Coverage

This README file describe how to collect coverage information using our tool.

We have three stages to get fuzzing performance results.

Stage 1: use the target program to open test PDFs and record coverage information by DynamoRIO. It will take about 48 hours.

Stage 2: parse the coverage information and split them to belonging modules(executable files). It will take several hours.

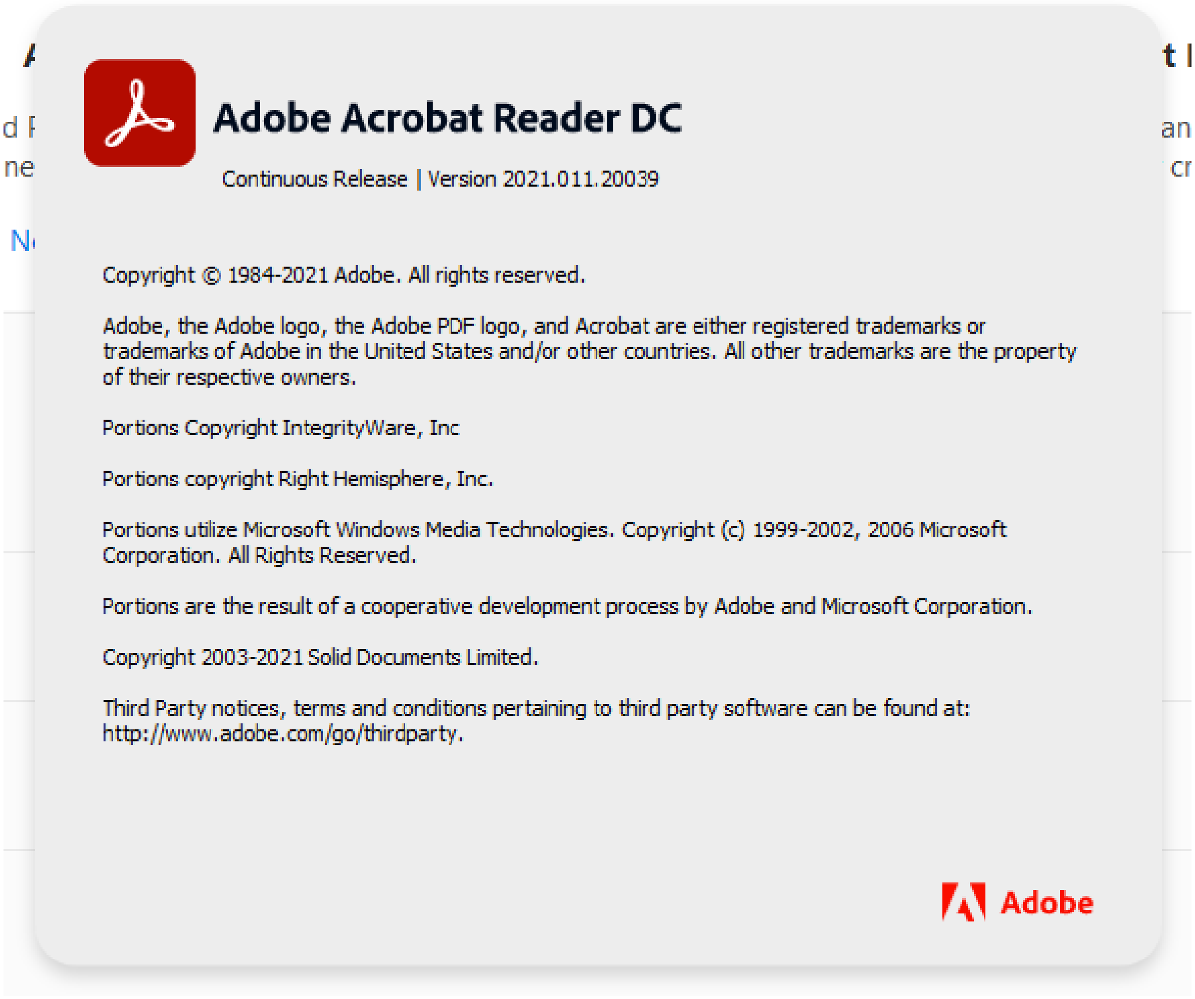
Stage 3: count instruction numbers of every module and add them together. It will take 10-20 minutes.

The following will talk about reproduction in detail.

`Note:`

`1. for easier reproduction, the provided coverage recording's code is constructed based on Adobe Acrobat Reader version 2021.011.20039 (for Foxit PDF Reader is version 11.2.1.53537), please check the version before reproduction.`

`2. in our experience, the version of Adobe Reader has little influece on the result of coverage recording, we choose this version because it's the newest version when we construct the coverage experiment， this leads to less crash in recording coverage, so we can get more accurate coverage information`



Adobe Acrobat Reader DC

Continuous Release | Version 2021.011.20039

Copyright © 1984-2021 Adobe. All rights reserved.

Adobe, the Adobe logo, the Adobe PDF logo, and Acrobat are either registered trademarks or trademarks of Adobe in the United States and/or other countries. All other trademarks are the property of their respective owners.

Portions Copyright IntegrityWare, Inc

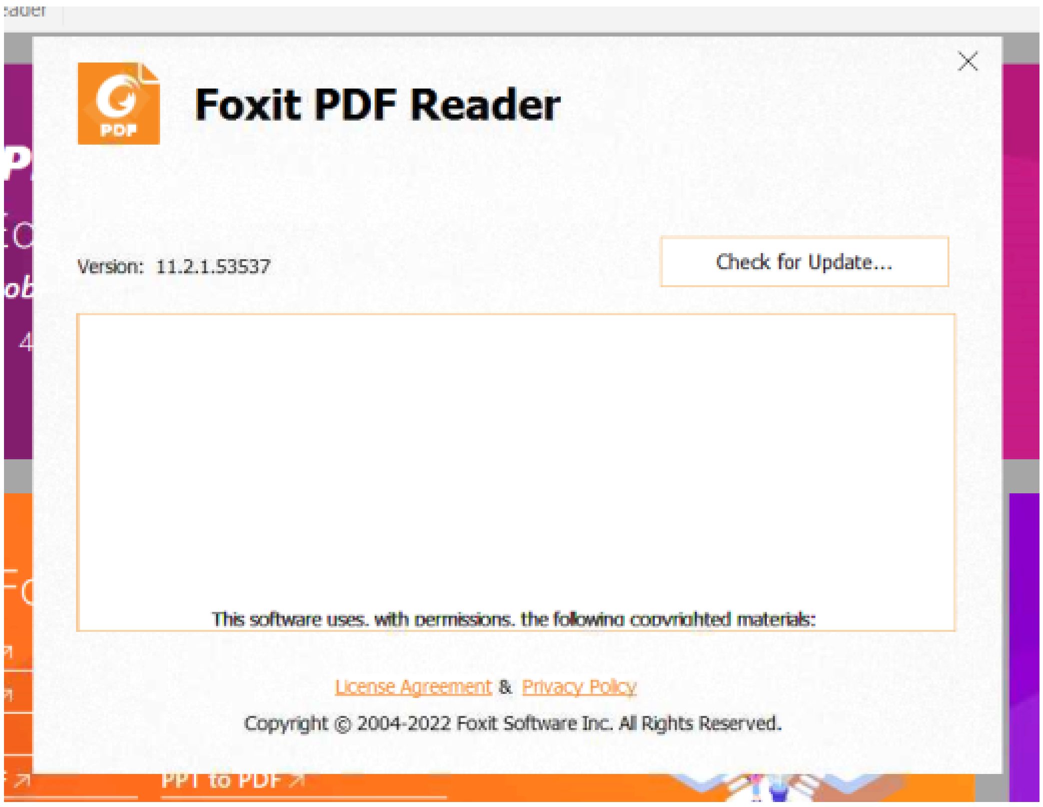
Portions copyright Right Hemisphere, Inc.

Portions utilize Microsoft Windows Media Technologies. Copyright (c) 1999-2002, 2006 Microsoft Corporation. All Rights Reserved.

Portions are the result of a cooperative development process by Adobe and Microsoft Corporation.

Copyright 2003-2021 Solid Documents Limited.

Third Party notices, terms and conditions pertaining to third party software can be found at: http://www.adobe.com/go/thirdparty.



Foxit PDF Reader

Version: 11.2.1.53537

This software uses, with permissions, the following copyrighted materials:

Copyright © 2004-2022 Foxit Software Inc. All Rights Reserved.

Stage 1(record_cov)

At this stage, we use the target PDF Reader to open test PDFs, and use DynamoRIO to record the coverage information

folder structure

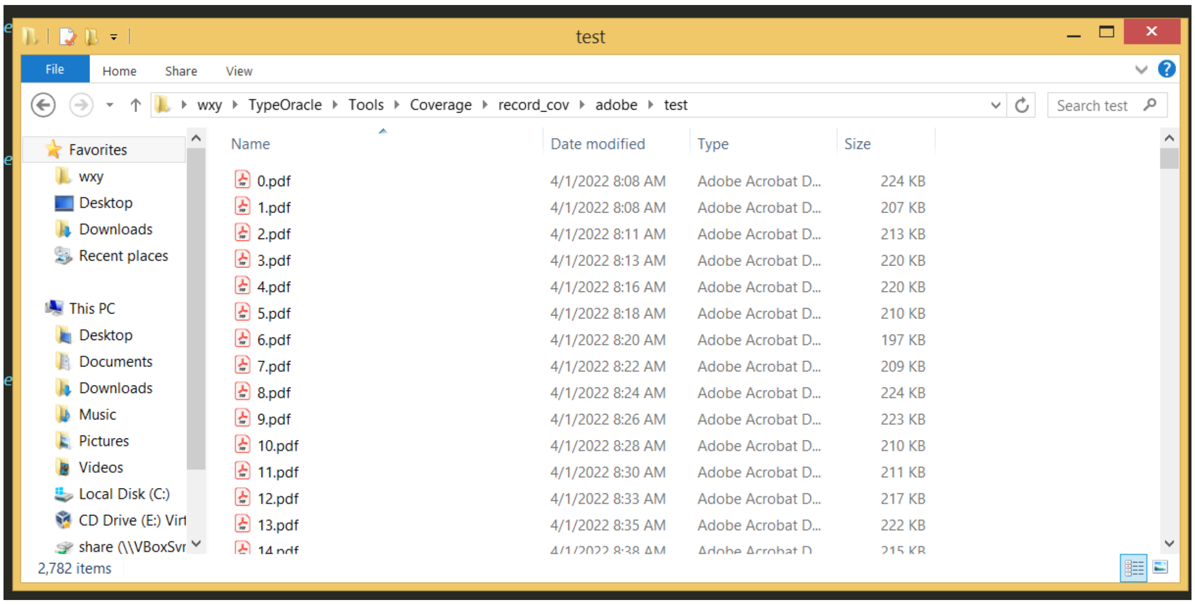
```
- sample_result (the folder store the sample result of this step)

- test (the folder store the test PDFs)

- monitor.py (to minitor the PDF Reader)

- run.py
```

how to reproduce

1. make sure the Page Heap is turned off (execute following command and click yes, for more information about Page Heap, please refer to C:\Users\wxy\TypeOracle\Other\README.pdf)

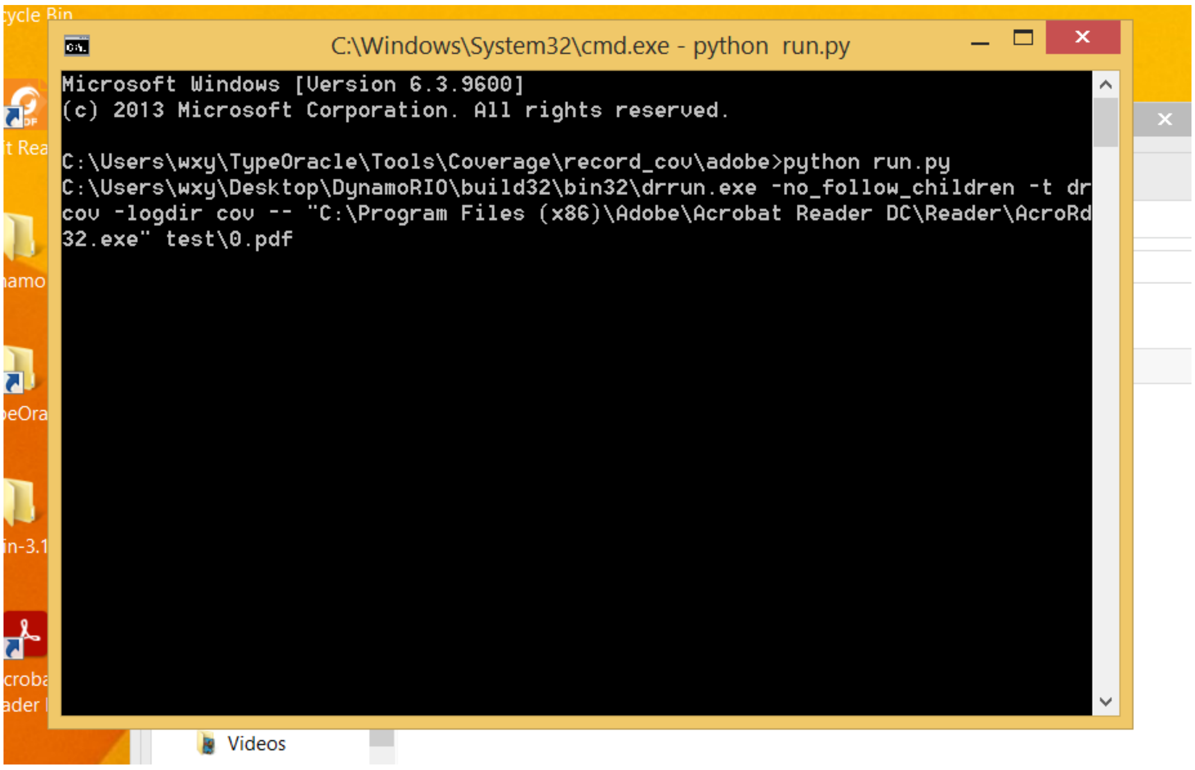
```
"C:\Program Files (x86)\Windows Kits\8.1\Debuggers\x86\gflags.exe" /p /disable
"C:\Program Files (x86)\Adobe\Acrobat Reader DC\Reader\AcroRd32.exe"
```

2. prepare the test PDFs, copy them to the `test` folder



Name	Date modified	Type	Size
0.pdf	4/1/2022 8:08 AM	Adobe Acrobat D...	224 KB
1.pdf	4/1/2022 8:08 AM	Adobe Acrobat D...	207 KB
2.pdf	4/1/2022 8:11 AM	Adobe Acrobat D...	213 KB
3.pdf	4/1/2022 8:13 AM	Adobe Acrobat D...	220 KB
4.pdf	4/1/2022 8:16 AM	Adobe Acrobat D...	220 KB
5.pdf	4/1/2022 8:18 AM	Adobe Acrobat D...	210 KB
6.pdf	4/1/2022 8:20 AM	Adobe Acrobat D...	197 KB
7.pdf	4/1/2022 8:22 AM	Adobe Acrobat D...	209 KB
8.pdf	4/1/2022 8:24 AM	Adobe Acrobat D...	224 KB
9.pdf	4/1/2022 8:26 AM	Adobe Acrobat D...	223 KB
10.pdf	4/1/2022 8:28 AM	Adobe Acrobat D...	210 KB
11.pdf	4/1/2022 8:30 AM	Adobe Acrobat D...	211 KB
12.pdf	4/1/2022 8:33 AM	Adobe Acrobat D...	217 KB
13.pdf	4/1/2022 8:35 AM	Adobe Acrobat D...	222 KB
14.pdf	4/1/2022 8:38 AM	Adobe Acrobat D	215 KB

3. execute run.py to start recording, the recording result will be store in `cov` folder(for uncompressed results) and `db` folder(for compressed results)



```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\wxy\TypeOracle\Tools\Coverage\record_cov\adobe>python run.py
C:\Users\wxy\Desktop\DynamoRIO\build32\bin32\drrun.exe -no_follow_children -t dr
cov -logdir cov -- "C:\Program Files (x86)\Adobe\Acrobat Reader DC\Reader\AcroRd
32.exe" test\0.pdf
```

Stage 2(parse_covfile)

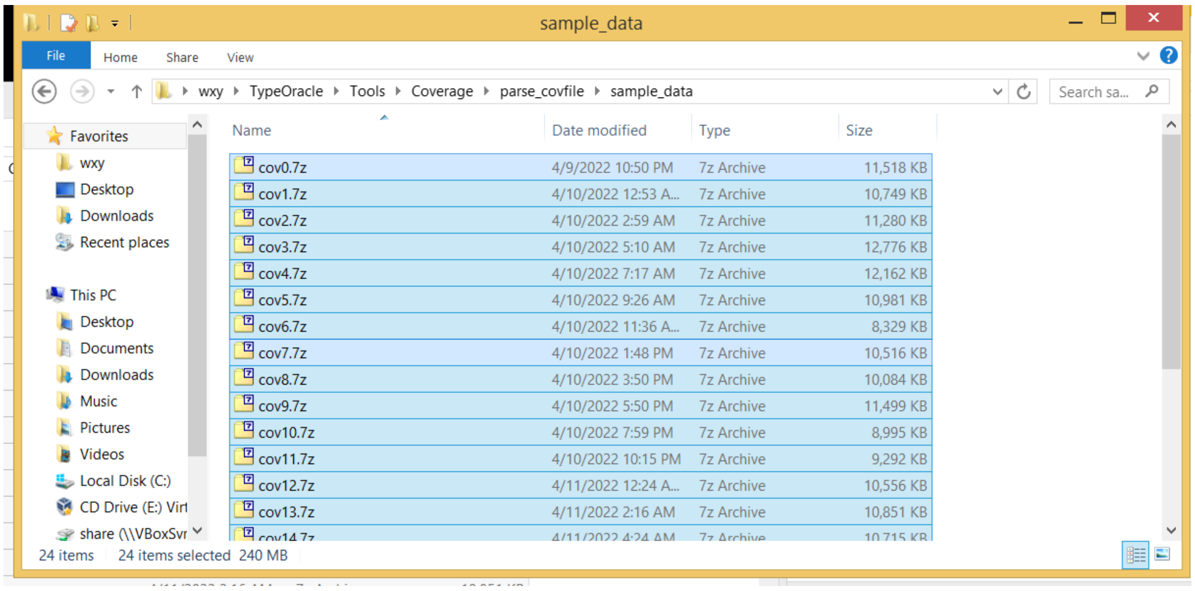
This step is to parse the coverage information collected by DynamoRio in step 1, and split them to belonging modules(executable files)

folder structure

```
- input
  - base_log (the recored coverage information of a PDF file without any
JavaScript codes)
  - merge (code that split the coverage information to each module)

- sample_result (the folder store the sample result of this step)

- base.log(the recored coverage information of a PDF file without any JavaScript
codes)

- frame.py

- merge.py (code that split the coverage information to each module)
```

how to reproduce

1. copy coverage file from C:\Users\wxy\TypeOracle\Tools\Coverage\record_cov\adobe\db to C:\Users\wxy\TypeOracle\Tools\Coverage\parse_covfile\sample_data

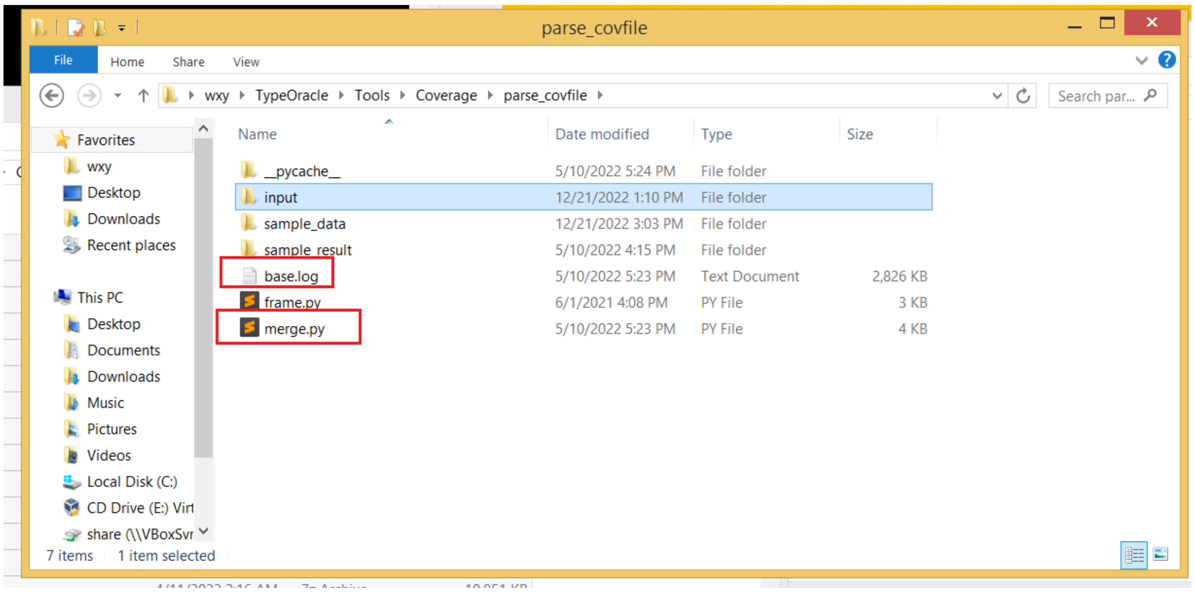


wxy ▸ TypeOracle ▸ Tools ▸ Coverage ▸ parse_covfile ▸ sample_data

Name	Date modified	Type	Size
cov0.7z	4/9/2022 10:50 PM	7z Archive	11,518 KB
cov1.7z	4/10/2022 12:53 A...	7z Archive	10,749 KB
cov2.7z	4/10/2022 2:59 AM	7z Archive	11,280 KB
cov3.7z	4/10/2022 5:10 AM	7z Archive	12,776 KB
cov4.7z	4/10/2022 7:17 AM	7z Archive	12,162 KB
cov5.7z	4/10/2022 9:26 AM	7z Archive	10,981 KB
cov6.7z	4/10/2022 11:36 A...	7z Archive	8,329 KB
cov7.7z	4/10/2022 1:48 PM	7z Archive	10,516 KB
cov8.7z	4/10/2022 3:50 PM	7z Archive	10,084 KB
cov9.7z	4/10/2022 5:50 PM	7z Archive	11,499 KB
cov10.7z	4/10/2022 7:59 PM	7z Archive	8,995 KB
cov11.7z	4/10/2022 10:15 PM	7z Archive	9,292 KB
cov12.7z	4/11/2022 12:24 A...	7z Archive	10,556 KB
cov13.7z	4/11/2022 2:16 AM	7z Archive	10,851 KB
cov14.7z	4/11/2022 4:24 AM	7z Archive	10,715 KB

24 items 24 items selected 240 MB

2. copy base.log from C:\Users\wxy\TypeOracle\Tools\Coverage\parse_covfile\input\base_log\adobe (when parsing Adobe's result) / C:\Users\wxy\TypeOracle\Tools\Coverage\parse_covfile\input\base_log\foxit (when parsing Foxit's result), copy merge.py from C:\Users\wxy\TypeOracle\Tools\Coverage\parse_covfile\input\merge\adobe (when parsing Adobe's result) / C:\Users\wxy\TypeOracle\Tools\Coverage\parse_covfile\input\merge\foxit (when parsing Foxit's result)



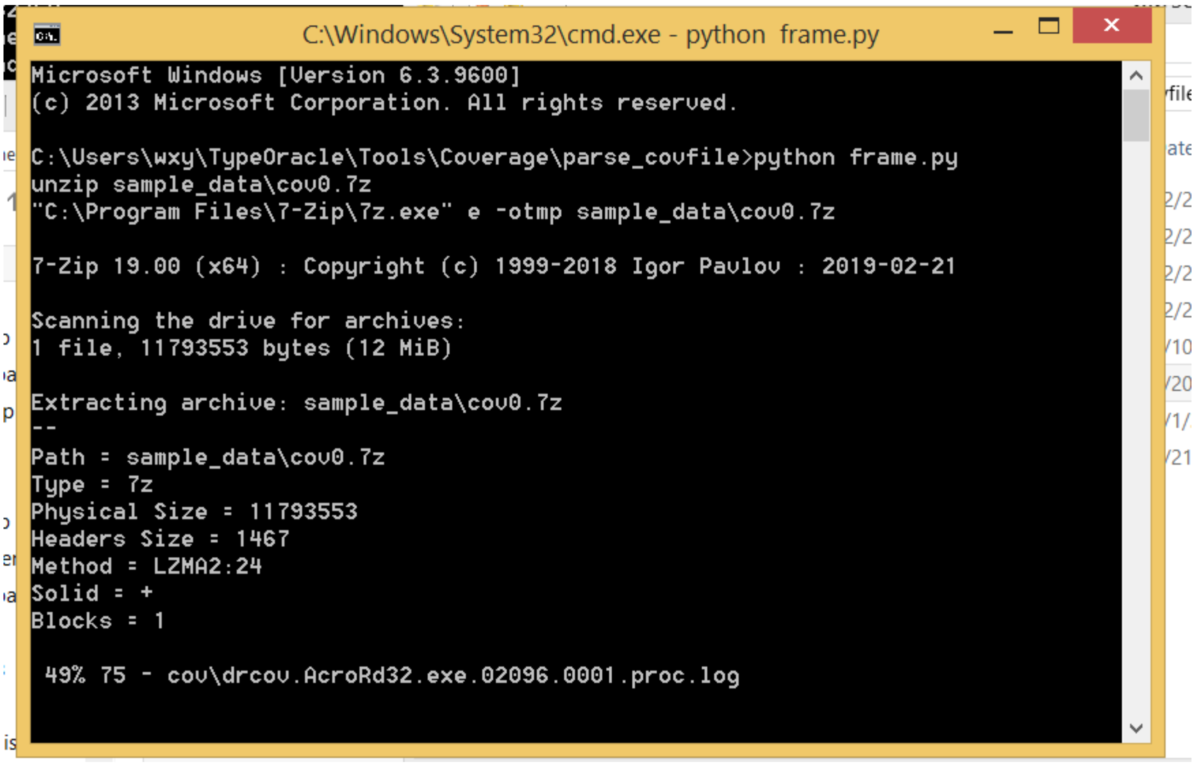
wxy ▸ TypeOracle ▸ Tools ▸ Coverage ▸ parse_covfile ▸

Name	Date modified	Type	Size
__pycache__	5/10/2022 5:24 PM	File folder	
input	12/21/2022 1:10 PM	File folder	
sample_data	12/21/2022 3:03 PM	File folder	
sample_result	5/10/2022 4:15 PM	File folder	
base.log	5/10/2022 5:23 PM	Text Document	2,826 KB
frame.py	6/1/2021 4:08 PM	PY File	3 KB
merge.py	5/10/2022 5:23 PM	PY File	4 KB

7 items 1 item selected

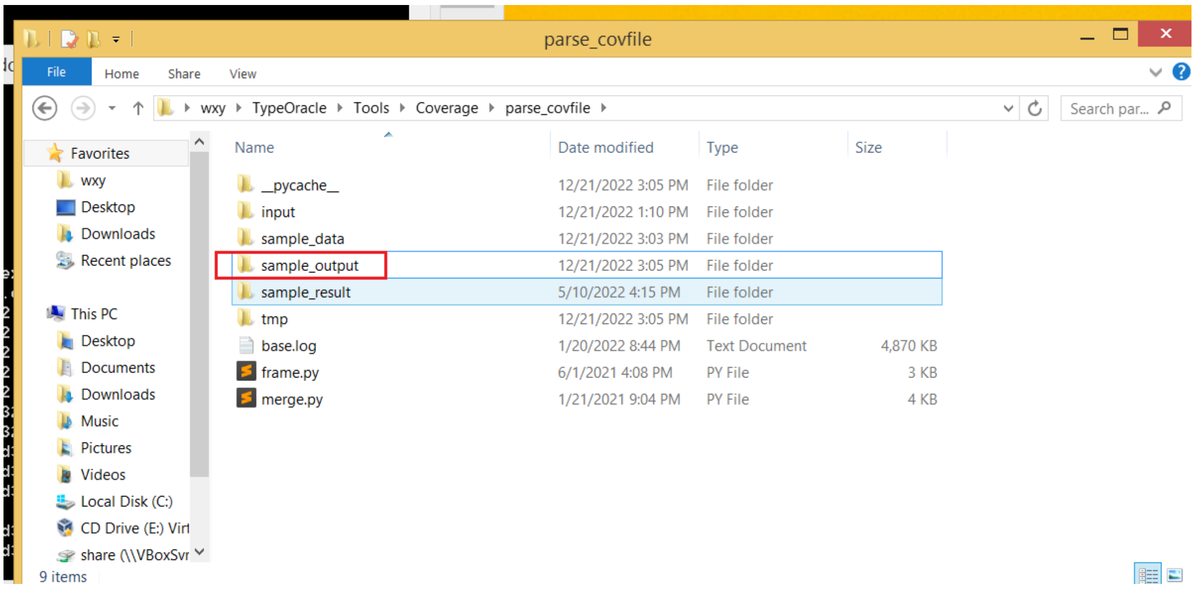
3. run the tool and parse coverage information (it will take several hours)

```
python frame.py
```



```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\wxy\TypeOracle\Tools\Coverage\parse_covfile>python frame.py
unzip sample_data\cov0.7z
"C:\Program Files\7-Zip\7z.exe" e -otmp sample_data\cov0.7z

7-Zip 19.00 (x64) : Copyright (c) 1999-2018 Igor Pavlov : 2019-02-21

Scanning the drive for archives:
1 file, 11793553 bytes (12 MiB)

Extracting archive: sample_data\cov0.7z
--
Path = sample_data\cov0.7z
Type = 7z
Physical Size = 11793553
Headers Size = 1467
Method = LZMA2:24
Solid = +
Blocks = 1

 49% 75 - cov\drcov.AcroRd32.exe.02096.0001.proc.log
```

4. the reuslt is stored in `sample_output` folder



Name	Date modified	Type	Size
__pycache__	12/21/2022 3:05 PM	File folder	
input	12/21/2022 1:10 PM	File folder	
sample_data	12/21/2022 3:03 PM	File folder	
sample_output	12/21/2022 3:05 PM	File folder	
sample_result	5/10/2022 4:15 PM	File folder	
tmp	12/21/2022 3:05 PM	File folder	
base.log	1/20/2022 8:44 PM	Text Document	4,870 KB
frame.py	6/1/2021 4:08 PM	PY File	3 KB
merge.py	1/21/2021 9:04 PM	PY File	4 KB

Stage 3(bbkn2insn)

This stage is to count instruction numbers of every module and add them together.

The coverage information recorded by DynamoRIO is the basic blocks that hitted by the test PDFs, so we need to count the instruction numbers in the basic blocks and add them together.

For Adobe Reader, all the instructions belongs to the modules in `C:\Program Files (x86)\Adobe\Acrobat Reader DC\Reader\plug_ins` , AcroRd32.dll and AcroRd32.exe.

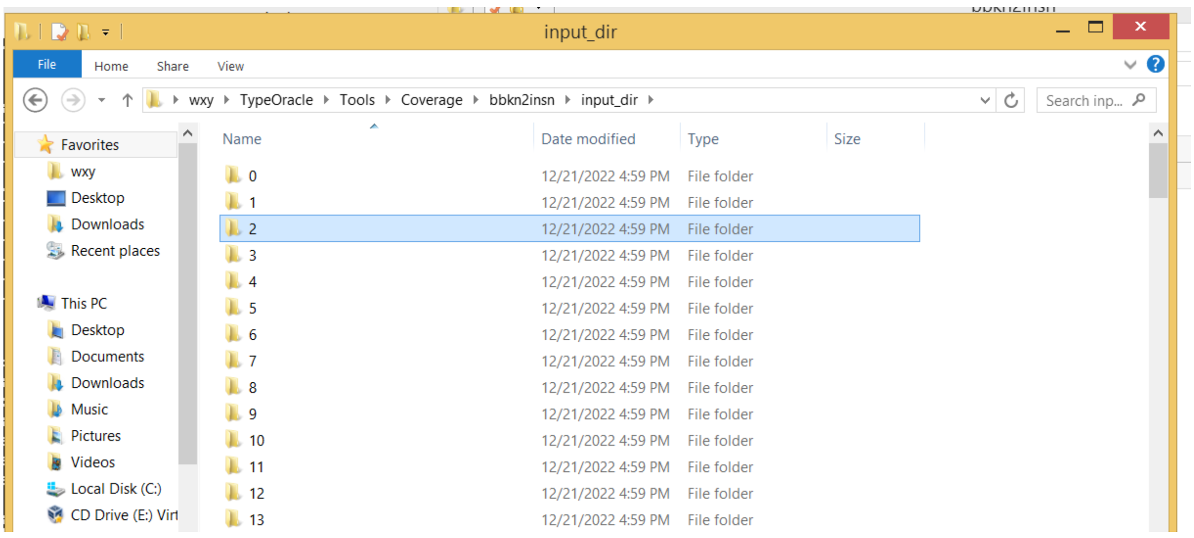
For Foxit Reader, all the instructions belongs to one executable file: FoxitPDFReader.exe/FoxitReader.exe

folder structure

```
- adobe_idb (the folder contains all adobe modules' results generated by IDA pro)

- input_dir (store the input)

- output_dir (store the results, which is the instruction numbers)

- sample_result (the folder store the sample result of this step)

- batch.py

- combine.py (add the instrction numbers in every module)

- foxitcmd.txt

- FoxitPDFReader.exe.idb (the result generated by parsing FoxitPDFReader.exe through IDA pro)

- inscount.py (script executed in IDA pro to count instruction number)
```

how to reproduce

1. copy coverage file fromC:\Users\wxy\TypeOracle\Tools\Coverage\parse_covfile\sample_output to C:\Users\wxy\TypeOracle\Tools\Coverage\bbkn2insn\input_dir

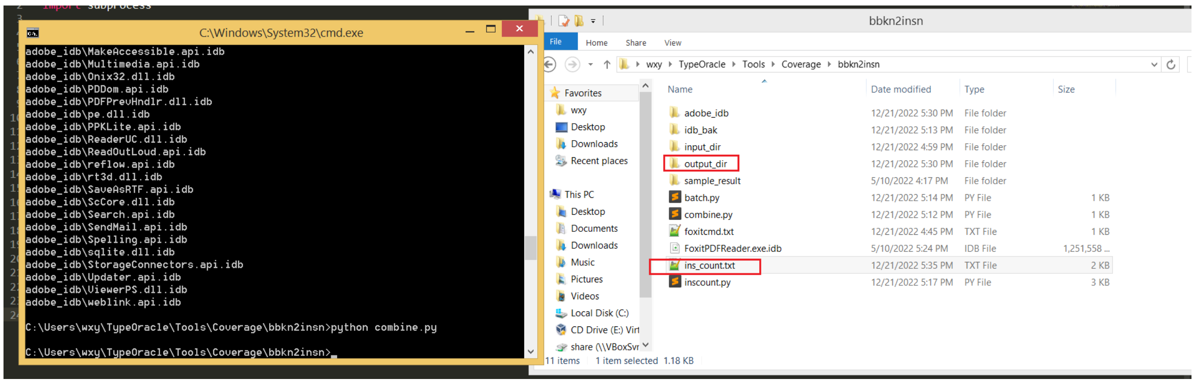


Name	Date modified	Type	Size
0	12/21/2022 4:59 PM	File folder	
1	12/21/2022 4:59 PM	File folder	
2	12/21/2022 4:59 PM	File folder	
3	12/21/2022 4:59 PM	File folder	
4	12/21/2022 4:59 PM	File folder	
5	12/21/2022 4:59 PM	File folder	
6	12/21/2022 4:59 PM	File folder	
7	12/21/2022 4:59 PM	File folder	
8	12/21/2022 4:59 PM	File folder	
9	12/21/2022 4:59 PM	File folder	
10	12/21/2022 4:59 PM	File folder	
11	12/21/2022 4:59 PM	File folder	
12	12/21/2022 4:59 PM	File folder	
13	12/21/2022 4:59 PM	File folder	

2. run the tool to count instruction numbers (10-20 minutes)

for Adobe Reader:

```
python batch.py
python combine.py
```



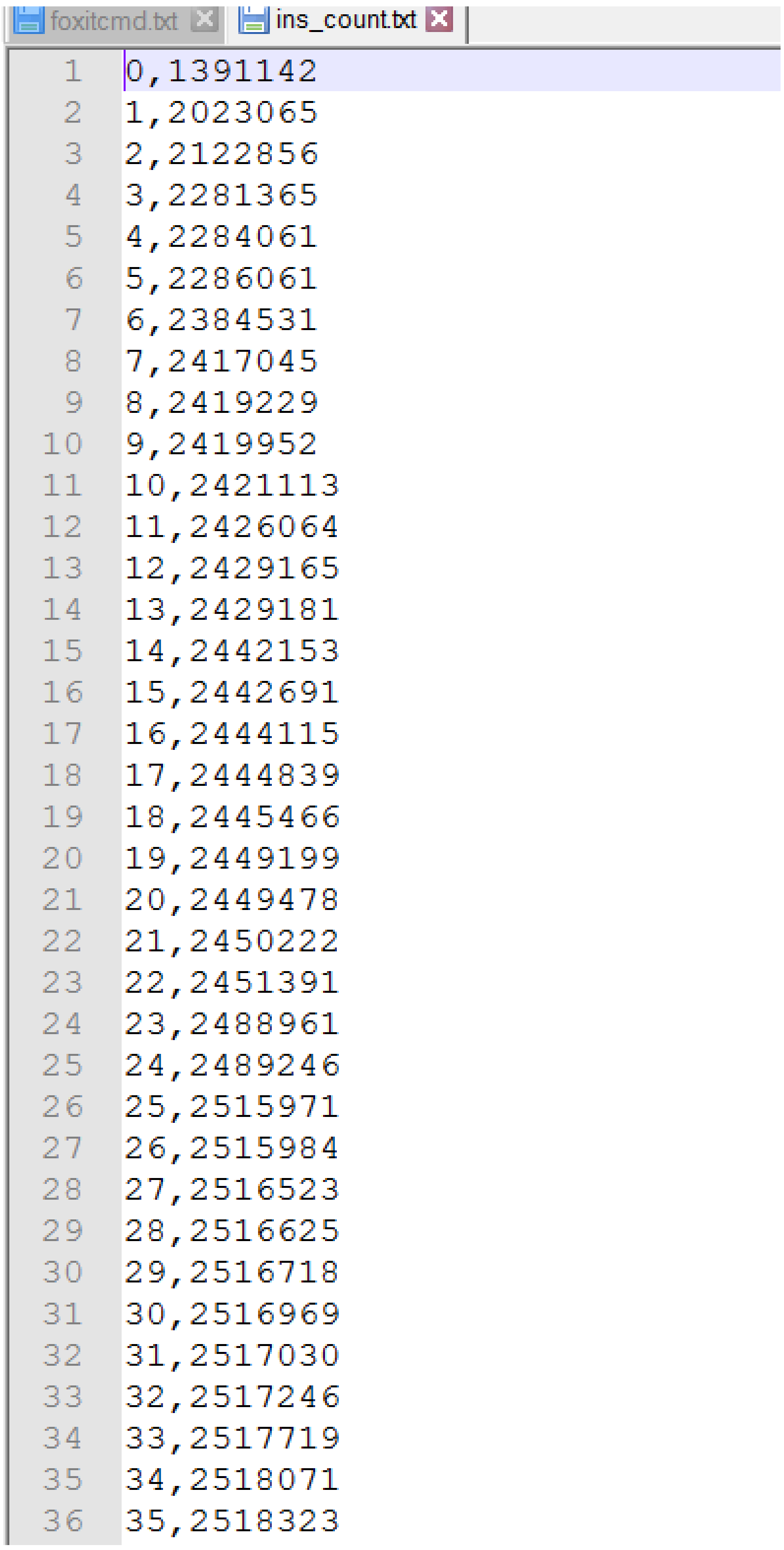
```
adobe_idb\MakeAccessible.api.idb
adobe_idb\Multimedia.api.idb
adobe_idb\Onix32.dll.idb
adobe_idb\PDDom.api.idb
adobe_idb\PDFPrevHndlr.dll.idb
adobe_idb\pe.dll.idb
adobe_idb\PPKLite.api.idb
adobe_idb\ReaderUC.dll.idb
adobe_idb\ReadOutLoud.api.idb
adobe_idb\reflow.api.idb
adobe_idb\rt3d.dll.idb
adobe_idb\SaveAsRTF.api.idb
adobe_idb\ScCore.dll.idb
adobe_idb\Search.api.idb
adobe_idb\SendMail.api.idb
adobe_idb\Spelling.api.idb
adobe_idb\sqlite.dll.idb
adobe_idb\StorageConnectors.api.idb
adobe_idb\Updater.api.idb
adobe_idb\ViewerPS.dll.idb
adobe_idb\weblink.api.idb

C:\Users\wxy\TypeOracle\Tools\Coverage\bbkn2insn>python combine.py

C:\Users\wxy\TypeOracle\Tools\Coverage\bbkn2insn>_
```

wxy ▸ TypeOracle ▸ Tools ▸ Coverage ▸ bbkn2insn

Name	Date modified	Type	Size
adobe_idb	12/21/2022 5:30 PM	File folder	
idb_bak	12/21/2022 5:13 PM	File folder	
input_dir	12/21/2022 4:59 PM	File folder	
output_dir	12/21/2022 5:30 PM	File folder	
sample_result	5/10/2022 4:17 PM	File folder	
batch.py	12/21/2022 5:14 PM	PY File	1 KB
combine.py	12/21/2022 5:12 PM	PY File	1 KB
foxitcmd.txt	12/21/2022 4:45 PM	TXT File	1 KB
FoxitPDFReader.exe.idb	5/10/2022 5:24 PM	IDB File	1,251,558 ...
ins_count.txt	12/21/2022 5:35 PM	TXT File	2 KB
inscount.py	12/21/2022 5:17 PM	PY File	3 KB

11 items 1 item selected 1.18 KB

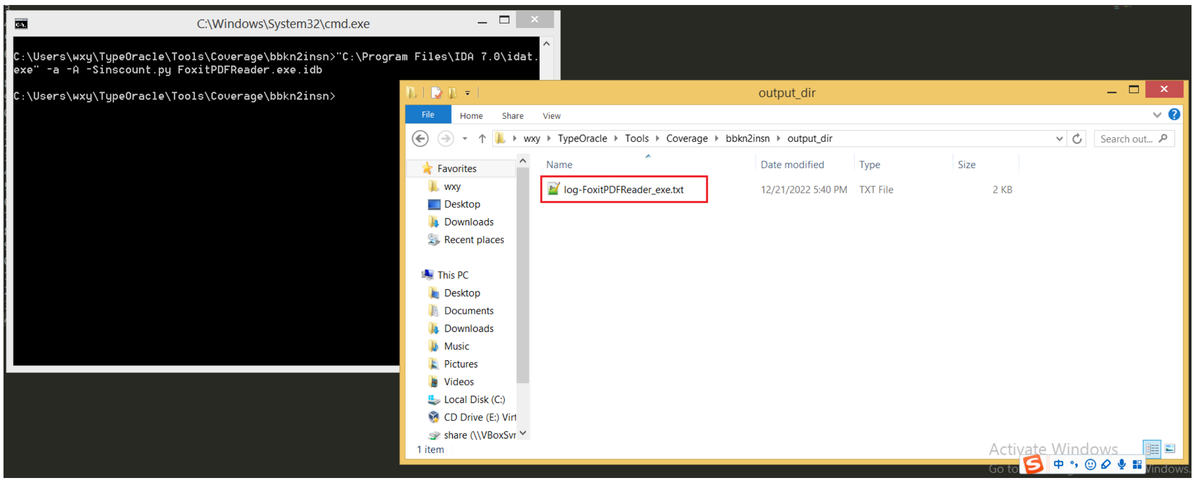
the result is in ins_count.txt



```
0,1391142
1,2023065
2,2122856
3,2281365
4,2284061
5,2286061
6,2384531
7,2417045
8,2419229
9,2419952
10,2421113
11,2426064
12,2429165
13,2429181
14,2442153
15,2442691
16,2444115
17,2444839
18,2445466
19,2449199
20,2449478
21,2450222
22,2451391
23,2488961
24,2489246
25,2515971
26,2515984
27,2516523
28,2516625
29,2516718
30,2516969
31,2517030
32,2517246
33,2517719
34,2518071
35,2518323
```

for Foxit Reader(execute the command in foxitcmd.txt):

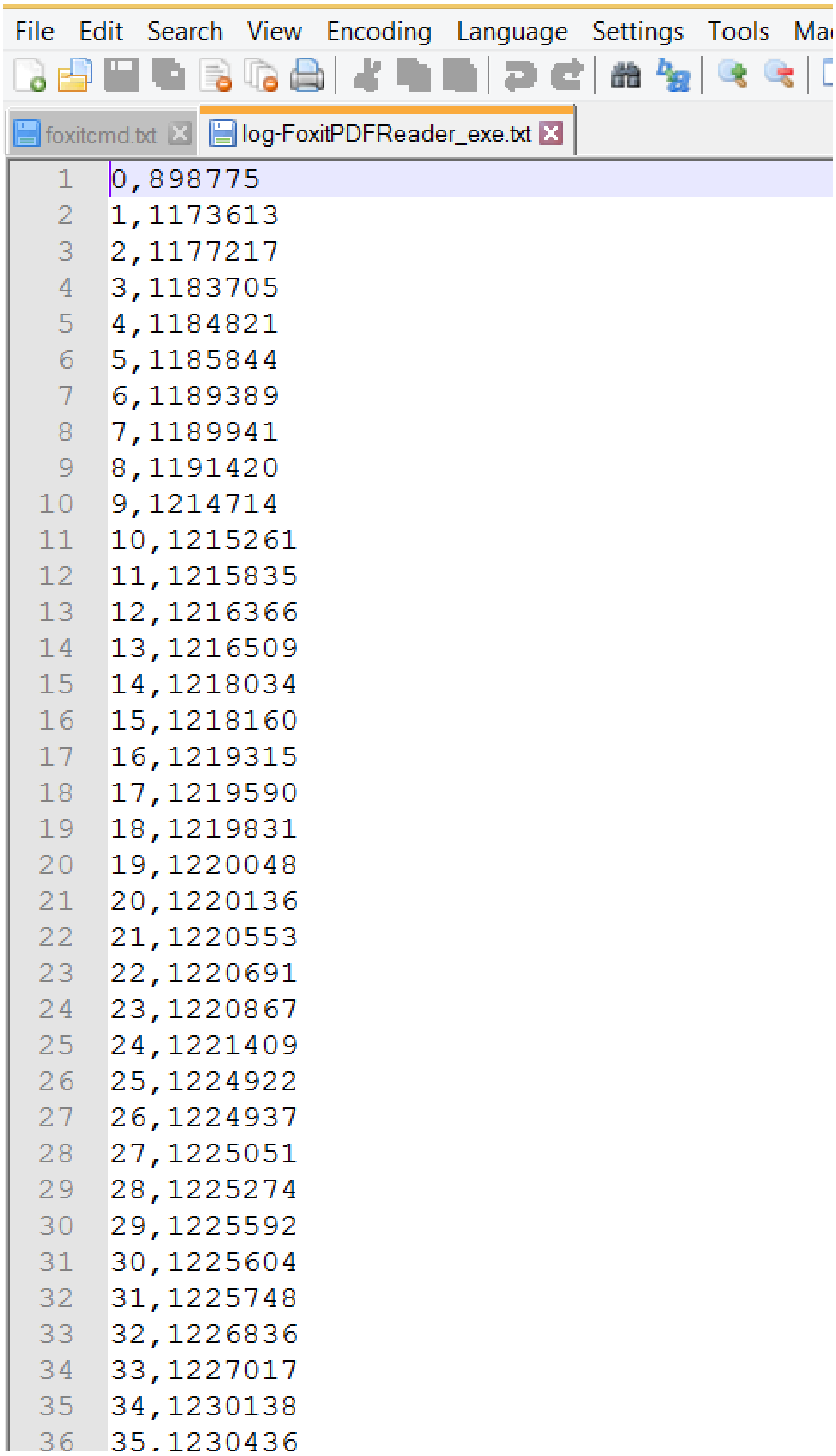
```
"C:\Program Files\IDA 7.0\idat.exe" -a -A -Sinscount.py FoxitPDFReader.exe.idb
```



```
C:\Users\wxy\TypeOracle\Tools\Coverage\bbkn2insn>"C:\Program Files\IDA 7.0\idat.
exe" -a -A -Sinscount.py FoxitPDFReader.exe.idb

C:\Users\wxy\TypeOracle\Tools\Coverage\bbkn2insn>
```

```
wxy ▸ TypeOracle ▸ Tools ▸ Coverage ▸ bbkn2insn ▸ output_dir

Name                          Date modified        Type       Size
log-FoxitPDFReader_exe.txt    12/21/2022 5:40 PM   TXT File   2 KB

1 item
```

the result is in `output_dir/log-FoxitPDFReader_exe.txt`



```
0,898775
1,1173613
2,1177217
3,1183705
4,1184821
5,1185844
6,1189389
7,1189941
8,1191420
9,1214714
10,1215261
11,1215835
12,1216366
13,1216509
14,1218034
15,1218160
16,1219315
17,1219590
18,1219831
19,1220048
20,1220136
21,1220553
22,1220691
23,1220867
24,1221409
25,1224922
26,1224937
27,1225051
28,1225274
29,1225592
30,1225604
31,1225748
32,1226836
33,1227017
34,1230138
35,1230436
```