README.md

PA9 - More String Manipulation

Due Date: Thursday, Jun 3, 11:59PM Pacific Time

Learning Goals

In this assignment, you will extend from PA7 and add more string manipulation features.

Collaboration

Different assignments in this course have different collaboration policies. On this assignment, you cannot share code publicly on Piazza or with anyone other than the course staff. If you need to share code, ask in a private Piazza post or in 1-on-1 hours. Still do ask any public code questions about lecture code, quizzes that are past, or other conceptual questions, just no code from this PA. You can't get help from anyone but the course staff on this PA.

On this assignment, we **encourage** you to share publicly and with other students what you think the expected output should be on particular examples. For example, you could share a `java StringSearch ...` command you tried out, and show the results, and check with other students if they agree on the behavior. This allows you to discuss how the assignment is supposed to work without sharing any code, and you also might want to share examples you found interesting!

Resubmission/Late Policy

- We will not accept this PA late.
- Because we are near the end of the quarter and there will be two more exams for the staff to grade, we do not have the bandwidth to set up a resubmission assignment. Make sure you submit your work **on time**.

Task 1

Since this PA is an extension of PA7, you can finish PA7 first and submit it to `PA7 Resubmission` assignment on Gradescope to check your work.

(Remember that you can earn up to 90% credit back for doing that, so it's really worth it to do so!)

As an alternative, you can use the starter code without finishing PA7. You can get a significant amount of credit for just the PA9 tasks (based on starter code), but getting *full credit* requires that you get PA7 working.

Task 2

You can find starter code in this repository

You can use the starter code provided, or directly build on top of your PA7 `StringSearch.java` if it passed PA7 autograder tests.

1. Add the following query option to your PA7 implementation:

   `word=<string>` – matches lines where the given string appears either at the beginning of the line followed by a space, the end of the line preceded by a space, or somewhere in the middle of the line with spaces on either side.

   Examples: The query `word='bat'` would match these lines:

   - bat man really should be written as one word

    - step up to the bat
    - didn't bat an eye

  It would not match these lines:

    - battery-powered
    - bubble bath

2. Add the following transform option to your PA7 implementation:

`range=<number>,<number>` – selects a range of characters from the line starting at the first number (inclusive) and ending at the second number (exclusive), like `substring`. If the line is too short for the first index, the empty string is produced. If the line is too short for the second index, the transform should take as many characters as possible.

Examples: The transform `range=4,7` would have the following effects:

- `hello` → `o`
- `four` → an empty string
- `parametric` → `met`

Task 3

Answer the following questions in the **Task3.docx** file.

- Around how much code did you **add** (on top of the starter code or your PA7 `StringSearch.java`) in order to add these new features? Identify the new classes/methods/parts of methods that were added.
- Around how much code did you **change** (on top of the starter code or your PA7 `StringSearch.java`) in order to add these new features? Identify the code that was there before and after.
- If you made other changes to your program to add these features, describe those changes as well.

Testing

We encourage you to try different `java StringSearch ...` commands to test your program before submission.

We will use the following kinds of tests to grade your code:

- Tests where we use only a single new word query

- Tests where we use only a single old query from PA7 and the new range transform
- Tests where we use only a single new word query and a single new range transform
- Tests where we use a sequence of queries where one or more of them is a new word query and some are old queries
- Tests where we use a single existing query and a sequence of transforms where one or more them is a new range
- …

The following commands, when run at the command line, should produce the following output:

```
$ java StringSearch "poem.txt" "word='a'"
This is a short file
Also a haiku
$ java StringSearch "poem.txt" "word='is'" "range=1,10"
his is a
t contain
$ java StringSearch "poem.txt" "contains='text'" "range=3,15"
contains tex
$ java StringSearch "poem.txt" "word='a'&less=15" "upper"
ALSO A HAIKU
$ java StringSearch "poem.txt" "not(contains='and')" "range=2,100"
is is a short file
so a haiku
$ java StringSearch "poem.txt" "greater=1&contains='is'&word='file'"
"replace='o';'0'&range=0,16"
This is a sh0rt
```

Submission

Export the `Task3.docx` file as a PDF file. Submit your `StringSearch.java` and your PDF file to PA9 on Gradescope. There will be an autograder available while the assignment is out, and we will also give you feedback on your code after submission. Again, there will **not** be an opportunity for you to resubmit after the deadline, so please check your work and submit it on time.

Releases

No releases published

Packages

No packages published

Contributors 2


shihualu


ucsd-miranda Greg Miranda

Languages

- **Java** 90.0%
- **Shell** 5.8%
- **Batchfile** 4.2%