Assignment 2: Maintaining & Evolving OSS

Due date: 03/10/2020

1   Introduction

By now, you should be familiar with *jEdit*[1], a free and open-source text editor; and *PDFsam*[2], a free and open source application to manage PDF files. In this assignment, you are to implement **four** change requests: **two** for *jEdit* and **two** for *PDFsam*.

> You are to choose the two change requests to implement for each system out of the options given in Sections 2 and 3.

Instructions for getting the source code, building and deploying jEdit and PDFsam were given in A1. Although those instructions were based on the *Eclipse IDE*, you were instructed to use any other IDE you felt comfortable with. Apply the same rule to complete this assignment.

When preparing this assignment, consider that the goals of this activity are:

  i.   To provide students with a space to implement changes in existing software systems; and
  ii.   To put in practice concepts related to incremental change.

Remember that the assignments in this class are to be developed in pairs, so your first task is to find a teammate.

> Read carefully the assignment before proceeding!

2   jEdit

Choose two of the following change requests and implement them in jEdit. Remember that you are to use the source code tagged as **jedit-5.5.0**.

2.1   Change request #je-1

The left side of the *status bar* of jEdit reports: the line number containing the caret, the column position of the caret, the character offset of the caret from the beginning of the file, and the number of characters in the file (see Figure 1). The request is to modify the status bar to show: the word offset of the caret from the beginning of the file and the number of words in the file. As a result of this change, the status bar for the example of Figure 1 should report: **216,23 (3777/775740)(536/124706)**.

For the sake of consistency, the new caret display options should be added to the **Status Bar** preference dialog as check boxes (Figure 2).
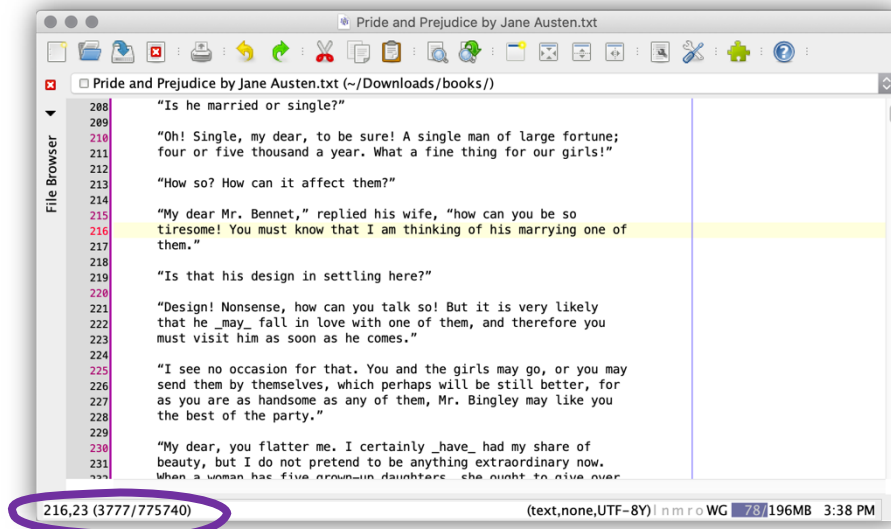
---

[1] http://www.jedit.org/
[2] https://pdfsam.org

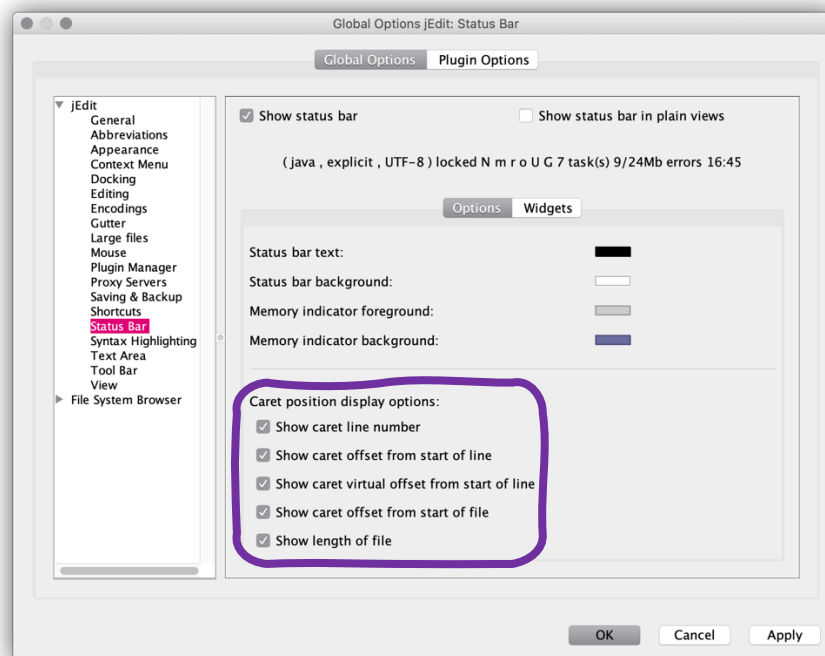*Figure 1. Caret position information in jEdit*

Figure 2. Preference dialog for Status Bar options

2.2   Change request #je-2

Currently, jEdit displays the horizontal and vertical scroll bars (see Figure 3) whenever the content of the opened document exceeds the size of the editor. Implement a **Toggle Scroll Bars** option in the **View** menu that allows to show/hide the scrollbars.
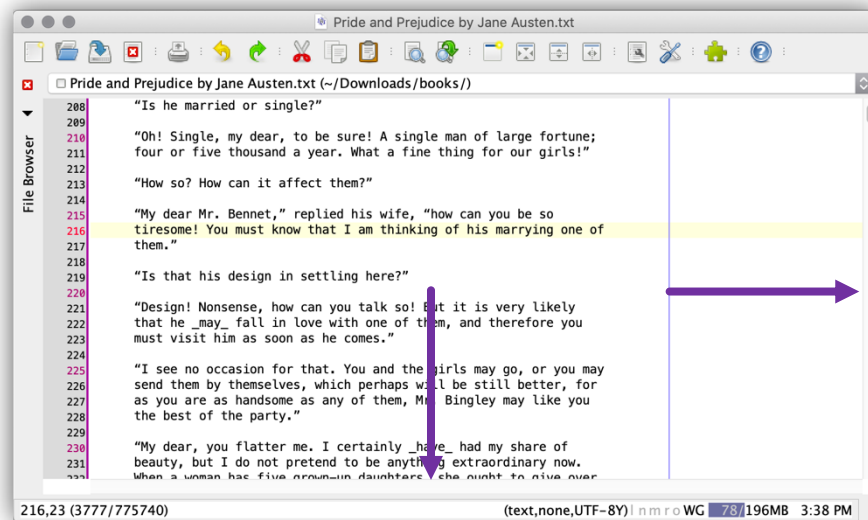
Figure 3. Scroll bars in the jEdit text editor

2.3   Change request #je-3

In jEdit, the HyperSearch feature should list all occurrences of the search string (different **Search>Find**, which only locates the next match). However, the list of results of a HyperSearch only highlights the first occurrence of the search string in any given line, as shown in Figure 4 for the search string "**the**". Note that lines 522, 533, and 534, among others, contain more than one occurrence of this string, but only the first one is highlighted (in purple). The request is to fix the HyperSearch feature, so that its list of results highlights all occurrences of a search string.
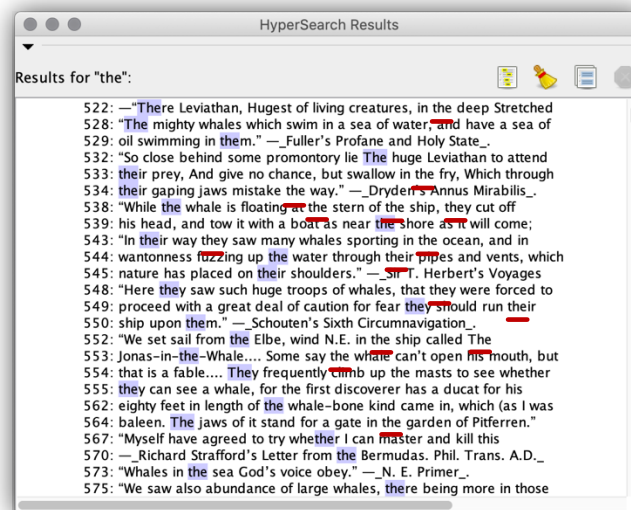

Figure 4. List of results of a HyperSearch in jEdit. Only the first match is highlighted (in purple) in each line.

3   PDFsam

Choose two of the following change requests and implement them in PDFsam. Remember that you are to use the source code tagged as **v4.0.5.**

3.1   Change Request #ps1

The **Split by size** module of PDFsam allows to split a PDF file in files of a given size (see Figure 5). Ideally, the created files should be smaller than the given size, but this does not always happen (see Figure 6). You are requested to fix this functionality, so that the created files never exceed the specified size, unless the created file contains a single page that by itself exceeds the limit size.
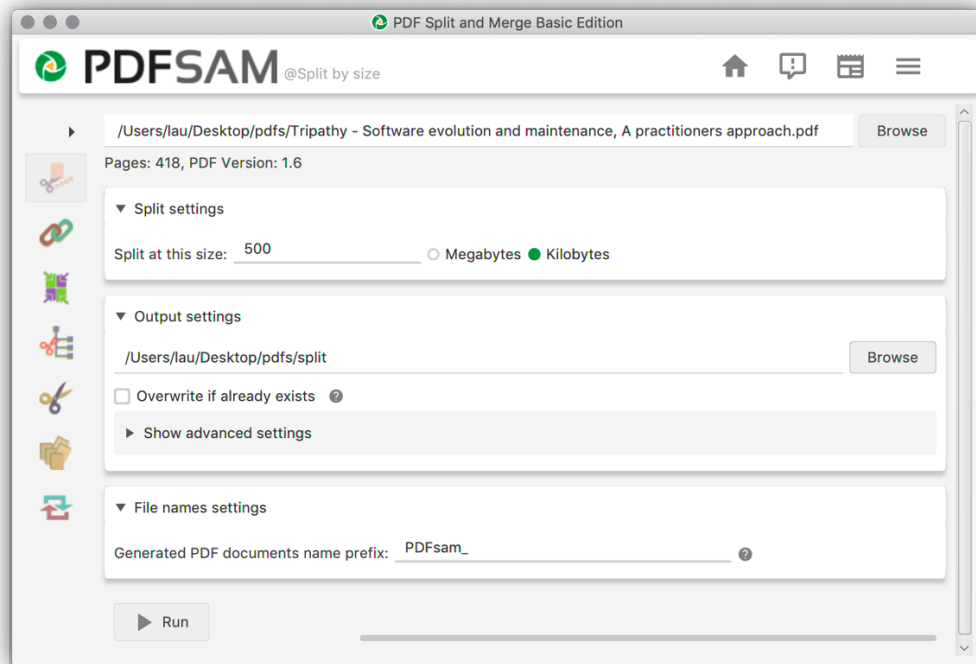

Figure 5.   Split by size module of PDFsam

Figure 6.   Files created by PDFsam after splitting at 500 Kilobytes

3.2   Change Request #ps2

The **Merge** module throws an exception upon attempting to merge page ranges that intersect (see Figure 7 and Figure 8). You are to fix this issue by allowing intersection of ranges during the merging operation.
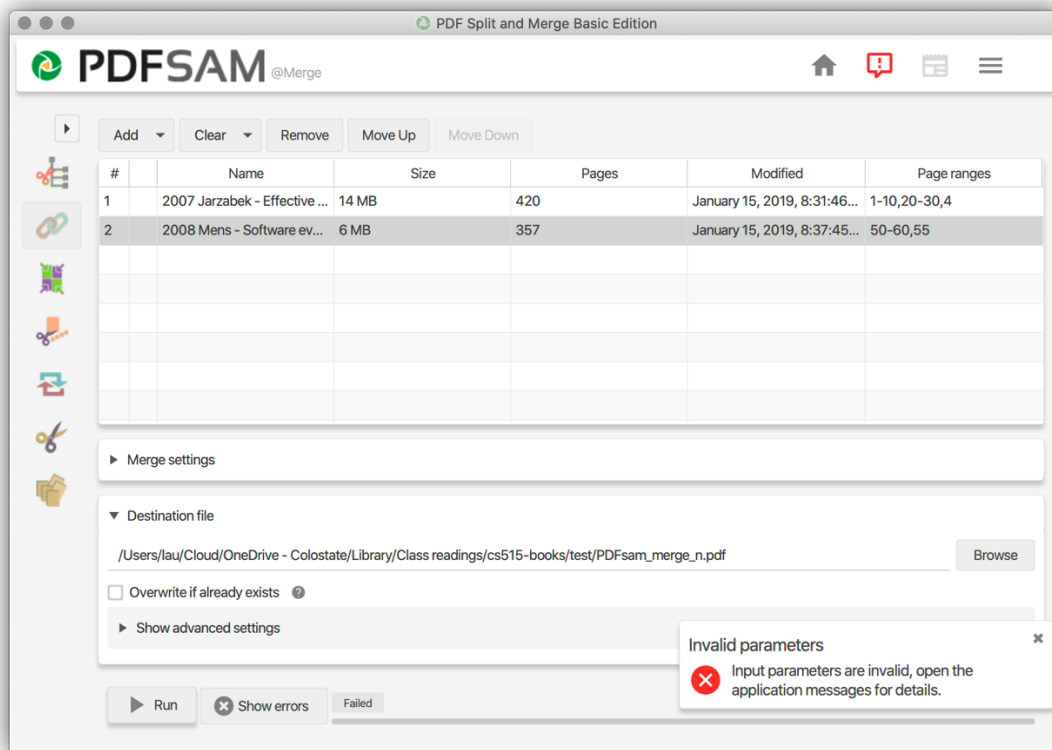
Figure 7. Merge module of PDFsam—Error when trying to merge overlapping page ranges
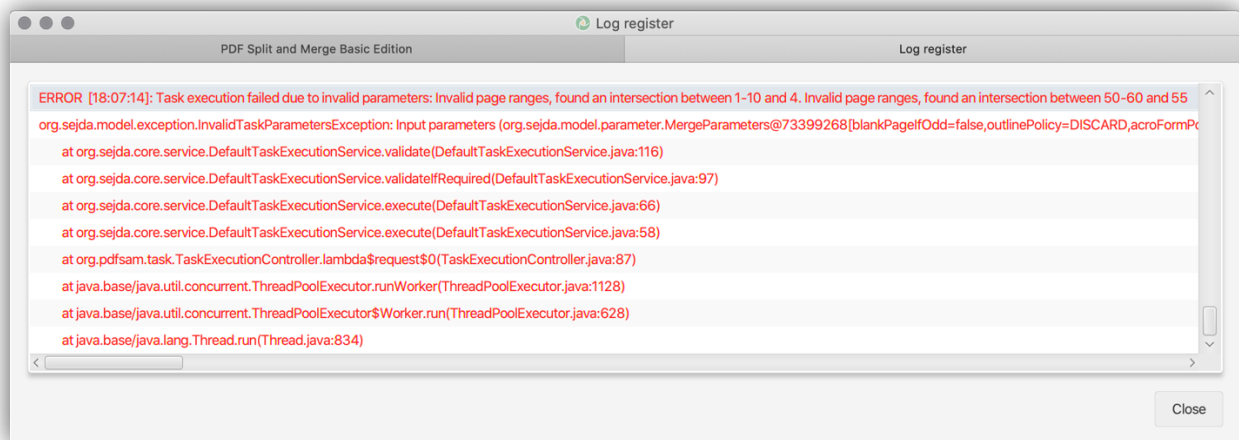

Figure 8. Error registered by PDFsam when trying to merge overlapping page ranges

3.3   Change Request #ps3

The **Rotate** module allows to rotate the pages of multiple PDF documents by:

(i) Providing a range of pages to rotate, which can be done for each document; or

(ii) Selecting even, odd or all pages to rotate for those documents whose range was not defined.

These two options are exclusive, i.e., the even/odd/all-pages selection is disregarded whenever the page range is specified for a document. For example, the results of executing the task depicted in Figure 9 are:

- For document #1, a pdf file consisting of 371 pages, where all even pages are rotated 90° clockwise.

- For document #2, a pdf file consisting of 612 pages, where the first 10 pages are rotated 90° clockwise.

- For document #3, a pdf file consisting of 420 pages, where pages 20 to 100 are rotated 90° clockwise.

You are requested to modify this feature to consider **both** the page-range and the even/odd/all options **in conjunction** when executing the job. This means that for the task depicted in Figure 9, the results should be:

- For document #1, a pdf file consisting of 371 pages, where all even pages are rotated 90° clockwise.

- For document #2, a pdf file consisting of 612 pages, where all even pages in the range 1 to 10 (i.e., 2, 4, 6, 8, 10) are rotated 90° clockwise.

- For document #3, a pdf file consisting of 420 pages, where all even pages in the range 20 to 100 (e.g., 20, 22, 24, …) are rotated 90° clockwise.
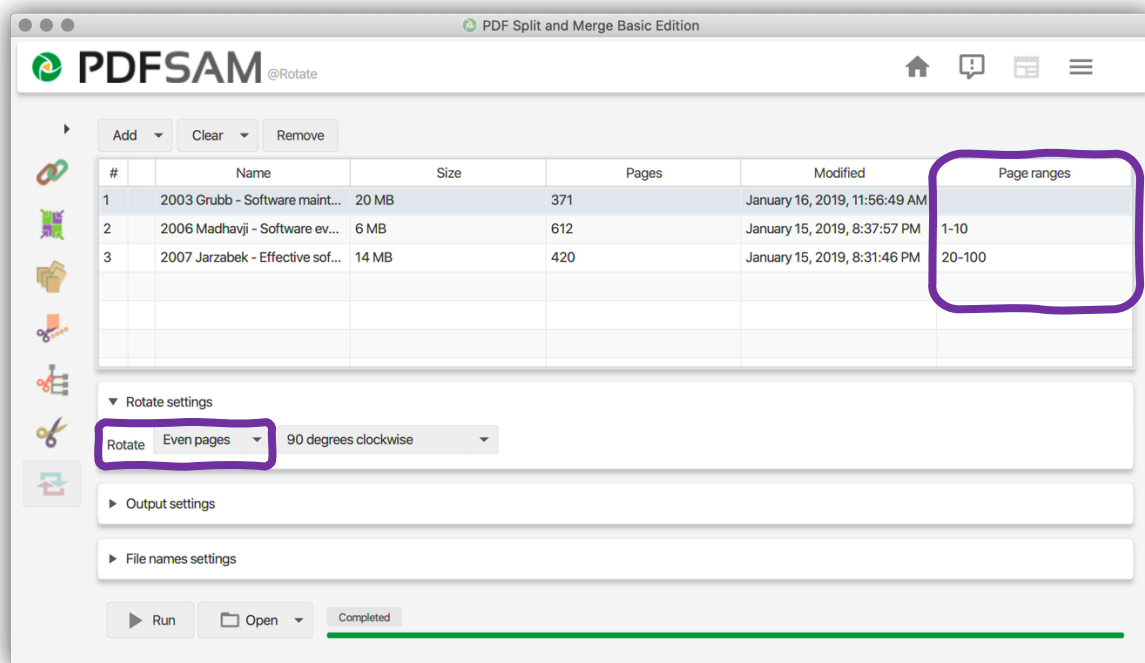

Figure 9. Rotate module of PDFsam—Page selection options

4   Deliverables

There are two types of deliverables for this assignment:

1. **Modified source code of the systems in GitHub**. Consider the next instructions:
   a.   Use the repositories created in A1 as a starting point for this assignment.
   b.   Modify the source code according to the chosen change requests. Commit your changes as needed and include the corresponding change-request ID as part of each commit message.
   c.   Do not forget to push the changes to your GitHub repositories.

     d.   Add the instructor[3] and TA[4] as collaborators of your repositories.

2. **Change log for each change request**. Consider the next instructions:
   a.   Download the change log form from Canvas.
   b.   Fill-in the change log form for each implemented change request (ideally while implementing the change requests).
   c.   Add the filled forms to your GitHub repository.
   d.   Submit the logs through Canvas (one file per change request).

The instructions need to be completed no later than the due date by 12:30pm (i.e., before class).

5   Notes

- Name the repository after your team name as follows: **cs515-001-s20-teamname-projectname**, where **teamname** is the name that you have chosen for your team, and **projectname** is either jedit or pdfsam.
- About the logs:
  - Name your logs as follows: **cs515-001-teamname-changerequestid**, where **teamname** is the name that you have chosen for your team, and **changerequestid** is the ID of the change request you are implementing (e.g., je-3).
  - The logs should not contain any gap in the process you followed. Provide enough details about queries, tools used, classes visited, decisions, etc. Any person should be able to precisely understand and replicate the process you followed. It is very important to clearly describe the rationale of your tasks and decisions.
  - Do not forget to evaluate the process you went through and provide enough details in the conclusions in the change request logs. Include details on why some steps were harder than others.
  - Specify in the logs who of the team members did the coding and who did the documentation.
- You will not receive credit for this assignment if:
  - You do not fill and submit the logs.
  - The files you submit cannot be opened.
  - Your code does not compile or does not pass the tests.
  - You do not add the instructor as collaborator in your repository.
- You will get points off if:
  - Your logs contain typos or grammar errors.
  - Your submission does not fit the brief.
  - Your submission does not follow the specified naming conventions.
  - The projects are not stored in a **GitHub** repository.

---

[3] GitHub account: lvmorenoc
[4] GitHub account: salonich