Java SearchWithParallelSpliterator Phrase MatchSpliterator Ctor & tryAdvance() Method

Douglas C. Schmidt

d.schmidt@vanderbilt.edu

www.dre.vanderbilt.edu/~schmidt

Professor of Computer Science

Institute for Software Integrated Systems

Vanderbilt University

Nashville, Tennessee, USA

Learning Objectives in this Part of the Lesson

- Be aware of how a parallel spliterator can improve parallel stream performance
- Know the intent of—& fields in—the PhraseMatchSpliterator
- Recognize the PhraseMatchSpliterator constructor & tryAdvance() method implementation

```
class PhraseMatchSpliterator
      implements Spliterator<Result> {
  ...
  PhraseMatchSpliterator(CharSequence input,
                         String phrase) { ... }

  boolean tryAdvance(Consumer<? super Result> action) { ... }
  ...
```

These methods are identical w/the SearchWithSequentialStreams class

See "*Java Sequential SearchStreamGang Example: Applying Spliterator*"

Analysis of PhaseMatch Spliterator Constructor

Analysis of PhraseMatchSpliterator Constructor

- PhraseMatchSpliterator uses Java regex to create a stream of SearchResults Result objects that match the # of times a phrase appears in an input string

```
class PhraseMatchSpliterator implements Spliterator<Result> {
  ...
  PhraseMatchSpliterator(CharSequence input, String phrase) {
    String regexPhrase = "\\b" + phrase.trim().replaceAll
                                    ("\\s+", "\\\\b\\\\s+\\\\b")
                               + "\\b"; ...

    mPattern = Pattern.compile(regexPhrase,
                  Pattern.CASE_INSENSITIVE | Pattern.DOTALL);
    mPhraseMatcher = mPattern.matcher(input);
    mInput = input; mPhrase = phrase;
    mMinSplitSize = input.length() / 2;
  } ...
```

See SearchStreamGang/src/main/java/livelessons/utils/PhraseMatchSpliterator.java

Analysis of PhraseMatchSpliterator Constructor

- PhraseMatchSpliterator uses Java regex to create a stream of SearchResults Result objects that match the # of times a phrase appears in an input string

```
class PhraseMatchSpliterator implements Spliterator<Result> {
  ...
  PhraseMatchSpliterator(CharSequence input, String phrase) {
    String regexPhrase = "\\b" + phrase.trim().replaceAll
                                   ("\\s+", "\\\\b\\\\s+\\\\b")
                       + "\\b"; ...
    mPattern = Pattern.compile(regexPhrase,
                               Pattern.CASE_INSENSITIVE | Pattern.DOTALL);
    mPhraseMatcher = mPattern.matcher(input);
    mInput = input; mPhrase = phrase;
    mMinSplitSize = input.length() / 2;
  } ...
```

One work of Shakespeare & a phrase to search for in this work

Analysis of PhraseMatchSpliterator Constructor

- PhraseMatchSpliterator uses Java regex to create a stream of SearchResults Result objects that match the # of times a phrase appears in an input string

```
class PhraseMatchSpliterator implements Spliterator<Result> {
  ...
  PhraseMatchSpliterator(CharSequence input, String phrase) {
    String regexPhrase = "\\b" + phrase.trim().replaceAll
                                   ("\\s+", "\\\\b\\\\s+\\\\b")
                       + "\\b"; ...

    mPattern = Pattern.compile(regexPhrase,
                               Pattern.CASE_INSENSITIVE | Pattern.DOTALL);
    mPhraseMatcher = mPattern.matcher(input);
    mInput = input; mPhrase = phrase;
    mMinSplitSize = input.length() / 2;
  } ...
```

Create a regex that matches phrases

See docs.oracle.com/javase/8/docs/api/java/util/regex/Pattern.html

Analysis of PhraseMatchSpliterator Constructor

- PhraseMatchSpliterator uses Java regex to create a stream of SearchResults Result objects that match the # of times a phrase appears in an input string

```
class PhraseMatchSpliterator implements Spliterator<Result> {
  ...
  PhraseMatchSpliterator(CharSequence input, String phrase) {
    String regexPhrase = "\\b" + phrase.trim().replaceAll
                                    ("\\s+", "\\\\b\\\\s+\\\\b")
                       + "\\b"; ...
    mPattern = Pattern.compile(regexPhrase,
                               Pattern.CASE_INSENSITIVE | Pattern.DOTALL);
    mPhraseMatcher = mPattern.matcher(input);
    mInput = input; mPhrase = phrase;
    mMinSplitSize = input.length() / 2;
  } ...
```

A regex is compiled into a pattern that matches a phrase across lines

See docs.oracle.com/javase/8/docs/api/java/util/regex/Pattern.html

Analysis of PhraseMatchSpliterator Constructor

- PhraseMatchSpliterator uses Java regex to create a stream of SearchResults Result objects that match the # of times a phrase appears in an input string

```
class PhraseMatchSpliterator implements Spliterator<Result> {
  ...
  PhraseMatchSpliterator(CharSequence input, String phrase) {
    String regexPhrase = "\\b" + phrase.trim().replaceAll
                                        ("\\s+", "\\\\b\\\\s+\\\\b")
                                   + "\\b"; ...

    mPattern = Pattern.compile(regexPhrase,
                   Pattern.CASE_INSENSITIVE | Pattern.DOTALL);
    mPhraseMatcher = mPattern.matcher(input);
    mInput = input; mPhrase = phrase;
    mMinSplitSize = input.length() / 2;
  } ...
```

A matcher is created to search the input for the regex pattern

See docs.oracle.com/javase/8/docs/api/java/util/regex/Matcher.html

Analysis of PhraseMatchSpliterator Constructor

- PhraseMatchSpliterator uses Java regex to create a stream of SearchResults Result objects that match the # of times a phrase appears in an input string

```
class PhraseMatchSpliterator implements Spliterator<Result> {
  ...
  PhraseMatchSpliterator(CharSequence input, String phrase) {
    String regexPhrase = "\\b" + phrase.trim().replaceAll
                                    ("\\s+", "\\\\b\\\\s+\\\\b")
                               + "\\b"; ...

    mPattern = Pattern.compile(regexPhrase,
                  Pattern.CASE_INSENSITIVE | Pattern.DOTALL);
    mPhraseMatcher = mPattern.matcher(input);
    mInput = input; mPhrase = phrase;
    mMinSplitSize = input.length() / 2;
  } ...
```

Set key fields with params

Analysis of PhraseMatchSpliterator Constructor

- PhraseMatchSpliterator uses Java regex to create a stream of SearchResults Result objects that match the # of times a phrase appears in an input string

```
class PhraseMatchSpliterator implements Spliterator<Result> {
  ...
  PhraseMatchSpliterator(CharSequence input, String phrase) {
    String regexPhrase = "\\b" + phrase.trim().replaceAll
                                    ("\\s+", "\\\\b\\\\s+\\\\b")
                               + "\\b"; ...

    mPattern = Pattern.compile(regexPhrase,
                  Pattern.CASE_INSENSITIVE | Pattern.DOTALL);
    mPhraseMatcher = mPattern.matcher(input);
    mInput = input; mPhrase = phrase;
    mMinSplitSize = input.length() / 2;
  } ...
```

Define the min split size

This field is used by the trySplit() method for a parallel spliterator

Analysis of PhraseMatchSpliterator tryAdvance() Method

- PhraseMatchSpliterator uses Java regex to create a stream of SearchResults Result objects that match the # of times a phrase appears in an input string

```
class PhraseMatchSpliterator implements Spliterator<Result> {
  ...
  boolean tryAdvance(Consumer<? super Result> action) {
    if (!mPhraseMatcher.find())
        return false;

    else {
        action.accept(new Result
                              (mOffset + mPhraseMatcher.start()));
        return true;
    }
  }
  ...
```

Called by Java streams framework, which attempts to advance the spliterator by one matching phrase

See docs.oracle.com/javase/8/docs/api/java/util/Spliterator.html#tryAdvance

Analysis of PhraseMatchSpliterator tryAdvance() Method

- PhraseMatchSpliterator uses Java regex to create a stream of SearchResults Result objects that match the # of times a phrase appears in an input string

```
class PhraseMatchSpliterator implements Spliterator<Result> {
  ...
  boolean tryAdvance(Consumer<? super Result> action) {
    if (!mPhraseMatcher.find())
        return false;

    else {
        action.accept(new Result
                              (mOffset + mPhraseMatcher.start()));
        return true;
    }
  }
  ...
```

Passes the result (if any) back "by reference" to the streams framework

See docs.oracle.com/javase/8/docs/api/java/util/function/Consumer.html

Analysis of PhraseMatchSpliterator tryAdvance() Method

- PhraseMatchSpliterator uses Java regex to create a stream of SearchResults Result objects that match the # of times a phrase appears in an input string

```
class PhraseMatchSpliterator implements Spliterator<Result> {
  ...
  boolean tryAdvance(Consumer<? super Result> action) {
    if (!mPhraseMatcher.find())
        return false;

    else {
        action.accept(new Result
                              (mOffset + mPhraseMatcher.start()));
        return true;
    }
  }
  ...
```

Check if any remaining phrases in the input match the regex

See docs.oracle.com/javase/8/docs/api/java/util/regex/Matcher.html#find

Analysis of PhraseMatchSpliterator tryAdvance() Method

- PhraseMatchSpliterator uses Java regex to create a stream of SearchResults Result objects that match the # of times a phrase appears in an input string

```
class PhraseMatchSpliterator implements Spliterator<Result> {
  ...
  boolean tryAdvance(Consumer<? super Result> action) {
    if (!mPhraseMatcher.find())
        return false;

    else {
        action.accept(new Result
                              (mOffset + mPhraseMatcher.start()));
        return true;
    }
  }
  ...
```

Inform the streams framework to cease calling tryAdvance() if there's no match

Analysis of PhraseMatchSpliterator tryAdvance() Method

- PhraseMatchSpliterator uses Java regex to create a stream of SearchResults Result objects that match the # of times a phrase appears in an input string

```
class PhraseMatchSpliterator implements Spliterator<Result> {
  ...
  boolean tryAdvance(Consumer<? super Result> action) {
    if (!mPhraseMatcher.find())
        return false;

    else {
        action.accept(new Result
                              (mOffset + mPhraseMatcher.start()));
        return true;
    }
  }
  ...
```

If there is a match, then accept() keeps track of which index in the input string the match occurred

See docs.oracle.com/javase/8/docs/api/java/util/function/Consumer.html#accept

Analysis of PhraseMatchSpliterator tryAdvance() Method

- PhraseMatchSpliterator uses Java regex to create a stream of SearchResults Result objects that match the # of times a phrase appears in an input string

```
class PhraseMatchSpliterator implements Spliterator<Result> {
  ...
  boolean tryAdvance(Consumer<? super Result> action) {
    if (!mPhraseMatcher.find())
        return false;

    else {
        action.accept(new Result
                              (mOffset + mPhraseMatcher.start()));
        return true;
    }
  }
  ...
```

Inform the streams framework to continue calling tryAdvance()

End of Java SearchWithParallel Spliterator PhraseMatch Spliterator Ctor & tryAdvance() Method