Project Report- Types Languages Compilers

Project Name: **JustYourRegularBRzOsJonsVersion**
Team Member: Jonathan Cappell jec2274

Refresher From the Proposal:

Goal: Take in a string, convert it to a 'regex', and construct a DFA (recognizer) using Brzozowski regular expression derivatives.

Method: Implement the mathematical formalism described in Owens, Reppy, and Turon *Regular-expression derivatives reexamined* (2009)
We (Me and Arman Jindal asj2152) have defined discrete steps below to be taken to implement this Goal.
Task:

1. Define a Regular Expression (RegExp) data type and Parser

$$
\begin{array}{lcll}
r, s & ::= & \emptyset & \text{empty set} \\
& | & \varepsilon & \text{empty string} \\
& | & a & a \in \Sigma \\
& | & r \cdot s & \text{concatenation} \\
& | & r^* & \text{Kleene-closure} \\
& | & r + s & \text{logical or (alternation)} \\
& | & r \,\&\, s & \text{logical and} \\
& | & \neg r & \text{complement}
\end{array}
$$

Recursively define a RegExp data type in the Haskell constructor and a checker function for it.

2. Create the helper function v(r) that checks if r is nullable.

$$\nu(r) = \begin{cases} \varepsilon & \text{if } r \text{ is nullable} \\ \emptyset & \text{otherwise.} \end{cases}$$

and is defined as follows:

$$
\begin{array}{rcl}
\nu(\varepsilon) & = & \varepsilon \\
\nu(a) & = & \emptyset \\
\nu(\emptyset) & = & \emptyset \\
\nu(r \cdot s) & = & \nu(r) \,\&\, \nu(s) \\
\nu(r + s) & = & \nu(r) + \nu(s) \\
\nu(r^*) & = & \varepsilon \\
\nu(r \,\&\, s) & = & \nu(r) \,\&\, \nu(s) \\
\nu(\neg r) & = & \begin{cases} \varepsilon & \text{if } \nu(r) = \emptyset \\ \emptyset & \text{if } \nu(r) = \varepsilon \end{cases}
\end{array}
$$

3. Compute the derivative of a, with respect to a symbol **a**:

$$
\begin{array}{rcl}
\partial_a \varepsilon & = & \emptyset \\
\partial_a a & = & \varepsilon \\
\partial_a b & = & \emptyset \quad \text{for } b \neq a \\
\partial_a \emptyset & = & \emptyset \\
\partial_a (r \cdot s) & = & \partial_a r \cdot s + \nu(r) \cdot \partial_a s \\
\partial_a (r^*) & = & \partial_a r \cdot r^* \\
\partial_a (r + s) & = & \partial_a r + \partial_a s \\
\partial_a (r \,\&\, s) & = & \partial_a r \,\&\, \partial_a s \\
\partial_a (\neg r) & = & \neg(\partial_a r)
\end{array}
$$

The rules are extended to strings as follows:

$$
\begin{array}{rcl}
\partial_\varepsilon r & = & r \\
\partial_{ua} r & = & \partial_a (\partial_u r)
\end{array}
$$

4. Create a Haskell data type of a DFA
5. Use the v(r) function and the RegExp derivatives to create a DFA implementing the algorithm below:

```
fun goto q (c, (𝒬, δ)) =
    let q_c = ∂_c q
    in
      if ∃q′ ∈ 𝒬 such that q′ ≡ q_c
        then (𝒬, δ ∪ {(q, c) ↦ q′})
        else
          let 𝒬′ = 𝒬 ∪ {q_c}
          let δ′ = δ ∪ {(q, c) ↦ q_c}
          in explore (𝒬′, δ′, q_c)

and explore (𝒬, δ, q) = fold (goto q) (𝒬, δ) Σ

fun mkDFA r =
    let q_0 = ∂_ε r
    let (𝒬, δ) = explore ({q_0}, {}, q_0)
    let ℱ = {q | q ∈ 𝒬 and ν(q) = ε}
    in ⟨𝒬, q_0, ℱ, δ⟩
```

Fig. 1. DFA construction using RE derivatives

6. Include test cases with RegExps and strings that are in and out of their language. Import the external RegularExpression checker and validate it against the generated DFA to test that we have correctly implemented the algorithm.

All images above are taken from the paper *Regular-expression Derivatives reexamined* By Scott Owens, John Reppy, and Aaron Turon.

<u>What Happened:</u>

Although we proposed the project jointly we implemented the same proposed project separately as two separate projects. All pieces are working from almost whatever angle I view it from. I worked using my implementation of HW2 as the starting point, build system, and example code. It can take derivatives of regular expressions, construct regular expressions, match the regex against a string using the derivatives, construct, DFAs, run DFAs, convert a regex to a DFA using its derivatives, and it can print out steps as to what it is doing during the evaluation of an input or running the DFA. Over the course of implementing the project from the proposal, I learned that the Brzozowski derivatives are themselves regular expressions and that fact is quite neat. Here is the picture from the Owens paper as to how to do matching with just the derivatives and not using a DFA.

$$r \sim \varepsilon \quad \Leftrightarrow \quad \nu(r) = \varepsilon$$
$$r \sim a \cdot w \quad \Leftrightarrow \quad \partial_a r \sim w$$

Also from the Owens paper is an explanation of how to simplify the Regex to help make the equivalence testing of states less catastrophic to witness.

$$\begin{array}{rcllrcl}
r \,\&\, r & \approx & r & (*) & r + r & \approx & r \\
r \,\&\, s & \approx & s \,\&\, r & (*) & r + s & \approx & s + r \\
(r \,\&\, s) \,\&\, t & \approx & r \,\&\, (s \,\&\, t) & (*) & (r + s) + t & \approx & r + (s + t) \\
\emptyset \,\&\, r & \approx & \emptyset & & \neg\emptyset + r & \approx & \neg\emptyset \\
\neg\emptyset \,\&\, r & \approx & r & & \emptyset + r & \approx & r \\
\\
(r \cdot s) \cdot t & \approx & r \cdot (s \cdot t) & & (r^*)^* & \approx & r^* \\
\emptyset \cdot r & \approx & \emptyset & & \varepsilon^* & \approx & \varepsilon \\
r \cdot \emptyset & \approx & \emptyset & & \emptyset^* & \approx & \varepsilon \\
\varepsilon \cdot r & \approx & r & & \neg(\neg r) & \approx & r \\
r \cdot \varepsilon & \approx & r & & & &
\end{array}$$

These were useful to implement so I implemented them in the project.

The Files:

There are several new files including DFAAST.hs, HWDFA.hs, HWREG.hs, ParseString.hs, REGAST.hs, and README_FINAL_PROJECT.md, as well as edited main.hs, lambda.cabal, and package.yaml. The package changes are to install another library to help check the truthfulness of the evaluation. The main is edited to be more useful as a playground with the Regex parser. There is no DFA parser but the project proposal never called for one. The ParseString.hs has the parses function which takes in input as a string and parses it into a Regex of the form defined in REGAST.hs. The REGAST.hs has the definition of a Regex syntax tree and a function to pretty print out the Regex. HWREG.hs has many things including the functions to calculate Brzozowski derivatives (which are Regex) and the nullable function along with various helper functions and displaying functions. The paper actually describes a way to use the derivatives directly to perform matching and that file uses that implementation and uses the external library to check both that implementation and the external library for regex simultaneously. The other library was found to output some odd results due to it matching only some of the string instead of the entire string and possibly other different choices about the way their library works, see the new readme file for an example. The DFAAST.hs has the data type for the DFA implementation along with a pretty print function and a function to allow the construction of the DFA from the regular expression using the algorithm described in step 5, although the explore function was extracted to be just the fold and a function from the HWREG.hs needed to be used to help check for equivalence which is on the order of an exponential number of concatenations. Additionally, various optimization techniques are possible from the paper although they were not in the proposal and mostly not implemented. The file HWDFA.hs has various functions to be convenient and run the DFA on an input. The README_FINAL_PROJECT.md has an explanation of the various parts that are a little bit tricky along with an example ghci log.

Some Thoughts and an Example of How Things Turned Out:

Here is an example from the new readme file that shows a prettyprint of a DFA that was created from a string that was parsed into a regex and then calculated and used the derivatives to create the DFA and pretty print it out:

```
ghci> string_to_dfa "a b"
States: Empty String,Empty Set,b,a concated b,
Start State: a concated b
Accept States: Empty String,
Transition Function:
(Empty String,a),Empty Set)
(Empty String,b),Empty Set)
(Empty String,c),Empty Set)
(Empty Set,a),Empty Set)
(Empty Set,b),Empty Set)
(Empty Set,c),Empty Set)
(b,a),Empty Set)
(b,b),Empty String)
(b,c),Empty Set)
(a concated b,a),b)
(a concated b,b),Empty Set)
(a concated b,c),Empty Set)

Alphabet: abc
```

The transition function outputs in the pretty print in the form ((state, character_input), next_state). The implementation is set up to handle the extended definition of regular expressions just like the paper mentions. The implementation of the Regex is a bit unintuitive as the Literal String builder is meant to have only one Char value inside of it and perform concatenation to everything else but instead does checking in other places and sometimes uses that to hold some information. Overall the parser took by far the longest and most difficult effort to implement as it was rather unintuitive and had a lot of pieces that needed to be correct along with Haskell doing its best to avoid allowing the use of infix operations and confusing documentation. The Literal String implementation does not perform as part 3 from the proposal suggests but instead relies on the Concatenation of many Literal String to work. The entire thing is set up to handle all operators in prefix notation for the regex. After going through this project it seems easier to see how people could build up a regex engine and perform matching, with a lot of help from research papers. See the README_FINAL_PROJECT.md for more information about how to use the project.

Exceptionally Helpful People And Resources:

This project is based on John Hui's helpful recommendation for projects on EdStem, here: https://edstem.org/us/courses/36106/discussion/2893941.

I worked with Arman Jindal asj2152 for the proposal.

Emily Sillars gave guidance, feedback, and pointers on Haskell.

The two papers this is based on helped a lot.
https://www.ccs.neu.edu/home/turon/re-deriv.pdf
https://dl.acm.org/doi/pdf/10.1145/321239.321249

The https://github.com/AlexanderJDupree/DFA-Check and https://github.com/cacay/regexp were helpful to view.

I also looked at quite a lot of slides from Professor Edwards other courses related to Haskell online, along with starter code from the HW2, and viewed the information about Regular Expression Derivatives on the course slides.

I used Michael Sipser's Book "Introduction to the Theory of Computation" as a reference text for some of the more technical theoretical aspects.

Also extra thanks to user Chris for posting this answer https://stackoverflow.com/questions/70468960/haskell-counting-with-foldr which with Emily Sillars and many other uses and viewing of explanations finally helped me understand the value of using fold functions and how to use them in Haskell. This was quite possibly the best explanation of it that I could find that is complex enough to be useful but simple enough to be understood.

Much More Exhaustive References of a Non Exhaustive With Possible Duplicates List of Website Resources I Briefly to Longly Viewed for This Project in no Particular Order:

https://mmhaskell.com/data-structures/hash-map | Hash Maps — Monday Morning Haskell
https://hackage.haskell.org/package/hashmap-1.3.3/docs/Data-HashMap.html#t:Map | Data.HashMap
http://learnyouahaskell.com/modules | Modules - Learn You a Haskell for Great Good!
https://www.cs.auckland.ac.nz/references/haskell/haskell-intro-html/modules.html | A Gentle Introduction to Haskell: Modules
http://www1.cs.columbia.edu/~sedwards/classes/2023/6998-spring-tlc/grammars.pdf | Context-Free Grammars
https://stackoverflow.com/questions/46826700/how-to-create-a-generic-complex-type-in-haskell | How to create a generic Complex type in haskell? - Stack Overflow
https://downloads.haskell.org/~ghc/7.0.3/docs/html/users_guide/pragmas.html | 7.13. Pragmas
https://stackoverflow.com/questions/7768066/how-do-you-define-state-in-haskell | How do you define state in Haskell? - Stack Overflow
https://stackoverflow.com/questions/25160308/complex-data-structures-in-haskell | Complex data structures in Haskell - Stack Overflow
https://stackoverflow.com/questions/43344823/not-in-scope-type-constructor-or-class | haskell - Not in scope: type constructor or class - Stack Overflow
https://stackoverflow.com/questions/49842512/creating-a-new-function-inside-a-function-based-on-the-parameters-of-the-first | haskell - Creating a new function inside a function based on the parameters of the first - Stack Overflow
https://stackoverflow.com/questions/13280159/haskell-foldl-and-foldr | fold - Haskell - foldl and foldr? - Stack Overflow
https://stackoverflow.com/questions/73550138/haskell-illegal-term-level-use-of-the-type-constructor-int | Haskell Illegal term-level use of the type constructor 'Int' - Stack Overflow
https://hackage.haskell.org/package/containers-0.4.0.0/docs/Data-Map.html | Data.Map
https://stackoverflow.com/questions/9524884/how-to-print-data-of-map-as-a-table | haskell - How to print data of Map as a table? - Stack Overflow
https://cheatsheet.codeslower.com/CheatSheet.pdf | CheatSheet.pdf
https://www.google.com/search?q=comparing%20two%20lists%20in%20haskell | comparing two lists in haskell - Google Search
https://hackage.haskell.org/package/containers-0.6.7/docs/Data-Set.html | Data.Set
https://hackage.haskell.org/package/base-4.18.0.0/docs/Data-Ord.html#t:Ord | Data.Ord
https://stackoverflow.com/questions/3065476/haskell-ord-instance-with-a-set | Haskell Ord instance with a Set - Stack Overflow
https://stackoverflow.com/questions/15180188/iterating-over-two-sets-while-updating-two-other-sets-in-haskell | nested loops - Iterating over two sets while updating two other sets in Haskell - Stack Overflow
https://self-learning-java-tutorial.blogspot.com/2016/06/haskell-convert-set-to-list.html | Programming for beginners: Haskell: Convert set to list
https://stackoverflow.com/questions/9014626/iterating-through-a-list-in-haskell | iterating through a list in haskell - Stack Overflow

https://wiki.haskell.org/Fold | Fold - HaskellWiki
https://wiki.haskell.org/Foldr_Foldl_Foldl%27 | Foldr Foldl Foldl' - HaskellWiki
https://stackoverflow.com/questions/1757740/how-does-foldr-work | haskell - How does foldr work? - Stack Overflow
https://stackoverflow.com/questions/70468960/haskell-counting-with-foldr | functional programming - Haskell - counting with foldr - Stack Overflow
https://stackoverflow.com/questions/24202756/how-to-define-and-use-global-array-in-haskell | functional programming - How to define and use global array in Haskell? - Stack Overflow
https://stackoverflow.com/questions/29343668/string-to-list-of-characters | haskell - String to list of characters - Stack Overflow
https://stackoverflow.com/questions/1053926/clojure-while-loop | java - Clojure While Loop - Stack Overflow
https://stackoverflow.com/questions/60398920/the-type-of-foldr-id | haskell - The type of foldr (.) id - Stack Overflow
https://stackoverflow.com/questions/22869058/complement-of-regular-expression-how-to-find | regex - complement of regular expression how to find - Stack Overflow
https://stackoverflow.com/questions/30150877/why-does-this-cause-ghci-to-hang | haskell - Why does this cause GHCI to hang? - Stack Overflow
https://github.com/kseo/secd/blob/master/src/SECD/Eval.hs#L12-L19 | secd/Eval.hs at master · kseo/secd
https://arxiv.org/pdf/2207.02129.pdf | 2207.02129.pdf
https://github.com/sweirich/pi-forall/tree/2022/version1 | pi-forall/version1 at 2022 · sweirich/pi-forall · GitHub
https://verigu.github.io/4115Fall2022/lectures/ocaml.pdf | An Introduction to OCaml
https://verigu.github.io/4115Fall2022/lectures/scanner.pdf | Scanner
https://www.google.com/search?q=haskell%20regular%20language%20checker | haskell regular language checker - Google Search
https://stackoverflow.com/questions/65673275/haskell-check-if-the-regular-expression-r-made-up-of-the-single-symbol-alphabet | regex - Haskell check if the regular expression r made up of the single symbol alphabet Σ = {a} defines language L(r) = a* - Stack Overflow
https://github.com/cacay/regexp | cacay/regexp: Haskell regular expression library that supports derivatives, equivalence, intersection, and complement.
https://wiki.haskell.org/Regular_expressions#Overview | Regular expressions - HaskellWiki
https://swtch.com/~rsc/regexp/regexp1.html | Regular Expression Matching Can Be Simple And Fast
https://www.cs.kent.ac.uk/people/staff/sjt/craft2e/regExp.pdf | regExp.pdf
https://www.google.com/search?q=how%20to%20pronounce%20Brzozowski | how to pronounce Brzozowski - Google Search
https://www.google.com/search?q=regex%20in%20haskell | regex in haskell - Google Search
https://hackage.haskell.org/package/regex-tdfa | regex-tdfa: Pure Haskell Tagged DFA Backend for "Text.Regex" (regex-base)
https://github.com/haskell-hvr/regex-tdfa | haskell-hvr/regex-tdfa: Pure Haskell Tagged DFA Backend for "Text.Regex" (regex-base)

https://www.google.com/search?q=can%20data%20types%20in%20haskell%20have%20functions | can data types in haskell have functions - Google Search
https://mmhaskell.com/liftoff/data-types | Haskell Data Types — Monday Morning Haskell
https://wumbo.net/symbols/negation/ | Negation Symbol (¬)
https://matthewmanela.com/blog/writing-a-regular-expression-parser-in-haskell-part-1/ | Writing a Regular Expression parser in Haskell: Part 1 - Matthew Manela
https://matthewmanela.com/blog/writing-a-regular-expression-parser-in-haskell-part-2/ | Writing a Regular Expression parser in Haskell: Part 2 - Matthew Manela
https://www.codeproject.com/Articles/5412/Writing-own-regular-expression-parser | Writing own regular expression parser - CodeProject
https://github.com/aistrate/RegexParser#parsing-the-regex-language | aistrate/RegexParser: Regular Expression parser (loosely mimicking .NET Framework's System.Text.RegularExpressions namespace).
https://hackage.haskell.org/package/megaparsec-9.3.0/docs/Text-Megaparsec-Char-Lexer.html#g:3 | Text.Megaparsec.Char.Lexer
https://hackage.haskell.org/package/megaparsec-9.3.0/docs/Text-Megaparsec-Char-Lexer.html#v:lexeme | Text.Megaparsec.Char.Lexer
https://stackoverflow.com/questions/67551824/how-to-make-a-multiline-comment-in-haskell | How to make a multiline comment in Haskell? - Stack Overflow
https://verigu.github.io/4115Spring2022/ | COMS W4115 (Spring 2022) | 4115Spring2022
https://markkarpov.com/tutorial/megaparsec.html#char-and-string-literals | Megaparsec tutorial
https://hackage.haskell.org/package/parsec-3.1.16.1/docs/Text-ParserCombinators-Parsec-Char.html#v:satisfy | Text.ParserCombinators.Parsec.Char
http://jakewheat.github.io/intro_to_parsing/#very-simple-expression-parsing | Intro to Parsing with Parsec in Haskell
https://stackoverflow.com/questions/16154592/haskell-no-instance-for-eq-a-arising-from-a-use-of | Haskell: No instance for (Eq a) arising from a use of `==' - Stack Overflow
https://www.google.com/search?q=*%20Couldn%27t%20match%20type%20%60Char%27%20with%20%60Text.Megaparsec.Internal.ParsecT%20Void%20String%20Data.Functor.Identity.Identity%20Char%27 | * Couldn't match type `Char' with `Text.Megaparsec.Internal.ParsecT Void String Data.Functor.Identity.Identity Char' - Google Search
https://hackage.haskell.org/package/megaparsec-9.0.1/docs/Text-Megaparsec.html#g:2 | Text.Megaparsec
https://www.ecosia.org/search?q=how%20to%20parse%20a%20regex%20in%20haskell%20into%20an%20ast&addon=chrome&addonversion=5.1.3&method=topbar | how to parse a regex in haskell into an ast - Ecosia - Web
https://hackage.haskell.org/package/megaparsec-9.3.0/docs/Text-Megaparsec-Char-Lexer.html | Text.Megaparsec.Char.Lexer
https://hackage.haskell.org/package/parsec-3.1.16.1/docs/Text-Parsec.html#g:2 | Text.Parsec
https://hackage.haskell.org/package/regex-tdfa-1.3.2/docs/Text-Regex-TDFA.html | Text.Regex.TDFA
https://regex101.com/ | regex101: build, test, and debug regex

https://www.google.com/search?q=regular%20expression%20alternation%20empty%20string%20and%20empty%20set | regular expression alternation empty string and empty set - Google Search
https://www.haskell.org/tutorial/patterns.html | A Gentle Introduction to Haskell: Patterns
https://verigu.github.io/4115Spring2022/assignments/hw1.html | Homework 1 | 4115Spring2022
https://stackoverflow.com/questions/37286376/what-does-mean-in-haskell | What does <$> mean in Haskell? - Stack Overflow
https://stackoverflow.com/questions/3479116/pattern-matching-variables-in-a-case-statement-in-haskell | string - Pattern matching variables in a case statement in Haskell - Stack Overflow
https://stackoverflow.com/questions/12934730/iterating-through-a-string-and-replacing-single-chars-with-substrings-in-haskell | Iterating through a String and replacing single chars with substrings in haskell - Stack Overflow
https://www.google.com/search?q=fold%20left%20in%20haskell%20explained | fold left in haskell explained - Google Search
https://stackoverflow.com/questions/14397128/how-does-the-flip-function-work | haskell - How does the flip function work? - Stack Overflow
http://www1.cs.columbia.edu/~sedwards/classes/2020/4995-fall/index.html | COMS W4995 003 Parallel Functional Programming
http://www1.cs.columbia.edu/~sedwards/classes/2020/4995-fall/basics.pdf | basics.pdf
http://www1.cs.columbia.edu/~sedwards/classes/2020/4995-fall/stack.pdf | stack.pdf
http://www.cs.columbia.edu/~sedwards/classes/2022/4995-fall/io.pdf | io.pdf
https://hackage.haskell.org/package/CheatSheet-1.11/src/CheatSheet.pdf | CheatSheet.pdf
https://www.haskell.org/tutorial/functions.html | A Gentle Introduction to Haskell: Functions
https://wiki.haskell.org/Function_composition | Function composition - HaskellWiki
http://www.sfu.ca/~tjd/383summer2019/haskell_folding_lhs.html | Folding in Haskell — 383summer2019 documentation
https://stackoverflow.com/questions/65712718/is-there-any-way-to-print-a-variable-inside-a-function-in-haskell | functional programming - Is there any way to print a variable inside a function in Haskell? - Stack Overflow
https://hackage.haskell.org/package/base-4.18.0.0/docs/Data-List.html#v:unfoldr | Data.List
https://stackoverflow.com/questions/19802039/remove-the-first-and-the-last-element-from-a-list-in-haskell | Remove the first and the last element from a list in Haskell - Stack Overflow
https://stackoverflow.com/questions/12461871/print-list-elements-in-new-lines | haskell - Print list elements in new lines - Stack Overflow
https://stackoverflow.com/questions/27609062/what-is-the-difference-between-mapm-and-mapm-in-haskell | What is the difference between mapM_ and mapM in Haskell? - Stack Overflow
https://arxiv.org/pdf/1908.05839v2.pdf | Bidirectional Typing
https://dl.acm.org/doi/10.1145/2500365.2500582 | Complete and easy bidirectional typechecking for higher-rank polymorphism | Proceedings of the 18th ACM SIGPLAN international conference on Functional programming
https://dl.acm.org/doi/pdf/10.1145/2500365.2500582 | 2500365.2500582
https://www.sciencedirect.com/science/article/pii/0167642387900207 | The categorical abstract machine - ScienceDirect

https://dl.acm.org/doi/10.1145/800194.805852 | Definitional interpreters for higher-order programming languages | Proceedings of the ACM annual conference - Volume 2
https://dl.acm.org/doi/pdf/10.1145/800194.805852 | 800194.805852
https://johnwickerson.wordpress.com/2017/08/15/pl-facts/ | Who has the most POPL and PLDI papers? – Wickopedia
https://felleisen.org/matthias/papers.html#popl16-tfgnvf | Publications
https://www2.ccs.neu.edu/racket/pubs/popl16-tfgnvf.pdf | popl16-tfgnvf.pdf
http://www1.cs.columbia.edu/~sedwards/classes/2020/4995-fall/io.pdf | io.pdf
http://www1.cs.columbia.edu/~sedwards/classes/2023/6998-spring-tlc/regexp.pdf | Strings and Regular Expressions
https://stackoverflow.com/questions/7376937/fastest-way-to-get-the-last-element-of-a-list-in-haskell | Fastest way to get the last element of a list in Haskell - Stack Overflow
https://stackoverflow.com/questions/37019716/how-do-we-convert-boolean-to-string-in-haskell | How do we convert boolean to string in Haskell - Stack Overflow
https://stackoverflow.com/questions/11487080/how-do-i-properly-define-an-empty-string-in-haskell | list - How do I properly define an empty string in haskell? - Stack Overflow
https://stackoverflow.com/questions/64320952/how-to-parse-a-string-in-haskell | How to parse a string in haskell? - Stack Overflow
https://stackoverflow.com/questions/51017223/can-i-define-more-than-one-function-under-the-where-syntax | haskell - Can I define more than one function under the where syntax? - Stack Overflow
https://www.educba.com/haskell-string/ | Haskell String | How String Type Works in Haskell | Examples
https://stackoverflow.com/questions/6168739/char-to-string-function | haskell - Char to string function - Stack Overflow
https://stackoverflow.com/questions/1874828/type-matching-in-haskell | Type Matching in Haskell - Stack Overflow
https://superuser.com/questions/790353/how-to-comment-out-multiple-lines-at-once-in-notepad | keyboard shortcuts - How to comment out multiple lines at once in Notepad++? - Super User
https://stackoverflow.com/questions/6700570/haskell-and-string-length | haskell and string length - Stack Overflow
https://stackoverflow.com/questions/5710078/in-haskell-performing-and-and-or-for-boolean-functions | combinators - In Haskell performing `and` and `or` for boolean functions - Stack Overflow
https://stackoverflow.com/questions/6070316/how-to-throw-an-exception-and-exit-the-program-in-haskell | How to throw an exception and exit the program in Haskell? - Stack Overflow
https://stackoverflow.com/questions/6009384/exception-handling-in-haskell | Exception handling in Haskell - Stack Overflow
https://www.haskelltutorials.com/guides/haskell-lists-ultimate-guide.html | Haskell Lists: The Ultimate Guide
https://riptutorial.com/haskell/example/16553/foldl | Haskell Language Tutorial => foldl
https://www.educba.com/haskell-not-equal/ | Haskell not equal | Learn How does not equal operator work?
https://www.educative.io/answers/how-to-get-the-first-char-from-a-string-in-haskell | How to get the first char from a string in Haskell

https://math.stackexchange.com/questions/472952/understanding-concatenating-the-empty-set-to-any-set | elementary set theory - Understanding concatenating the empty set to any set. - Mathematics Stack Exchange
https://www.google.com/search?q=brzoswki%20derivative%20calculator | brzoswki derivative calculator - Google Search
https://lcs.ios.ac.cn/~chm/papers/derivative-tr200910.pdf | derivative-tr200910.pdf
https://github.com/AlexanderJDupree/DFA-Check/blob/master/src/DFA.hs | DFA-Check/DFA.hs at master · AlexanderJDupree/DFA-Check · GitHub
https://github.com/AlexanderJDupree/DFA-Check | AlexanderJDupree/DFA-Check: Verification framework for Deterministic Finite Automaton (DFA), written in Haskell.
https://github.com/cacay/regexp/blob/master/src/RegExp/RegExp.hs | regexp/RegExp.hs at master · cacay/regexp · GitHub
https://github.com/cacay/regexp/tree/master/src | regexp/src at master · cacay/regexp · GitHub
https://dl.acm.org/doi/pdf/10.1145/321239.321249 | Derivatives of Regular Expressions
https://www.ccs.neu.edu/home/turon/re-deriv.pdf | re-deriv.pdf
https://homepages.dcc.ufmg.br/~camarao/fp/mimico/source/hel/parsec.html#ReferenceGuide | Parsec, a fast combinator parser (Leijen; 2001).pdf
https://gist.github.com/GuiBrandt/d8257426eb9947d3c478e582288128fb | Regular Expressions in haskell
https://stackoverflow.com/questions/26230759/ast-and-parsing-in-haskell | AST and Parsing in Haskell - Stack Overflow
https://smunix.github.io/dev.stephendiehl.com/fun/WYAH.pdf | Write You a Haskell
https://www.google.com/search?q=is%20the%20empty%20string%20in%20the%20alphabet%20of%20a%20regex%3F | is the empty string in the alphabet of a regex? - Google Search
https://math.stackexchange.com/questions/1689850/is-the-empty-string-always-in-a-finite-alphabet | logic - Is the empty string always in a finite alphabet? - Mathematics Stack Exchange
https://www.ecosia.org/search?q=how%20to%20unpack%20a%20just%20value%20in%20haskell&addon=chrome&addonversion=5.1.3&method=topbar | how to unpack a just value in haskell - Ecosia - Web
https://stackoverflow.com/questions/18808258/what-does-the-just-syntax-mean-in-haskell | What does the "Just" syntax mean in Haskell? - Stack Overflow
https://stackoverflow.com/questions/4940349/how-to-get-the-value-of-a-maybe-in-haskell | How to get the value of a Maybe in Haskell - Stack Overflow
https://stackoverflow.com/questions/12909456/haskell-how-to-print-value-and-return-in-function | iteration - Haskell, how to print value and return in function - Stack Overflow
https://www.educative.io/answers/how-to-append-an-element-to-a-haskell-list | How to append an element to a Haskell list
https://stackoverflow.com/questions/29089117/haskell-new-line-not-working | newline - Haskell new line not working - Stack Overflow