

## 4. Übung „Angewandte Bioinformatik mit Perl und R“

1. Entwerfen und implementieren Sie eine Klassenhierarchie für Sequenzen, die folgendes ermöglicht:
  - Sequenzobjekte bestehen allgemein aus ihrer Sequenz und gegebenenfalls einem Namen (einer ID, ...), sowohl Sequenz als auch Name sollen abgerufen werden können.
  - Eine Sequenz soll „wissen“, ob sie eine DNA-, eine RNA-, oder eine Protein-Sequenz ist.
  - Von DNA- und RNA-Sequenzen kann man das reverse Komplement bilden, von Protein-Sequenzen nicht.
  - DNA-Sequenzen können in RNA-Sequenzen transkribiert werden.
  - RNA-Sequenzen können in DNA-Sequenzen rück-transkribiert werden (reverse Transkriptase).
  - RNA-Sequenzen können in Protein-Sequenzen translatiert werden, DNA-Sequenzen nicht (direkt).

Außerdem soll es eine Klassen-Methode geben, die Sequenzen aus einer FastA-Datei einliest und eine Liste der entsprechenden Sequenz-Objekte zurückgibt.

Versuchen Sie bei der gesamten Implementierung, Code-Doppelungen zu vermeiden und die Vererbungshierarchie möglichst effizient zu nutzen. (8 Punkte)

2. Machen Sie sich mit der Dokumentation von BioPerl vertraut. Vergleichen Sie die Funktionalität Ihrer Sequenz-Klassen aus Aufgabe 1 mit derer der Klassen `PrimarySeq` und `Seq`. (2 Punkte)