Search\_file\_for\_string:

**Search\_file\_for\_string(path, key)**

**Equivalence Classes**

Search successful –

• Any valid file path

• Any key

Error –

• Any invalid file path

• Inputs not strings

**Test Scenarios**

Valid File Path:

Parameters: “C:\\Users\\Rodge\\Desktop\\New Text Document.txt”, “this”

Outcome: display context and line number

Valid File Path:

Parameters: “C:\\Users\\Rodge\\Desktop\\New Text Document.txt”, “”

Outcome: displays entire file separated according to line number

Invalid File Path:

Parameters: “C:\\Users\\Rodge\\Desktop\\”, “this”

Outcome: Error, no permission

Invalid File Path:

Parameters: “awkfawoebflaoweufb”, “this”

Outcome: Error, no such file or directory

Inputs not strings:

Parameters: [“C:\\Users\\Rodge\\Desktop\\New Text Document.txt”], “this”

Outcome: Error, type mismatch

Foreach\_file:

**foreach\_file(func, terminate\_early, include\_paths, include\_exts, exclude\_paths)**

**Equivalence Classes**

Error -

• Wrong number of parameters or wrong parameter types

• Invalid include\_paths

• Invalid exclude\_paths

• func parameter is given a function with more or less than one parameter

Each file matching the criteria called -

• valid include\_paths and exclude\_paths

• terminate\_early parameter is [False]

**Test Scenarios**

func parameter is given a function with more or less than one parameter:

Parameters: func parameter is given a function with more or less than one parameter

Outcome: type error, wrong argument number

Invalid include\_paths:

Parameters: include\_paths set to [“ergergsergrgser”]

Outcome: file path not found

Invalid exclude\_paths:

Parameters: exclude\_paths set to [“ergergsergrgser”]

Outcome: file path not found

valid include\_paths and exclude\_paths + terminate\_early parameter is [False]:

Parameters: include\_paths set to [“C:\\Users\\Rodge\\Desktop\\”], exclude\_paths set to [“C:\\Users\\Rodge\\Desktop\\ExcludeFolder”], include\_exts set to [“.txt”], terminate\_early set to [False]

Outcome: calls the function on every .txt file in the include\_paths, and excluding the exclude\_paths directory

Search\_for\_string:

**Search\_for\_string(result\_callback, finished\_callback, terminate\_search, key, include\_paths, include\_exts, exclude\_paths):**

**Equivalence Classes**

Error –

• result\_call back not a function with two inputs

• finished\_callback not a function with no input

• terminate\_search not a single element boolean list

• include\_paths not list of valid path(s)

• exclude\_paths not list of valid path(s)

• mismatched parameters

Successful Search -

• result\_call back a function with two inputs

• finished\_callback a function with no inputs

• terminate\_search is a single element list with either True or False

• include\_paths has list of valid path(s)

• exclude\_paths has list of valid path(s)

Exclude paths -

• Set exclude\_paths to valid path(s)

• No exclude paths

Extensions -

• Extension included

• No extension

**Test Scenarios**

result\_call back not a function with two inputs:

Parameters: result\_call back doesn’t take two inputs

Outcome: Argument mismatch

finished\_callback not a function with no input:

Parameters: finished\_callback not a function with no input

Outcome: Argument mismatch

terminate\_search not a single element boolean list:

Parameters: terminate\_search not a single element boolean list

Outcome: Doesn’t search

include\_paths not list of valid path(s):

Parameters: include\_paths set to ["SJfbsefe"], include\_exts set to [“.txt”], key set to “this”

Outcome: File path not found

exclude\_paths not list of valid path(s):

Parameters: include\_paths set to ["C:\\Users\\Rodge\\Desktop\\"], exclude\_paths set to ["awefbawekfba"], key set to “this”

Outcome: File path not found

mismatched parameters:

Parameters: missing required parameters

Outcome: argument error

result\_call back a function with two inputs:

Parameters: result\_call back is a function with two parameters

Outcome: Calls the function on every search hit

finished\_callback a function with no inputs:

Parameters: finished\_callback a function with no inputs

Outcome: Calls the function when search is finished

terminate\_search is a single element list with either True or False:

Parameters: terminate\_search is [False]

Outcome: Searches with no issues

terminate\_search is a single element list with either True or False:

Parameters: terminate\_search is [True]

Outcome: Search terminates early

include\_paths has list of valid path(s) + Extension included:

Parameters: include\_paths set to ["C:\\Users\\Rodge\\Desktop\\"], include\_exts set to [“.txt”], key set to “this”

Outcome: Search successfully returns with only .txt files being searched

include\_paths has list of valid path(s) + No extension:

Parameters: include\_paths set to ["C:\\Users\\Rodge\\Desktop\\"], key set to “this”

Outcome: Search successfully returns

include\_paths has list of valid path(s) + exclude\_paths has list of valid path(s):

Parameters: include\_paths set to ["C:\\Users\\Rodge\\Desktop\\"], exclude\_paths set to ["C:\\Users\\Rodge\\Desktop\\DummyDirectory"], key set to “this”

Outcome: Search successfully returns with the exclude path directory not searched

PkeAppWindow + GUI DoCs:

**Equivalence Classes:**

The equivalence classes for the GUI are too thorough (growing exponentially) to list them all. There are 5 text boxes to record inputs, and based upon the varied entrants to these boxes, a multitude of outputs can be expected.

**Test Cases:**

If no path is included:

1. Start the app
2. Enter the following strings to the respective text fields:
   1. Search For: “sys”
   2. Path(s) Included: “”
   3. Path(s) Excluded: “”
   4. Extension(s) Included: “py”
   5. Editor Path: “”
3. Press “Start Search” Button

(should see a message added “no file paths included in search”)

If no search term is included:

1. Start the app
2. Enter the following strings to the respective text fields:
   1. Search For: “”
   2. Path(s) Included: “GUI.py” (\*when in the code directory)
   3. Path(s) Excluded: “”
   4. Extension(s) Included: “py”
   5. Editor Path: “”
3. Press “Start Search” Button

(should see a message added “search bar is empty”)

Included path is also an entry in ‘paths to exclude’:

1. Start the app
2. Enter the following strings to the respective text fields:
   1. Search For: “sys”
   2. Path(s) Included: “GUI.py” (\*when in the code directory)
   3. Path(s) Excluded: “GUI.py”
   4. Extension(s) Included: “py”
   5. Editor Path: “”
3. Press “Start Search” Button

(should see results; paths included trumps paths excluded)

Clear results when results is already empty:

1. Start the app
2. Enter the following strings to the respective text fields:
   1. Search For: “”
   2. Path(s) Included: “”
   3. Path(s) Excluded: “”
   4. Extension(s) Included: “”
   5. Editor Path: “”
3. Press “Clear Search” Button

(nothing should happen)

For the extension input, if multiple file extensions are separated by a comma and a space, the later extensions no longer apply:

1. Start the app
2. Enter the following strings to the respective text fields:
   1. Search For: “3.6”
   2. Path(s) Included: “GUI.py” (\*when in the code directory)
   3. Path(s) Excluded: “”
   4. Extension(s) Included: “py, md”
   5. Editor Path: “”
3. Press “Start Search” Button

(the readme will not have any hits)

Text Editor Module:

**Equivalence Classes**

Editor Set -

• Valid full file path to a .exe file

Editor Not Set -

• file path not to a .exe file

• invalid file path

Opening File -

• Search is valid

• Search is invalid

• Current editor is not actually an editor

Editor Status -

• File name too long

• File name fits

**Test Scenarios**

Valid full file path to a .exe file:

Editor Path Parameters: C:\WINDOWS\system32\notepad.exe

Outcome: Editor is set to notepad.exe and displayed

file path not to a .exe file:

Editor Path Parameters: C:\Users\Rodge\Desktop\test.py

Outcome: Editor not set, input ignored

invalid file path:

Editor Path Parameters: aowebfoawiefe

Outcome: Editor not set, input ignored

Search is valid:

Parameters: search results valid

Outcome: each search result button opens to current editor, if set, otherwise does nothing

Search is invalid:

Parameters: search results invalid

Outcome: button in search result window doesn’t do anything

Current editor is not actually an editor:

Editor Path Parameters: C:\Users\Rodge\Desktop\NotAnEditor.exe

Outcome: Depending on the .exe, may crash, or just open the .exe without anything happening

File name too long:

Editor Path Parameters: C:\Users\Rodge\Desktop\ThisIsAVeryLongEditorName.exe

Outcome: the current editor label gets cut off

File name fits:

Editor Path Parameters: C:\Users\Rodge\Desktop\Editor.exe

Outcome: the current editor shows correctly

Search Term Too Broad:

**Equivalence Classes**

Crash-

• Search term too general

• No extensions

• No exclude paths

• Search directory too big

Success-

• Search parameters more specific instead of in general

**Test Scenarios**

**Search term too general/No extensions:**

**Parameters:**

Search Term: a

Include Path: C:\Users\Rodge\Desktop

Exclude Path: none

Extension: none

**Outcome:**

Program Crashes

**Search directory too big:**

**Parameters:**

Search Term: this

Include Path: C:\

Exclude Path: none

Extension: txt

**Outcome:**

Program Crashes

**Search parameters more specific instead of in general:**

**Parameters:**

Search Term: this

Include Path: C:\Users\Rodge\Desktop

Exclude Path: none

Extension: txt

**Outcome:**

Program Finishes

**Search parameters more specific instead of in general:**

**Parameters:**

Search Term: this

Include Path: C:\Users\Rodge\Desktop\DummyFolder

Exclude Path: none

Extension: none

**Outcome:**

Program Finishes

**Search parameters more specific instead of in general:**

**Parameters:**

Search Term: this

Include Path: C:\Users\Rodge\Desktop\

Exclude Path: C:\Users\Rodge\Desktop\ToExclude

Extension: none

**Outcome:**

Program Finishes