

Einführung in die Computerlinguistik – Einführung in Perl (2) – Regelbasierte Dialogsysteme

Dozentin: Wiebke Petersen

1.12.2009

reguläre Ausdrücke in Perl

```
1  #!/perl -w
2  print "Please␣enter␣password:␣";
3  $password = <>;
4  chomp($password);
5  # While the entered line doesn't contain "42"
6  while ($password !~ /42/) {
7  print "Access␣denied.\n";
8  print "Please␣enter␣password:␣";
9  $password = <>;
10 chomp($password);
11 }
12 print "Correct␣password!␣Welcome.␣";
```

- reguläre Ausdrücke in Perl werden wie folgt eingeschlossen: `"/regex/`
- `!~` fordert, dass der reguläre Ausdruck nirgendwo im String matcht
- `=~` fordert, dass er irgendwo im String matcht
- Schreiben Sie ein Programm, das Passwörter akzeptiert, die mindestens drei Großbuchstaben enthalten.

Schalter und Ersetzen

Schalter

- `/regex/g` führt die Suche nach `regex` wiederholt bis zum Ende des Strings aus.
- `/regex/i` ignoriert Groß- und Kleinschreibung.

Ersetzen

`s/regex1/regex2/` ersetzt `regex1` durch `regex2`.

```
1 $text = "The meat costs 10 Mark, the fish costs 15 Mark.";
2 $text =~ s/Mark/Euro/;
3 # "The meat costs 10 Euro, the fish costs 15 Mark."
```

```
1 $text = "The meat costs 10 Mark, the fish costs 15 Mark.";
2 $text =~ s/Mark/Euro/g;
3 # "The meat costs 10 Euro, the fish costs 15 Euro."
```

Gruppieren und Speichern

- `/(text)/` findet und speichert `text`
- Der erste Klammerinhalt wird in `$1`, der zweite in `$2` u.s.w. gespeichert.

```
1 # Substitution and grouping
2 $sumM = 0;      # initializing the variable with zero
3 $sumE = 0;      # initializing the variable with zero
4 $text = "The meat costs 10 Mark, the fish costs 15 Mark.";
5 while ($text =~ /(\d+) Mark/g) {
6   $sumM = $sumM + $1;
7   $sumE = $sumE + $1/2;
8 }
9 print "total: $sumM Mark or $sumE Euro";
```

Dialogsystem: Eliza (Weizenbaum 1966)

User: Men are all alike

ELIZA: IN WHAT WAY

User: They're always bugging us about something or other

ELIZA: CAN YOU THINK OF A SPECIFIC EXAMPLE?

User: Well, my boyfriend made me come here

ELIZA: YOUR BOYFRIEND MADE YOU COME HERE

User: He says I'm depressed much of the time

ELIZA: I AM SORRY TO HEAR THAT YOU ARE DEPRESSED

Implementierungen:

- <http://www.hsweis.schule.neu-ulm.de/eliza/eliza3.html>
- <http://bs.cyty.com/menschen/e-etzold/archiv/science/rat.htm>
- <http://folk.uib.no/hhiso/eliza/index.htm>

Dialogsystem: Eliza (Weizenbaum 1966)

① Wechsel von 1. zur 2. Person:

```
1 s/\bI('m|_am)\b_/you_/g
2 s/\bmy\b_/your/g
3 s/\bmine\b_/yours/g
```

② Generierung von Antworten:

```
1 s/.* you are (depressed|sad) .*/i am sorry to hear you are $1/
2 s/.* you are (depressed|sad) .*/why do you think you are $1/
3 s/.* all .*/in what way/
4 s/.* always .*/can you think of a specific example/
```

Übungseinheit (4)

- Schreiben Sie mithilfe der Angaben der vorangegangenen Folie ein Eliza-Programm.
- Wählen Sie sich ein beliebiges Themengebiet und schreiben Sie ein einfaches Dialogsystem (Beispiel: Fahrkarten-, Theaterkartenkauf).

Weiterführende Informationen

- Über den Turingtest
<http://plato.stanford.edu/entries/turing-test/>
- Zum Loebner Preis
<http://www.loebner.net/Prizef/loebner-prize.html>
- Kritik am Loebner Preis von Stuart Shieber
<http://www.eecs.harvard.edu/shieber/Biblio/Papers/loebner-rev-html/loebner-rev-html.html>
- Chatbot eines der Preisträger:
www.elbot.de