

Schlussprüfung Programmiertechniken in der Computerlinguistik I HS 11

Aufgabenstellung: Martin Volk/Simon Clematide

Prüfung vom 12. Januar 2012
Institut für Computerlinguistik, Universität Zürich

Vorname _____ Matrikelnummer _____

Nachname _____

Für Studierende der folgenden Studiengänge:

- ☐ BA - Studiengang Computerlinguistik (Phil. Fakultät)
- ☐ BA - Studiengang Computerlinguistik und Sprachtechnologie (Phil. Fakultät)
- ☐ BA-Studierende (Wirtschaftswissenschaftliche Fakultät)
- ☐ Studierende des Nebenfachs Informatik mit Studienbeginn ab WS 04/05
- ☐ Multidisziplinär (ETH)
- ☐ Andere:

Nur für Lizentiatsstudierende der Computerlinguistik als ein Fach aus der Phil. Fakultät:

Strasse: _____ Hauptfach: _____

PLZ/Ort: _____ E-Mail: _____

Die Prüfungsergebnisse von ECL und PCL von den Lizentiatsstudierenden werden (zusammen) per Post verschickt. *Bitte Adresse nicht vergessen!*

Aufgabe Nr.:	1	2	3	4	5	6	7	8	Summe
Punktzahl:	7	15	21	9	3	4	11	20	90
Davon erreicht:									

Note SU: _____ Note SP: _____

Endnote: _____ Bestanden: ☐ Ja ☐ Nein

Auf jedes separate Blatt mit Lösungen den Nachnamen schreiben!

Viel Erfolg!

3. Theoriefragen (21 Punkte)

4 (a) Was sind die Vorteile oder Nachteile von Textdateien in UTF-8 gegenüber Textdateien in Latin-1?

[illegible]

4 (b) Was sind die Vorteile oder Nachteile von Listenkompensation?

[illegible]

5 (c) Für welche Programmieraufgaben eignen sich Hashes (= Dictionaries) besser als Listen und umgekehrt? Charakterisiere einerseits die abstrakten Eigenschaften und gebe andererseits gute Beispiele dafür.

[illegible]

4 (d) Wie unterscheiden sich Methoden von Funktionen?

.....

.....

.....

.....

.....

.....

.....

.....

.....

4 (e) Erkläre kurz das Wesentliche, was Python als **objektorientierte** Sprache auszeichnet.

.....

.....

.....

.....

.....

.....

.....

.....

.....

9 4. **Reguläre Ausdrücke (9 Punkte)** Gegeben sei der folgende reguläre Ausdruck in Python:

```
pattern = r'\$?\d+[\.\d]*%?'
```

Kreuze das Kästchen an, falls dieser Ausdruck mittels `re.search(pattern, string)` in folgenden Strings matcht und schreibe die erste gematchte Zeichenkette auf die freie Linie. (Die einfachen Anführungszeichen begrenzen den String und gehören nicht zu den zu matchenden Zeichen.)

<input type="checkbox"/> '12,123%'	_____
<input type="checkbox"/> '\$ 12.7'	_____
<input type="checkbox"/> '12.789.123%%'	_____
<input type="checkbox"/> 'Dollars (\$)'	_____
<input type="checkbox"/> '\$.11%'	_____
<input type="checkbox"/> '1\$1'	_____

- 3 5. Programm kommentieren (3 Punkte)** Was macht das folgende Python-Programm? Kommentiere die 3 Zeilen und beschreibe damit die Programmfunktionalität.

```
text = 'programmieren macht spass'

for position in range(len(text)):

    print position, text[position]
```

- 6. Listenoperationen in Python (4 Punkte)** Gegeben sei eine Liste mit 5 Wörtern:

```
chomskys_sentence = ['colorless', 'green', 'ideas', 'sleep', 'well']
```

- 1 (a)** Welche Anweisung ersetzt das Wort 'well' in dieser Liste durch das Wort 'furiously'?

.....

- 1 (b)** Dann weiter: Welche Anweisung fügt die Liste
 martins_sentence = ['but', 'my', 'ideas', 'sleep', 'much', 'better']
 an die obige Liste (mit ersetzttem Wort) an, so dass die folgende Gesamtlste entsteht:

```
long_sentence = ['colorless', 'green', 'ideas', 'sleep', 'furiously',  
                'but', 'my', 'ideas', 'sleep', 'much', 'better']
```

.....

- 1 (c)** Wie kann man die Liste long_sentence in Python alphabetisch sortieren, so dass das Ergebnis so aussieht?

```
long_sentence_sorted = ['better', 'but', 'colorless', 'furiously',  
                      'green', 'ideas', 'ideas', 'much', 'my', 'sleep', 'sleep']
```

.....

- 1 (d)** Wie kann man die Liste long_sentence_sorted in Python bereinigen, so dass jedes Wort nur einmal in der Liste vorkommt und das Ergebnis so aussieht?

```
long_sentence_sorted_and_cleaned = ['better', 'but', 'colorless',  
                                   'furiously', 'green', 'ideas', 'much', 'my', 'sleep']
```

.....

7. Dictionaries (11 Punkte) Gegeben sei ein Dictionary (Hash) mit 5 Wörtern und deren Wortart:

```
chomskys_words = {'colorless': 'ADJ', 'green': 'ADJ',
                  'ideas': 'NN', 'sleep': 'V', 'well': 'ADV'}
```

- 3 (a) Welche Anweisungen ersetzen das Wort 'well' durch das Wort 'furiously' mit der Wortart 'ADV'?

.....

- 3 (b) Gegeben sei ein weiteres Hash mit 6 Wörtern und deren Wortarten:

```
martins_words = {'but': 'CON', 'my': 'PRON', 'ideas': 'NN',
                 'sleep': 'V', 'much': 'ADV', 'better': 'ADV'}
```

Wie führt man alle Wörter aus beiden Hashes in ein neues Hash `all_my_words` zusammen?

.....

- 1 (c) Wie viele Schlüssel-Wert-Paare enthält das Hash `all_my_words`? _____

- 1 (d) In welcher Reihenfolge stehen die Schlüssel-Wert-Paare in diesem Hash `all_my_words`?

.....

- 3 (e) Was passiert, wenn du in dieses Hash `all_my_words` zusätzlich den Eintrag 'green' (im Sinne von *das Golfgrün*) mit der Wortart 'NN' einträgst? Wie könnte man dabei allfällig entstehende Probleme lösen?

.....

