

# Vorlesung Information und Codierung (INCO)

*Dr. Alexander Herrigel*

*HS 2014, ZHAW*

## Praktikum LZ77 Codierung/Decodierung (03.11.2014)

### Zusammenfassung

Dieses Praktikum behandelt die Codierung und Decodierung von Substitutionsverfahren. Es soll von Ihnen das LZ77-Verfahren, das in der Vorlesung besprochen wurde, für die Codierung und Decodierung implementiert werden. Es ist speziell zu beachten, dass die codierten Ergebnisse in einem binären File abgespeichert werden und zur Decodierung von diesem wieder gelesen werden.

### Einleitung

Das LZ77-Verfahren zur Codierung und Decodierung wurde in der Vorlesung behandelt.

### Dateien und Eclipse-Projekt

Es steht Ihnen bereits ein Java-Programm und ein Eclipse-Projekt zur Verfügung, das Ihnen als Template dient und es Ihnen erlaubt, sich auf das Wesentliche der Aufgabenstellung zu konzentrieren. Wie die anderen Template können Sie das Template von Github [Template-Link](#) herunterladen. Die in dem Praktikum verwendeten Daten-Dateien sind die gleichen, wie in dem letzten Praktikum.

### Aufgabe

Vervollständigen Sie das Eclipse-Template Projekt so, dass Sie die LZ77 Codierung und Decodierung vollständig unterstützen können.

- Das Programm-Template benutzt zwei Command-Line Parameter, um die Komprimierung und Dekomprimierung auszuführen. Mit dem Command-Line Parameter *e* und dem Dateinamen der zu komprimierenden Datei wird die Komprimierung, mit dem Command-Line Parameter *d* und dem Dateinamen der komprimierten Datei wird die Dekomprimierung aufgerufen. Wenn ein File vollständig gelesen und komprimiert wurde, wird das Ergebnis in `Dateiname.lz77encoded` abgespeichert. Wenn eine komprimierte Datei gelesen wird, wird das Ergebnis in der Datei `Dateiname.lz77decoded` abgespeichert. Die komprimierten Daten werden in einer binäre Datei abgespeichert.
- Das Template besitzt 5 verschiedene Klassen, die Klasse `UserErrorException`, die Klasse `ProgramParameterParser`, die Klasse `FoundStringEncData`, `LZMain` und die Klasse `LZ77Encoder`. Für die Implementierung ist für Sie nur die Klasse `LZ77Encoder` interessant. In dieser Klasse gibt es zwei Methoden, `encode` und `decode`, die Sie noch ergänzen müssen.
- Die zwei Aufgaben finden Sie in den Kommentaren von `ToDo [1.1]` und `ToDo [1.2]`.
- Wie beim letzten Praktikum können Sie das Programm ohne zusätzliche Programmierung ausführen. Werden jedoch die Methoden mit den fehlenden Implementierungen ausgeführt, wird eine Exception erzeugt.
- Wenn Sie eine entsprechende Run-Configuration in Eclipse spezifizieren, können Sie für die verschiedenen Dateien das Programm bequem in dem Eclipse IDE ausführen (siehe letztes Praktikum).

- Mit Hilfe der Datei Muster.txt sollten Sie einfach in der Lage sein zu eruieren, ob ihr Java-Code wirklich die Logik ausführt, die der Code ausführen sollte.
- Ihr Programm sollte für alle Dateien fehlerfrei die Komprimierung und Dekomprimierung durchführen.

## **Wichtige Bedingungen für die Abgabe**

- Es ist es erforderlich, dass für jede Abgabe ein entsprechendes Repository auf github eingerichtet wird, damit der Code einfach eingesehen werden kann.
- Es gilt die folgende Namenskonvention für das Repository: ZHAW\_INCO\_LZ77\_Vorname\_Nachname.
- Falls eine Gruppenabgabe von zwei Personen stattfindet, gilt die folgende Konvention für das Repository: ZHAW\_INCO\_Praktikum\_LZ77\_Vorname1\_Nachname1\_Vorname2\_Nachname2.
- Es sind auf ihrem github Repository alle Source Dateien und Dokumente abzuspeichern, die für die Abgabe relevant sind.
- Es wird erwartet, dass im Minimum alle Dateien abgespeichert werden, die für ein Eclipse-Projekt erforderlich sind.