

Bei diesem Aufgabenblatt handelt es sich um ein Mini-Projekt, in welchem die Themen Aussagenlogik und Grammatiken praktisch angewandt werden. Das Projekt gliedert sich in drei Teile.

- 1) Entwickeln Sie einen Parser für aussagenlogische Formeln. Verwenden Sie hierzu einen Parsergenerator (z.B. *Lex & Yacc* mit C oder *Happy*¹ mit Haskell), welcher auf eine von Ihnen angegebene Grammatik basiert.

Hinweis: Beachten Sie, dass dieser Parser im Gegensatz zum Beispiel in der Vorlesung die Stringrepräsentation der Formeln nicht direkt auswerten, sondern in eine adäquate Datenstruktur umwandeln soll.

Hinweis: Falls Sie sich für Haskell entscheiden, können Sie unsere Vorlage verwenden, in welcher die Zieldatenstruktur bereits vorgegeben ist. Mit dem Befehl `happy [file].y` erhalten sie ein Haskell-Modul, welches in unserem Hauptmodul bereits eingebunden ist. Der Parser kann dann durch die entsprechenden Funktionen `parser` sowie `lexer` im Haskell-Programm verwendet werden.

- 2) Implementieren Sie folgende Funktionen für aussagenlogische Formeln:
 - a) Eine Evaluierung von Formeln unter der Angabe einer Belegung
 - b) Eine Vereinfachung von Formeln gemäß der in der Vorlesung besprochenen Äquivalenzen
 - c) Die Methode von Quine

Denken Sie bei der Implementierung der Methode von Quine daran, dass die ersten beiden Funktionen hilfreich sein könnten.

- 3) Verbessern Sie Ihren Parser dahingehend, dass auch Stringrepräsentationen mit einer minimalen Anzahl an Klammern akzeptiert werden (d. h. der Parser soll die in der Vorlesung besprochenen Konventionen bezüglich Präzedenz und Assoziativität berücksichtigen).

Laden Sie Ihre Lösung in den entsprechenden Abgabeordner auf OLAT hoch.

¹<https://www.haskell.org/happy/>