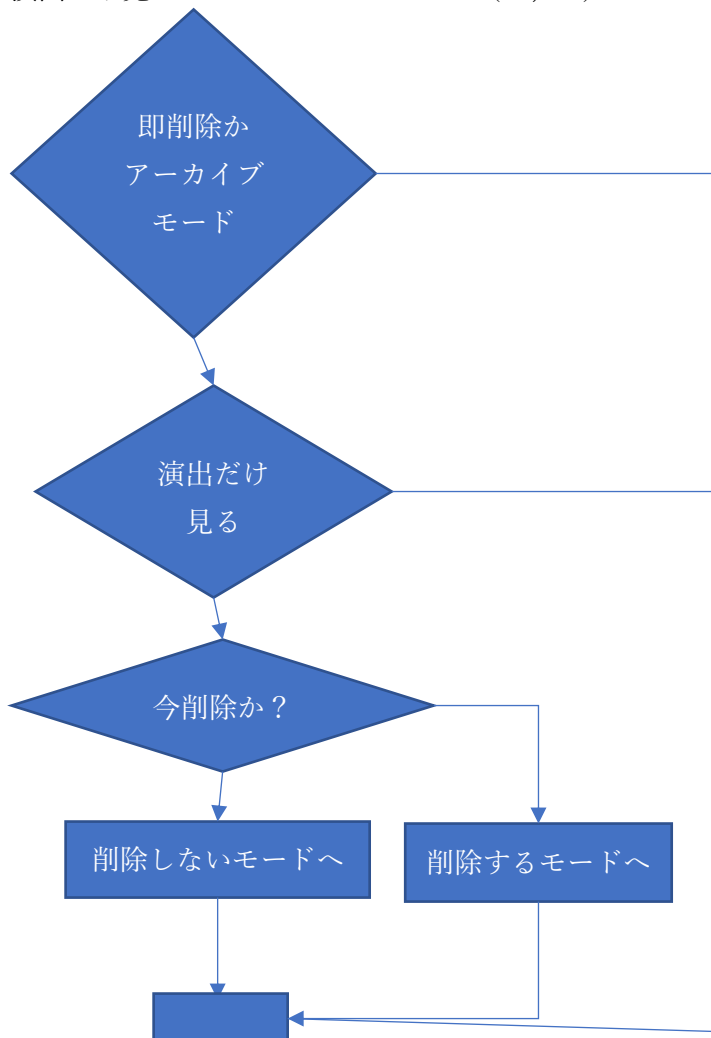
よく使うような項目は関数化してあり（ボタンをセッティング、パネルをセッティング、ファイル削除等）、それらを再帰的に呼び出すことで、フォルダ内のフォルダも処理できるようにしてある。

フローチャート

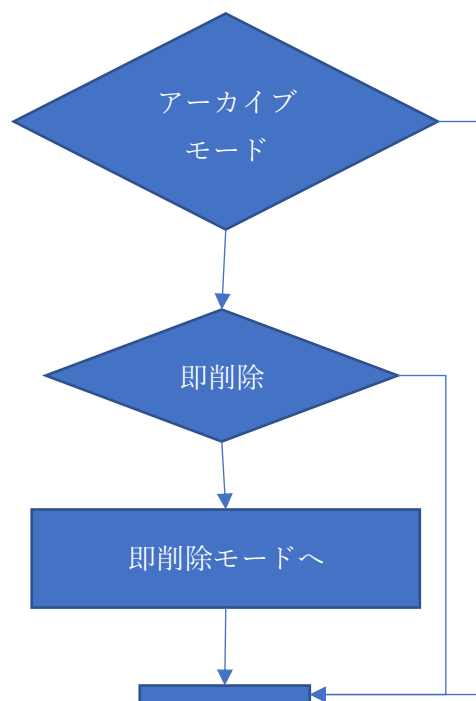


<実際の処理>

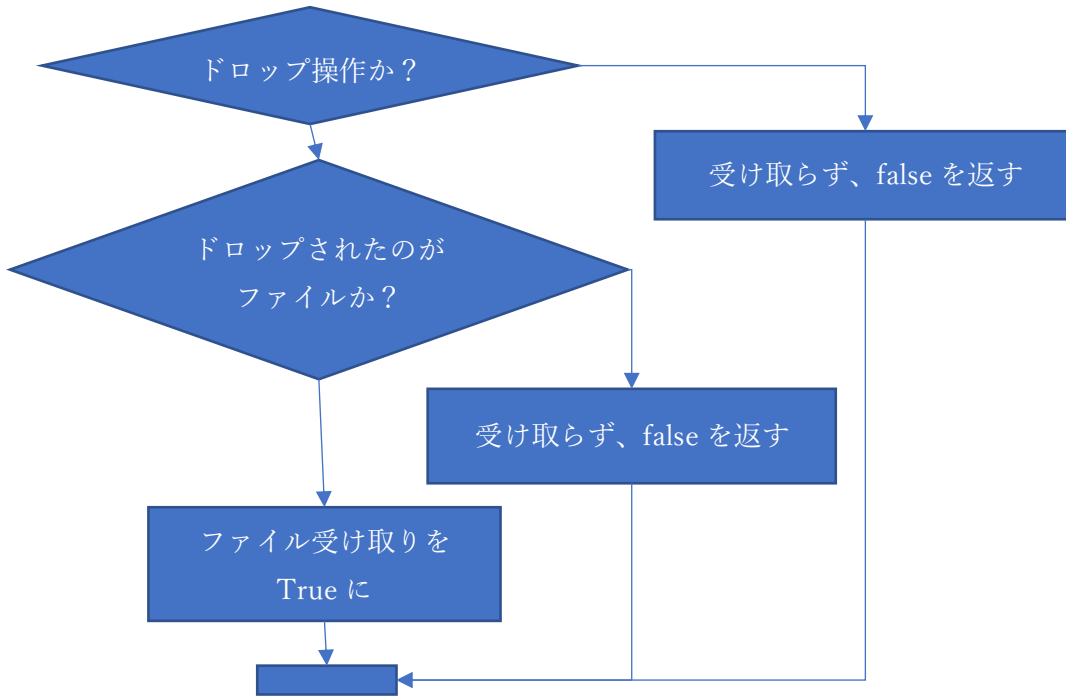
演出だけ見るボタンのフローチャート (10, 0)



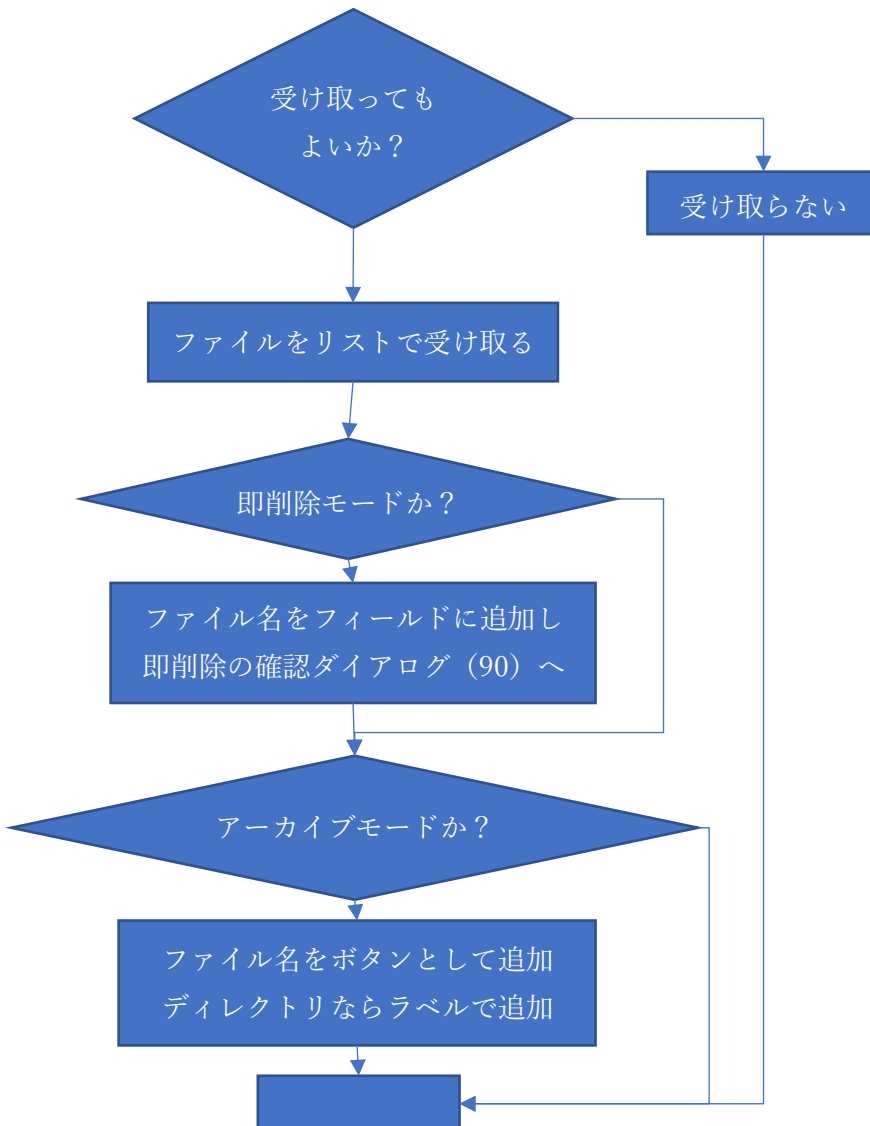
モード変更のフローチャート (即削除も同様、10, 0)



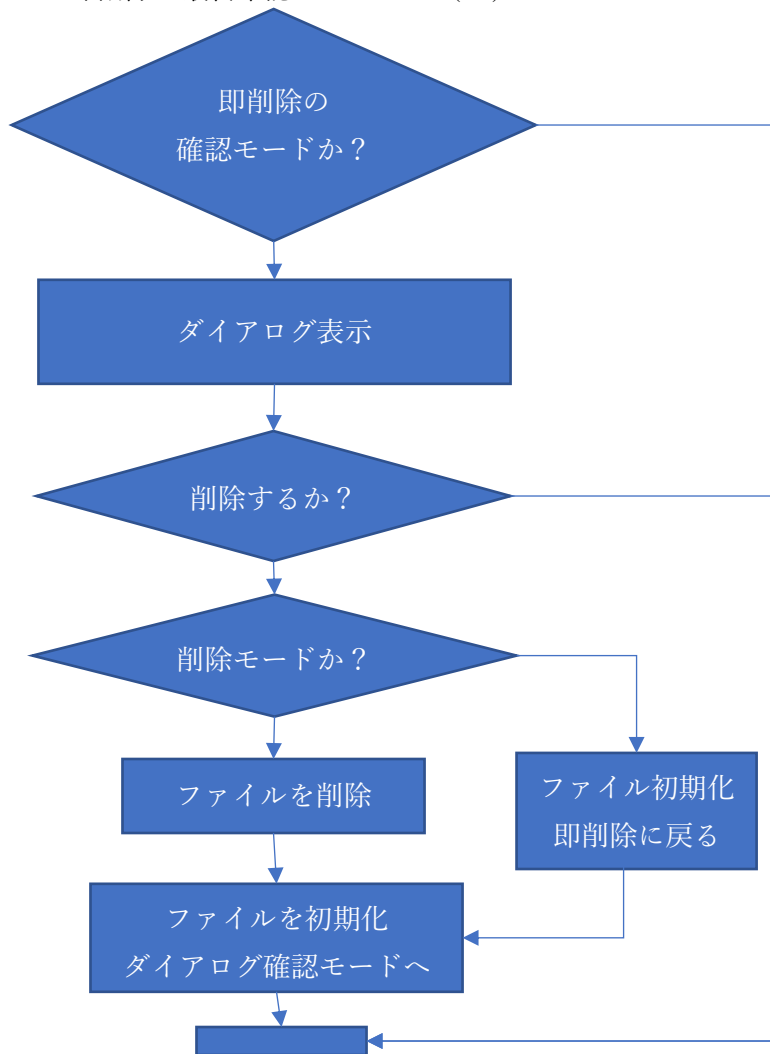
ドロップ操作



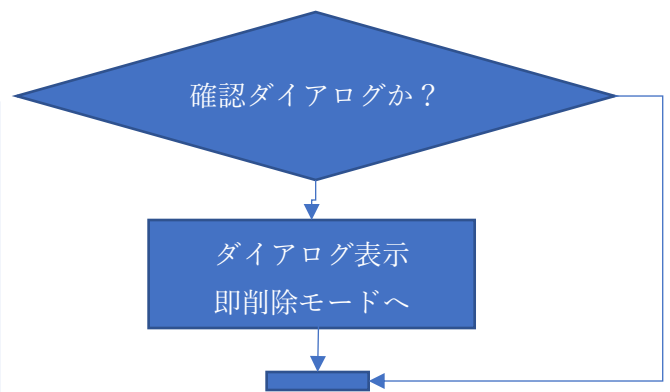
ファイルインポート



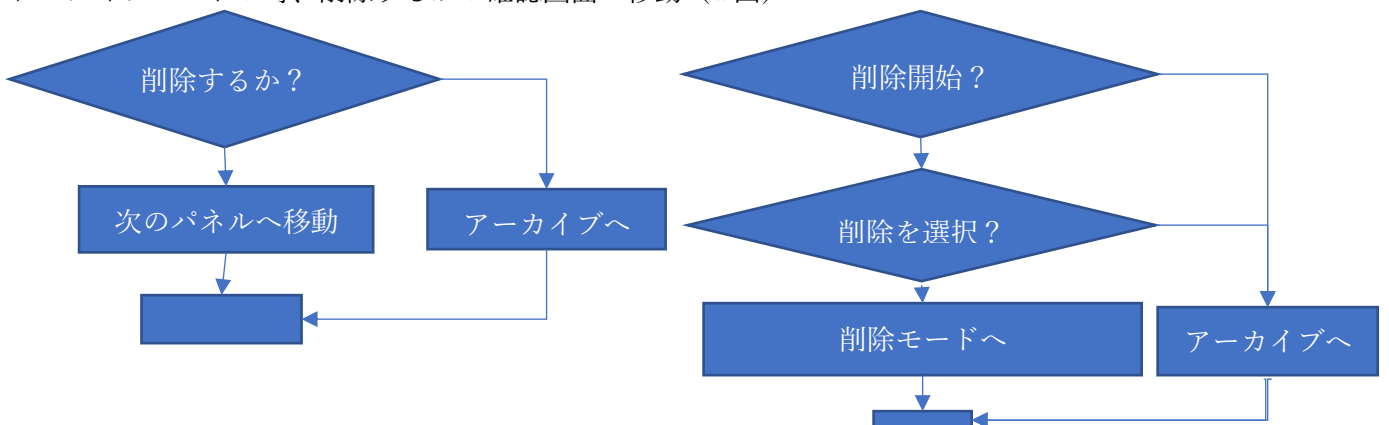
即削除の最終確認ダイアログ (90)



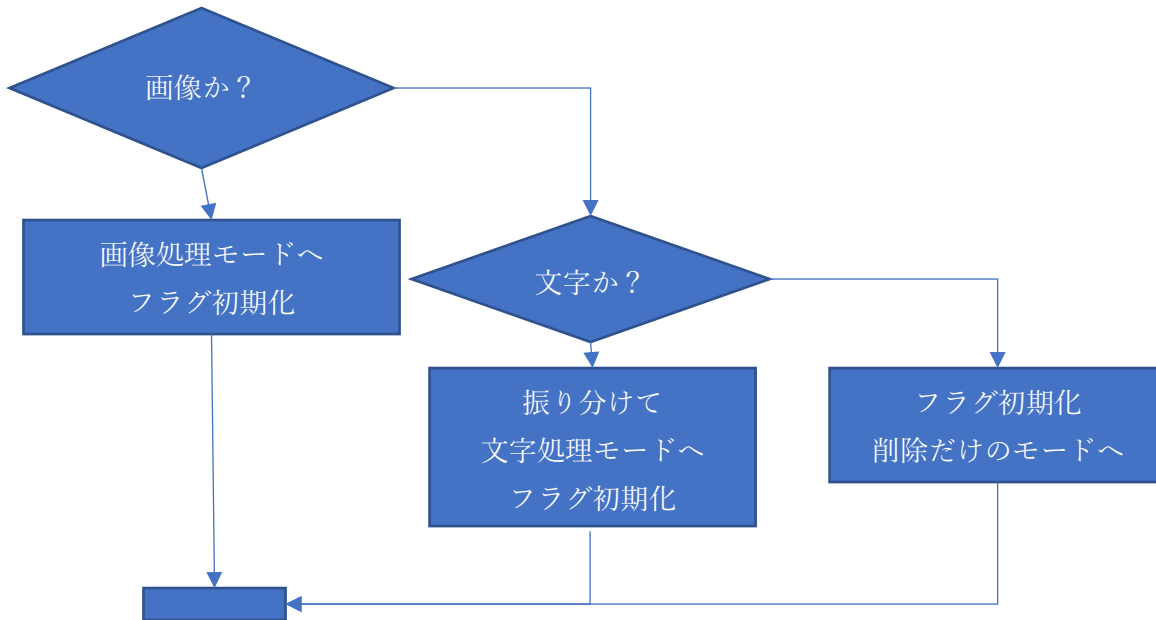
即削除の削除後ダイアログ (5)



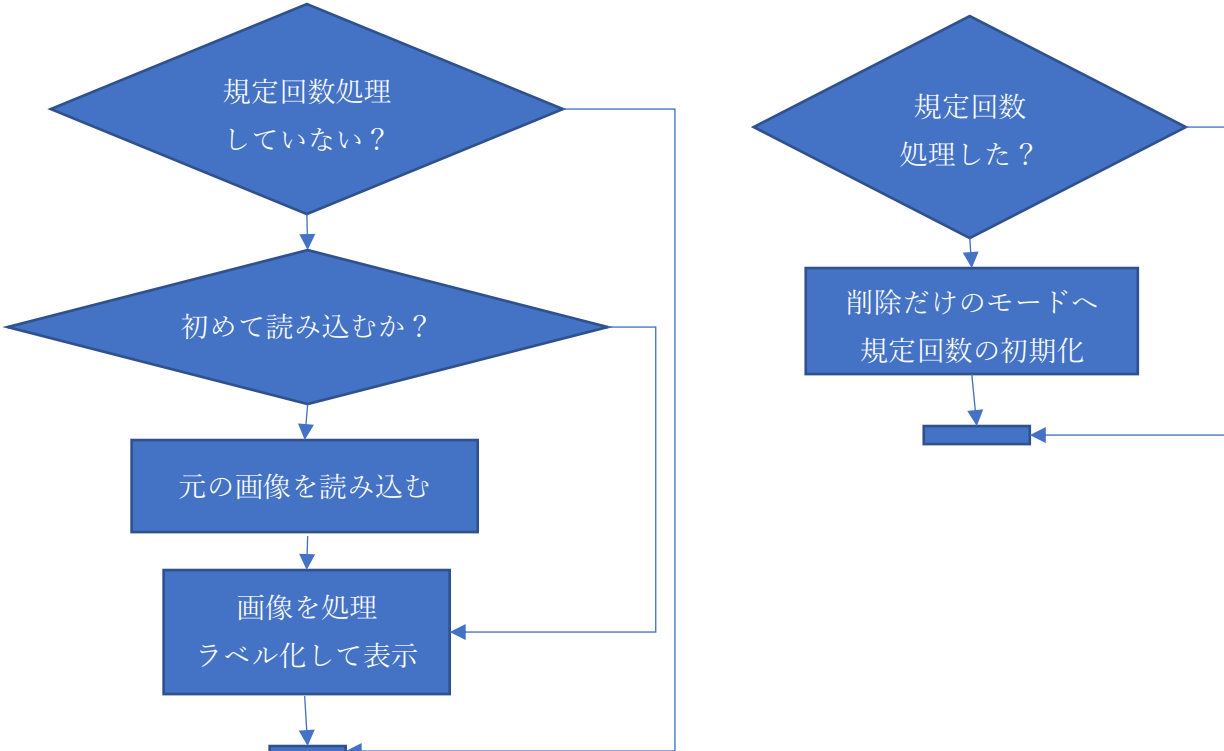
アーカイブモードの時、削除するか確認画面に移動 (4 回)



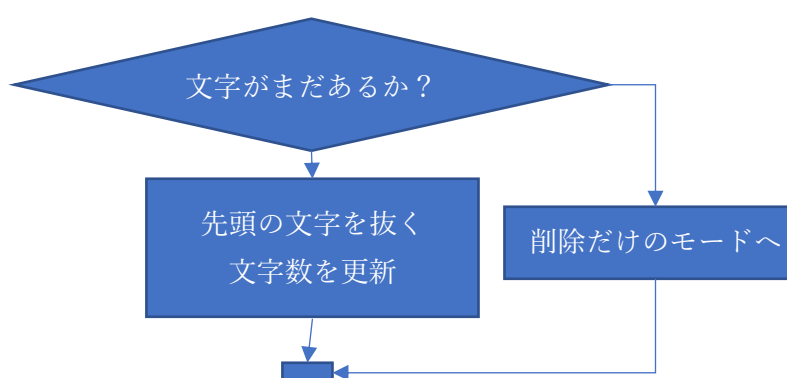
削除モードの具体的選択 (40)



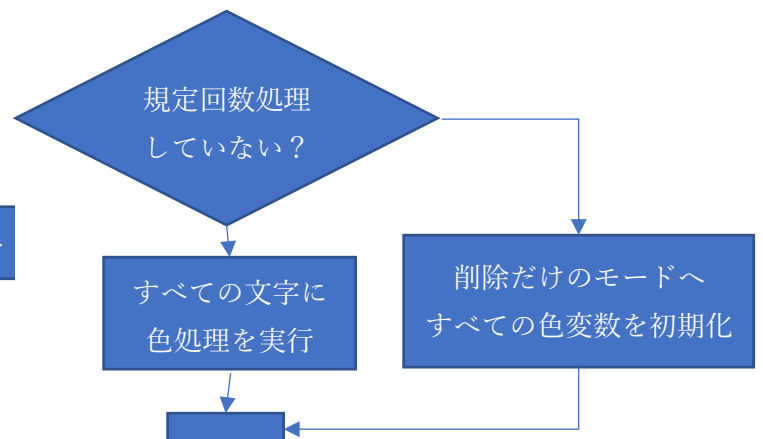
画像の処理モード (50)



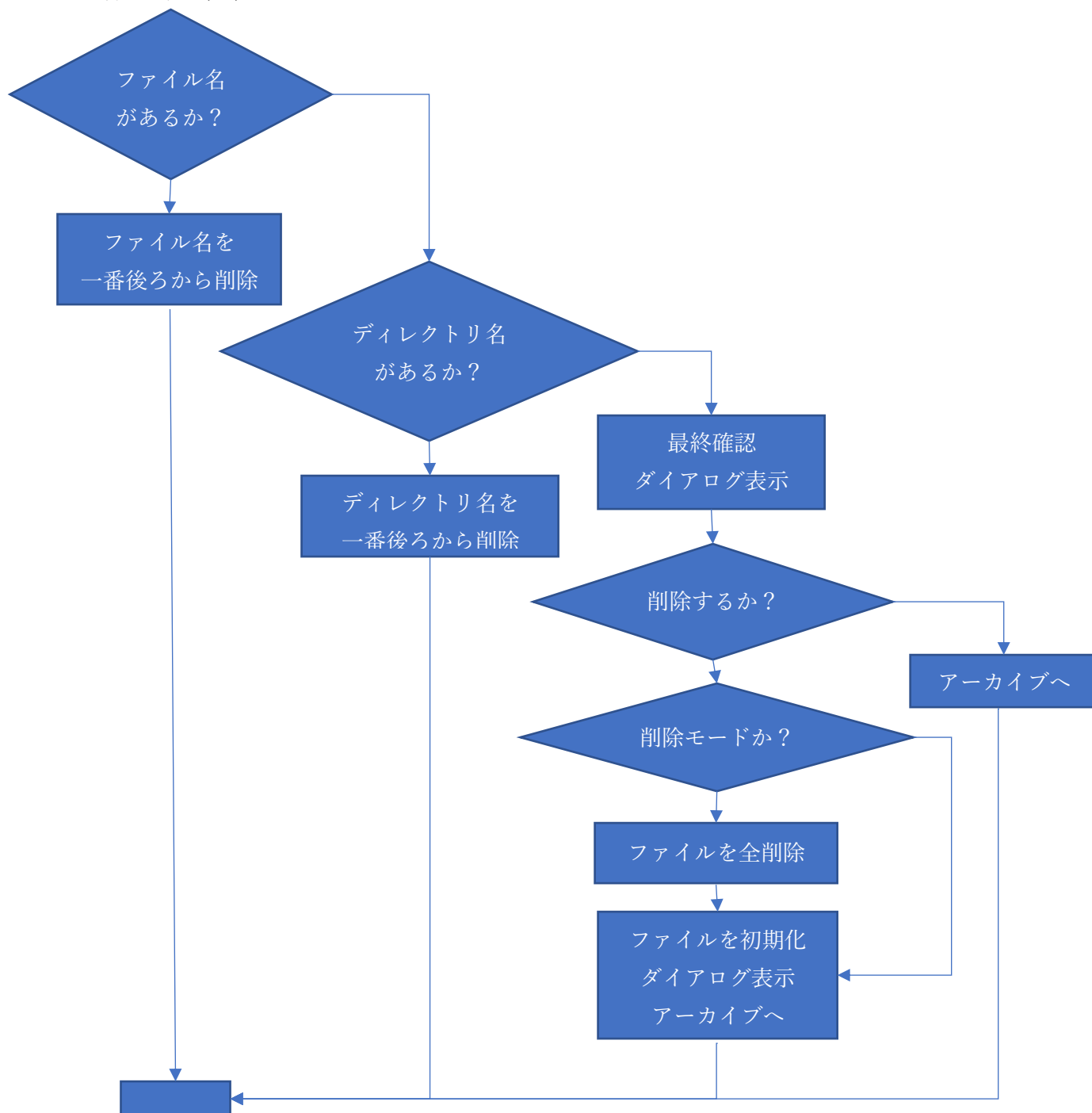
文字処理モード 1 (62)



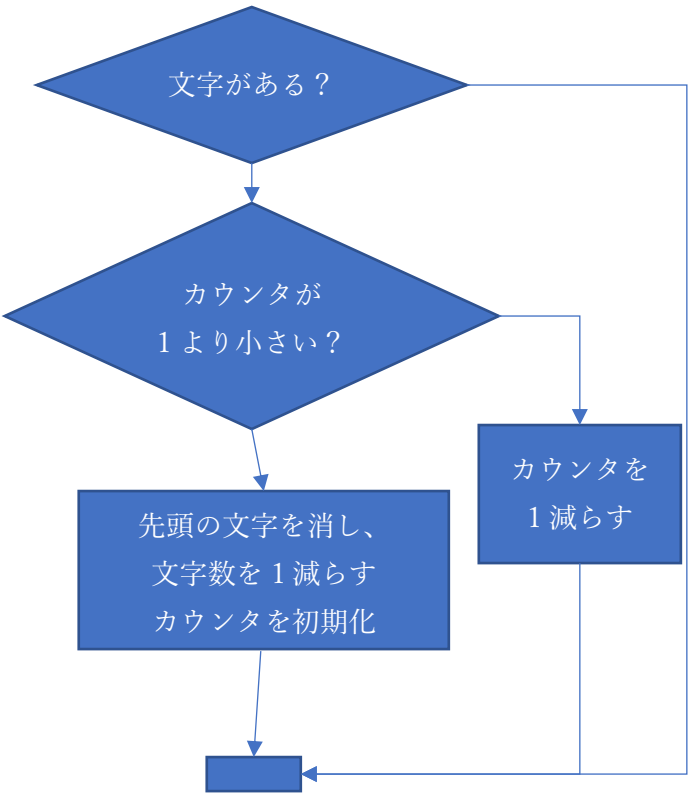
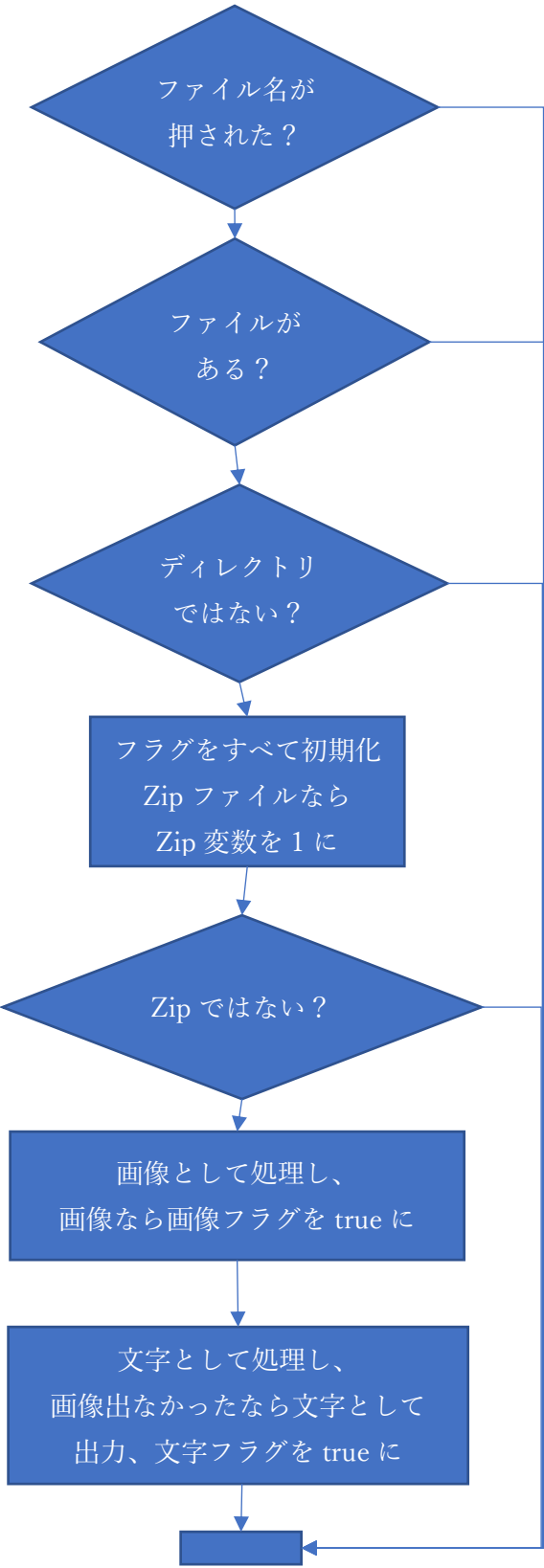
文字処理モード 2 (60)



ファイル名を削除 (20)



即削除パネルに文字があったら、削除していく



各クラスの説明

TrashAndArchives	メインクラス。処理をすべて記述している。
PaintCanvas	文字・画像描画用のキャンバスクラス。文字表示を使用した。
DropFileHandler	ファイルのドラッグアンドドロップを有効にするクラス。 このクラスは参考サイトを参考に作った。

使用方法と実行画面の画像

ファイルを実行すると以下の画面を表示する。
デフォルトで「即削除/演出は見るがデータも削除する」モードになっている。



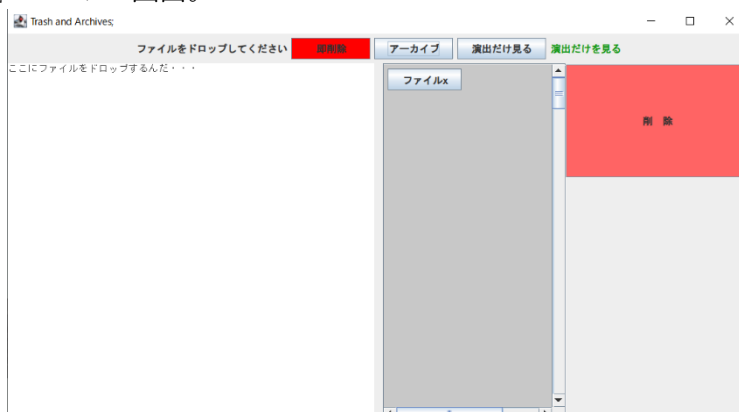
画面の「演出だけ見る」を押すと、演出だけを見られるようにして、データを削除しなくなる。



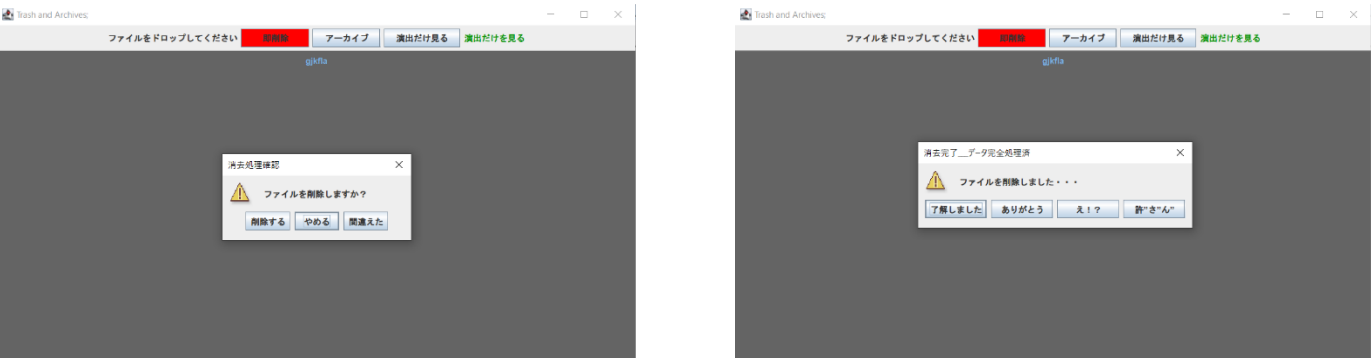
以下からは、演出だけを見るモードで実行する。(データを削除するときにはもう一度「演出だけ見る」を押す)

「即削除」/「アーカイブ」でモードを変更できる。上記は「即削除」モード。

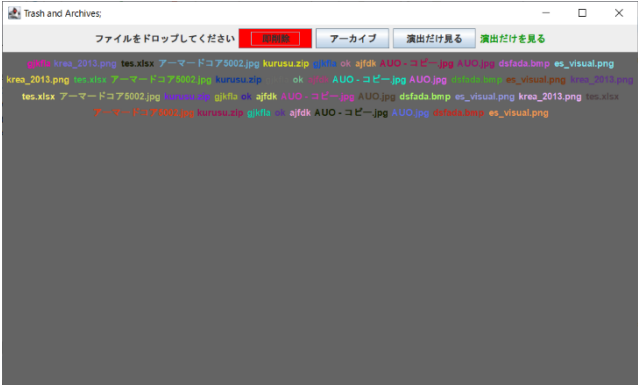
以下は「アーカイブ」モードの画面。



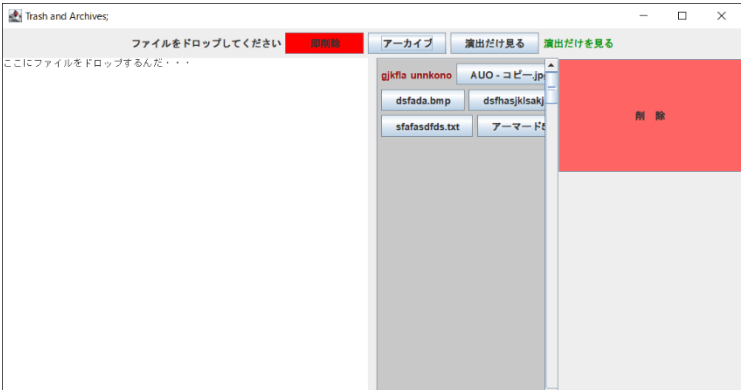
どちらのモードでも、ファイルを画面にドラッグアンドドロップすることで、ファイルを受け取れる。
「即削除」では、以下のダイアログが表示される。



また、ファイル名を色とりどりに表示し、数秒で消えるようにしている。



「アーカイブ」では、画面が更新され、以下のようにファイル名はボタンに、フォルダ名は文字列で表示される。



画像ファイルを押すと、左に画像がサイズそのまま表示される。



テキストファイルを押すと、左に文字列が表示される。



右側の「削除」を押すと、確認ウィンドウが表示される。

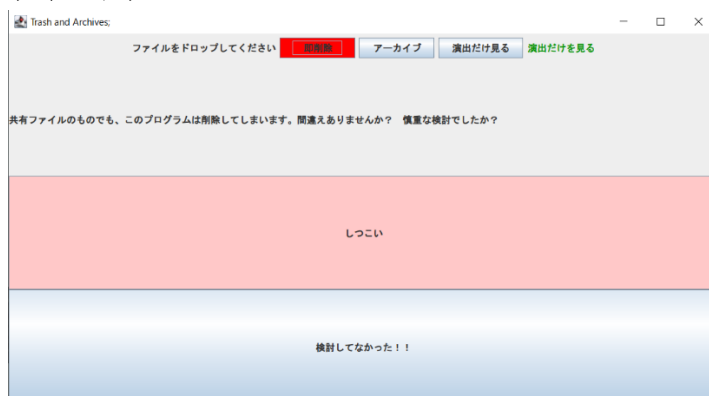
ウィンドウ 1



ウィンドウ 2



ウィンドウ 3



ウィンドウ 4

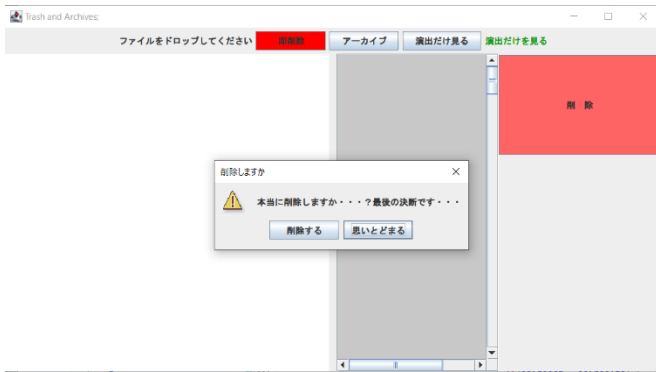


ウィンドウ 4 までいったところで、「削除開始」を押すと、以下のダイアログが表示される。
(下を選ぶと元の画面に戻る。)



「削除する」を選ぶと、画像なら白くなっていき、文字なら薄くなるか、先頭から取り除かれていく。そして、ファイル名・フォルダ名が消えていき、以下のダイアログが表示される。最終確認を行い、データを実際に削除する。

(削除モードの時は、ファイルが削除される)



工夫した点・苦労した点

- ・ GUI を随時更新する。

ファイルの削除や、画像のフィルタかけには、初めスレッドを使おうとしていた。

しかし、スレッドを止めて実行しても、UI に変化がなく、最後の処理で初めて UI が変化したため、うまくいかなかった。

そこで、タイマーを利用した。タイマーはその時間ごとに `ActionPerformed` を呼び出して実行するので、上記の処理を実行するのに最適だった。

- ・ ファイルをドロップで受け取る。

Java にドロップ操作の API があることを知るまでは、ダイアログで選択していた。

しかし、ダイアログ選択では利便性に欠けるため、調べてドロップ操作ができるようになった。

- ・ 受け取ったファイルの処理

画像と文字をそれぞれ処理する際、文字に先に代入してしまうとエラーが発生したため、画像でないか確かめしてから、文字テキストとして処理するようにした。

また、Zip ファイルを受け取った時も、プログラムが停止してしまうため、Zip か確認し、Zip なら処理をしないようにした。

加えて、インデックスを超えてしまうような数のファイルを表示しようとしていたため、上限を設定することでエラーを回避した。ファイルのボタンや、テキスト数などで、上限を超えないようにした。

- ・ ファイルを誤って削除してしまうことについての対処。

誤ってデータを消してしまう経験から、何があろうとも 4 回以上は確認させる必要性を感じた。

ただのごみ箱なら、確認は 1 度きりで、間違えて消してしまうかもしれない、大事なフォルダを消してしまう危険性もある。よって、確認はいかなる場合でも必要と考えた。

追加したかった項目

- ・ Zip 化による保存

Java に、Zip 化の機能はあったが、ファイルを Zip 化することができなかった。ファイルを選択し、出力先を設定して、1064 バイト形式で保存することで Zip 化できるが、List でファイルを管理していたためか、ファイルを Zip 化できず、実装しなかった。

- ・ Zip を解凍し、ファイルとして出力して表示する

上記のような Zip 化ができなかったため、Zip 解凍も実装できなかった。

- ・ 音楽 (WAV 等)、動画の処理

ファイルとして削除することはできるが、プレビューができない。また、音楽に関しては音を流せない。プレビューできる対応ファイルは、テキスト形式と画像の 2 つ。

ただ、プレビューのプログラムは拡張性を意識した構成にしてあるため、上記のファイルの処理ができれば、プログラムの対応ファイルを増やしやすいく考える。

- ・ PDF ファイルの読み込み不調

PDF ファイルを読み込むとエラーが発生する。上記同様に、ファイルが対応していないが、テキストとして処理しているため、オーバーフローなどのエラーが発生し、プログラムが停止すると考えられる。

感想

具体的なアプリのプログラムを作ることは、おそらく今回が初めてで、また Java も初めて使ったので、多くのことで躓いた。例えば、画像・文字の扱い、ファイルの解放、ドラッグアンドドロップ処理、GUI の処理などが挙げられる。

ファイルを削除するプログラムだったため、誤った操作によりファイルを削除することの恐怖と、対応策の思案をする機会があり、セキュリティとセーフティの重要性を知った。プログラムは、ファイルを操作することが多く、ファイルを使わない独立したプログラムであっても、コンフィグファイルを使ってパラメータを設定できるようにすることもある。ファイルの取り扱いには十分注意したい。

今回のプログラム制作で、Java のファイル管理と、画像処理、GUI 処理が最も学べた点と思う。ファイル操作は上記に述べた通りで、画像処理については、表示と色付け、画像ファイルの解放処理に重点を置いた。GUI 処理については、GUI の詳しい設定方法と、再度表示させる `updateUI();` 処理に重点を置いた。いずれの点も、今の情報社会において重要な項目のため、今後も学んでいきたい。

参考プログラムの場所 (サイト名とか)

ドラッグアンドドロップの参考サイト

<https://saku-java.be-ourselves.jp/2013/12/16/drag-and-drop-to-get-files/>

Java 全般の参考

<https://www.javadrive.jp/>

ソースコードの場所は共有フォルダ、
jissen1/2019/x3033122_program

または

/home/students/x3033122/Lecture/19_ProgJissen1/FinMyApplication/Trash_Archive/TrashAndArchives.java

の自分のファイルに入っている。