Musterlösung zu Blatt 10:

1./2./3./4. Holen Sie die ersten 4 Bücher (Wenn Sie wollen, holen Sie alle 66 Bücher) der Bibel mit dem UNIX Befehl wget : wget "http://gutenberg.spiegel.de/buch/5560/l" -O l.html für [l=1,..,4] und speichern Sie von der Kommandozeile aus, automatisch den Inhalt der www-Seite in den lokalen Dateien 1.html, 2.html, ... 4.html Überprüfen Sie die heruntergeladenen Bücher mit einem WWW-Browser. Verwenden sie den Lynx Befehl um die Bücher, die in den .html Dateien gespeichert sind in eine Textdatei zu konvertieren. Fügen Sie alle Bücher zu einer Datei bibel.txt zusammen.

```
wget "http://gutenberg.spiegel.de/buch/5560/1" -O 1.html
wget "http://gutenberg.spiegel.de/buch/5560/2" -O 2.html
wget "http://gutenberg.spiegel.de/buch/5560/3" -O 3.html
wget "http://gutenberg.spiegel.de/buch/5560/4" -O 4.html

lynx -dump -assume_charset=UTF-8 -hiddenlinks=ignore -nolist -verbose
1.html 2.html 3.html 4.html > bibel.txt
```

5. Wie kann man die Aufgabe 1-4 innerhalb eines PERL-Programms realisieren (Tipp: PERL- Routine system("wget ....) und sichern der Ausgabe des Befehls in einer Datei mit redirect)

```
#!/usr/bin/perl
# Autor: Nicola Greth
# Programm: laedt Buecher herunter und fuegt diese zu bibel.txt zusammen
use strict;
use locale;
use utf8;
{
  my $bibel="";
  open(OUT, ">:utf8", "bibel.txt");
  my $anzahl_buecher =4;

  for(my $i=1; $i<=$anzahl_buecher; $i++){
    my $url ="wget \"http://gutenberg.spiegel.de/buch/5560/$i\" -O $i.html";
    system($url);

    my $datei = "temp.txt";
    system("lynx -dump $i.html -assume_charset=UTF-8 -hiddenlinks=ignore
-nolist -verbose > $datei");
    open(IN, "<:utf8", $datei) or die "Datei $datei nicht gefunden!\n";

    undef $/; #damit man alles auf einmal lesen kann
    my $buch=<IN>;
    close(IN);
    system("rm $datei");

    $bibel = $bibel.$buch;
  }

  print(OUT $bibel);
  close(OUT); }
```

6./7./8./9. Erzeugen Sie eine sortierte Frequenzliste aller Wörter aus der Datei bibel.txt. Wieviele unterschiedliche Wörter kommen in der Bibel vor? Was sind die 10 wichtigsten großgeschriebenen Wörter? Was sind die 10 wichtigsten kleingeschriebenen Wörter?

```
#!/usr/bin/perl
# Autor: Nicola Greth
# Programm: erzeugt Frequenzliste und berechnet bestimmte Dinge
use strict;
use locale;
use utf8;
{
  my($datei, $zeile, @woerter, $wort, %lexikon);
  $datei = "bibel.txt";
  open (IN,$datei) or die "File $datei not found!";

  while($zeile = <IN>) {
    chomp($zeile);
    @woerter = split(/[\p{Z}\p{P}]+/,$zeile);

    foreach $wort (@woerter) {
      $lexikon{$wort}++;
    }
  }
  close(IN);

    my @woerter_sortiert = sort {$lexikon{$b} <=> $lexikon{$a}} keys %lexikon;
  my $anzahl = scalar @woerter_sortiert;
  print "In der Bibel sind $anzahl unterschiedlicher Wörter.\n";


  my $zaehler = 0;
  print "Die zehn wichtigsten großgeschriebenen Wörter sind:\n";

  foreach $wort (@woerter_sortiert) {
    if ($zaehler < 10 && $wort =~ /^\p{Lu}/){
      print "$lexikon{$wort}: $wort\n";
      $zaehler++;
    }
  }

  print "Die zehn wichtigsten kleingeschriebenen Wörter sind:\n";
  $zaehler = 0;

  foreach $wort (@woerter_sortiert) {
    if ($zaehler < 10 && $wort =~ /^\p{Ll}/){
      print "$lexikon{$wort}: $wort\n";
      $zaehler++;
    }
  }
}
```

10. Ermitteln Sie das längste Wort einer Zeile mit der Subroutine: longest_word()

```
#!/usr/bin/perl
# Autor: Nicola Greth
# Programm: schreibt Subroutine, die laengstes Wort aus einer Zeile
zurueckgibt

use strict;
use utf8;
use locale;
{
  my ($zeile, $laengstes);

  print "Bitte geben Sie eine Textzeile ein >>> ";
  chomp($zeile = <>);

  $laengstes = &longest_word($zeile);
  print "Das laengste Wort ist $laengstes.\n";
}

sub longest_word($) {
  my $zeile = $_[0];
  my $max = "";
  my @woerter = split(/[\p{Z}\p{P}]+/,$zeile);
  my $wort;
  foreach $wort (@woerter){
    if ( ($wort =~/^\p{L}+$/) && (length($wort) > length($max)) ){
      $max = $wort;
    }
  }
  return $max;
}
```

11. Ermitteln Sie die Anzahl der Wörter in einer Zeile mit der Subroutine: count_words()

```
#!/usr/bin/perl
# Autor: Nicola Greth
# Programm: schreibt Subroutine, die Anzahl der Woerter einer Zeile
zurueckgibt

use strict;
use utf8;
use locale;
{
  my ($zeile, $anzahl);

  print "Bitte geben Sie eine Textzeile ein >>> ";
  chomp($zeile = <>);

  $anzahl = &count_words($zeile);
  print "Die Zeile hat $anzahl Woerter.\n";  }
```

```
sub count_words($) {
  my $zeile = $_[0];
  my @woerter = split(/[\p{Z}\p{P}]+/,$zeile);
  return scalar (@woerter);
}
```

12. Verwenden Sie den Satzendeerkenner um die Datei in Sätze aufzuspalten

Text aus der Datei wird in Satzendeerkenner kopiert. Zurückgegebener Text wird in Datei bibel_satzend.txt kopiert.

13. Was sind die 10 häufigsten Wörter am Satzanfang

```
#!/usr/bin/perl
# Autor: Nicola Greth
# Programm: gibt die zehn haeufigsten Woerter am Satzanfang aus
use strict;
use locale;
use utf8;
{
  my %lexikon;
  my $datei = "bibel_satzend.txt";
  open (IN,$datei) or die "File $datei not found!";

  undef $/;# um gesamten Text in einen String einzulesen
  my $text =<IN>;
  close(IN);

  $text =~ s/\n/ /g;  # Newlines durch Space erzetzen
  $text =~ s/\s+/ /g; # Mehrere Spaces durch ein Space ersetzen

  my @saetze = split("{eos}", $text);

  foreach my $satz (@saetze){
    if ($satz =~/^\s*(\p{L}+)\b/) {
      $lexikon{$1}++;
    }
  }

  my @woerter_sortiert = sort {$lexikon{$b} <=> $lexikon{$a}} keys %lexikon;

  print "Die zehn haeufigsten Wörter am Satzanfang aus der Datei sind:\n";
  for(my $i=0; $i<10; $i++){
    print "$lexikon{$woerter_sortiert[$i]}: $woerter_sortiert[$i]\n";
  }
}
```

14. Wie lautet ein PERL-Programm, das die ersten 3 Zeilen eines jeden Kapitels von jedem Buch ausgibt. Die Ausgabe soll folgende Form haben

```
#!/usr/bin/perl
# Autor: Nicola Greth
# Programm: gibt die ersten drei Zeilen jeden Kapitels von jedem Buch aus
use strict;
use locale;
use utf8;
{
  my $zeile;
  my $datei = "bibel_satzend.txt";
  open (IN,$datei) or die "File $datei not found!";

  my $zaehler = 0;

  while($zeile = <IN>) {
    chomp($zeile);
    $zeile =~ s/\s+/ /g;

    if (($zeile =~ /[1-4]\. Buch Mose/) || ($zeile =~ /^Kapitel \p{N}+$/)) {
      print "$zeile\n";
      $zaehler = 0;
    }
    else {
      if ( $zeile =~ /^\p{Z}*\p{N}\./ ) {
        $zaehler++;

        if (($zaehler <= 3) && ($zeile ne "")) {
          print "$zeile\n"
        }
      }
    }
  }

  close(IN);
}
```

15. Welche morpholgischen Varianten des Wortes Herr gibt es und wie oft kommen die verschiedenen Varianten des Wortes Herr vor?

```
#!/usr/bin/perl
# Autor: Nicola Greth
# Programm: gibt morphologische Varianten des Wortes Herr aus
use strict;
use locale;
use utf8;
{
  my ($zeile, @woerter, $wort, %lexikon);
  my $datei = "bibel_satzend.txt";
  open (IN,$datei) or die "File $datei not found!";

  while($zeile = <IN>) {
    chomp($zeile);
    @woerter = split(/[\p{Z}\p{P}]+/,$zeile);

    foreach $wort (@woerter) {
      if($wort =~ /^Herr/i) {
        $lexikon{$wort}++;
      }
    }
  }

  my @woerter_sortiert = sort {$lexikon{$b} <=> $lexikon{$a}} keys %lexikon;
  print "Morpholgischen Varianten und die Haeufigkeit des Wortes \"Herr\":\n";

  foreach $wort (@woerter_sortiert){
    print "$lexikon{$wort}: $wort\n";
  }

  close(IN);
}
```