LIN 177 Homework 4

Hardy Jones

999397426

Professor Ojeda

Winter 2015

1. aɪ lʌv gɔlfɪŋ. ənfɔrtʃunətli, aɪ hæv nɑt bɪn gɔlfɪŋ ɪn jɪrz. maɪ skɔrz wɜr əbɪzməl. aɪ rutinli ʃɑt ɪn ðə hʌndrədz, bʌt aɪ wʊd præktəs rɪlɪdʒəsli. wɛn aɪ lɪvd ɪn waʃɪŋtən steɪt, aɪ juzd tu goʊ tu ðə draɪvɪŋ reɪndʒ æt list θri taɪmz ə wik. sʌn, reɪn, wətɛvər, aɪ wʊd bi æt ðə reɪndʒ præktesɪŋ maɪ pətɪŋ, draɪvɪŋ, or ʧɪpɪŋ. aɪ nɛvər tʊk profɛʃənəl lɛsənz, ðoʊ aɪ sʌspɛkt it wʊd hæv hɛlpt kwaɪt ə bɪt. aɪ dɪd teɪk ə kɔrs waɪl æt kəmjunɪti kɑlɪdʒ ðæt simd tu ɪmpruv maɪ geɪm ðoʊ. aɪ wʊd laɪk tʊ gɛt bæk ɪntu gɔlfɪŋ əgɛn.

2. (a) The phones are [p,b,t,d,k,g]

   (b) The phones are [æ,a]

   (c) The phones are [ɹ]

   (d) The phones are [j,w,i,ɪ,e,æ,u,ʊ,o,a,ə,ʌ]

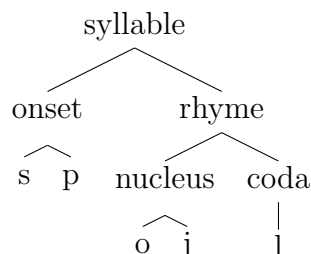3. (a) One minimal property is

   ```
   phone(X), sib(X), not(voi(X)).
   ```

   (b) One minimal property is

   ```
   phone(X), str(X), not(bck(X)), not(ctr(X)).
   ```

   (c) One minimal property is

   ```
   phone(X), lab(X), not(cnt(X)).
   ```

4. The structure is:

5. • We can solve this with some combinatorics.

   We want to find how many sequences of 6 phones or less there are. We can break this down to finding the number sequences of 0, 1, 2, 3, 4, 5, and 6 phones and summing them.

So we need to find the permutations with replacement, as it's possible to have more than one phone in a sequence.

Since there are 34 possible phones we want to find the following sum:

$$\sum_{i=0}^{6} 34^i = 34^0 + 34^1 + 34^2 + 34^3 + 34^4 + 34^5 + 34^6$$
$$= 1 + 34 + 1156 + 39304 + 1336336 + 45435424 + 1544804416$$
$$= 1591616671$$

So, there are 1591616671 possible sequences with 6 phones or less.

- Using the query:

```
findall(X, (syllable(X), length(X, Len), Len =< 6), _Y), length(_Y, YLen).
```

The result is:

```
YLen = 20608.
```

So there are 20608 English syllables according to `syllable.swipl`.

- 20608 is 0.00129% of the 1591616671 possible syllables.

- This percentage is important because it means the vast majority of sounds that can be made are not English syllables. It shows that English is a very small language in the scheme of things.

6.

```prolog
%% Start with some facts.

%% List all of the possible phones in Senufo
phone(a).
phone(e).
phone(i).
phone(k).
phone(o).
phone(p).
phone(t).
phone(u).

%% List all the consonants.
consonant(k).
consonant(p).
consonant(t).

%% List all the vowels.
vowel(a).
vowel(e).
vowel(i).
vowel(o).
vowel(u).

%% Now the rules

%% A 'syllable' is an 'onset' followed by a 'nucleus'.
syllable(Syl) :-
    onset(On),
    nucleus(Nu),
    append(On, Nu, Syl).

%% An 'onset' is just a 'phone' that is a 'consonant'.
onset([X]) :-
    phone(X),
    consonant(X).

%% A 'nucleus' is just a 'phone' that is a 'vowel'.
nucleus([X]) :-
    phone(X),
    vowel(X).
```

Some output from running this:

```prolog
?- findall(X, syllable(X), Y).
Y = [[k, a], [k, e], [k, i], [k, o], [k, u], [p, a], [p, e], [p|...], [...|...]|...].
```