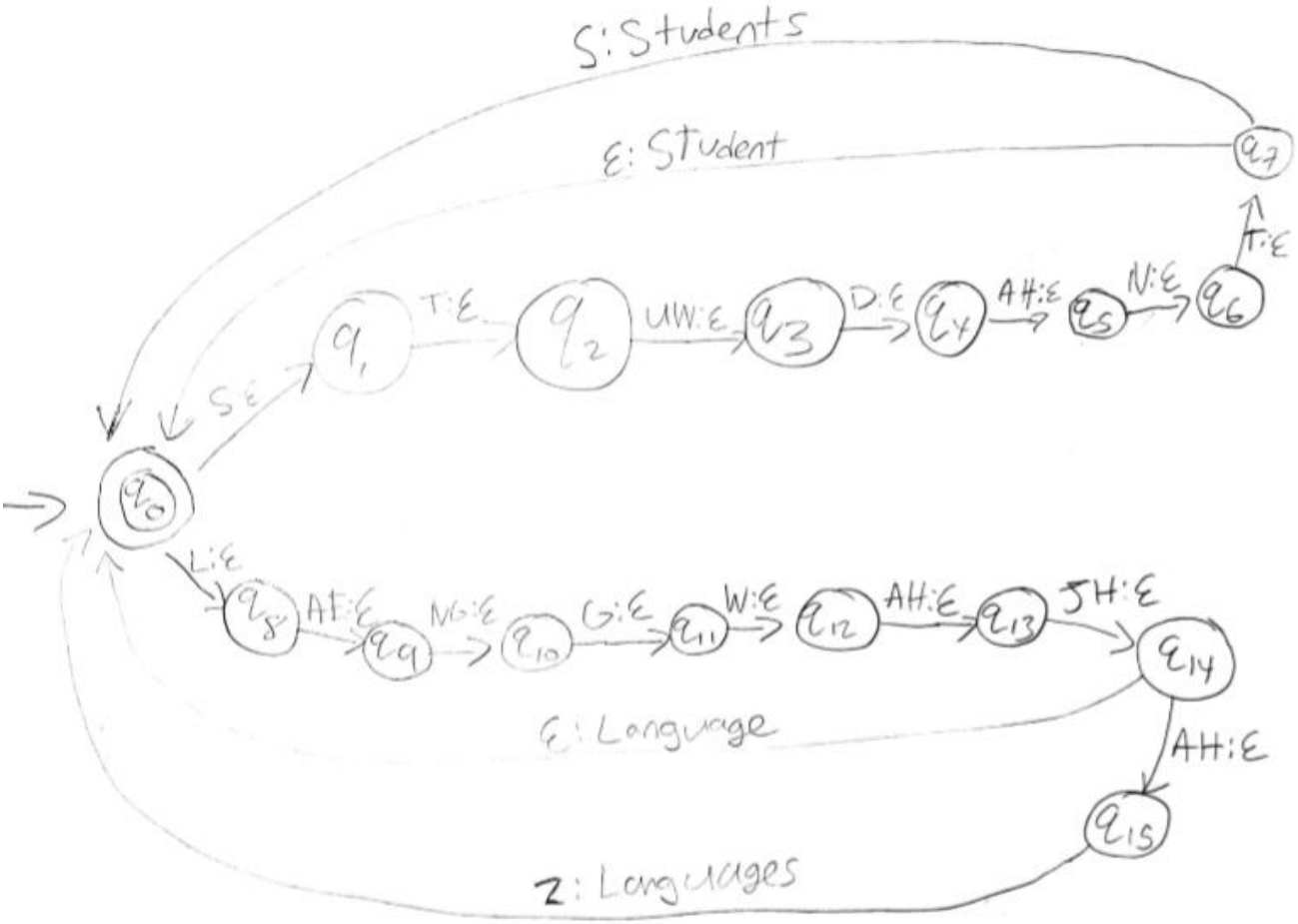
Muhammed Halbutogullari

913938544

LIN 177 HW 1

Part 1:



Part 2:

transition(1, s, 2, eps).

transition(2, t, 3, eps).

transition(3, uw, 4, eps).

transition(4, d, 5, eps).

transition(5, ah, 6, eps).

transition(6, n, 7, eps).

transition(7, t, 1, student).

transition(1, s, 8, eps).

transition(8, t, 9, eps).

transition(9, uw, 10, eps).

transition(10, d, 11, eps).

transition(11, ah, 12, eps).

transition(13, n, 14, eps).

transition(14, t, 15, eps).

transition(15, s, 1, students).

transition(1, l, 16, eps).

transition(16, ae, 17, eps).

transition(17, ng, 18, eps).

transition(18, g, 19, eps).

transition(19, w, 20, eps).

transition(20, ah, 21, eps).

transition(21, jh, 1, language).

transition(1, l, 22, eps).

transition(22, ae, 23, eps).

transition(23, ng, 24, eps).

transition(24, g, 25, eps).

transition(25, w, 26, eps).

transition(26, ah, 27, eps).

transition(27, jh, 28, eps).

transition(28, ah, 29, eps).

transition(29, z, 1, languages).

initial(1).

final(1).

Part 3:

The phrases that have the same pronunciation of ice cream are displayed as values of W. Here are all the possible values of W I got:

W = [ice, creme] ;

W = [ice, cream] ;

W = [aye, scream] ;

W = [eye, scream] ;

W = [ai, scream] ;

W = [i, scream].

I used fst(T, [computational, linguistics]) to get the following value for T:

T = [k, aa, m, p, y, uw, t, ey, sh, ah, n, ah, l, l, ih, ng, g, w, ih, s, t, ih, k, s]

I then ran:

fst([k, aa, m, p, y, uw, t, ey, sh, ah, n, ah, l, l, ih, ng, g, w, ih, s, t, ih, k, s], W).

And got a LOT of results:

W = [com, pew, tay, shun, uhl, linguistics] ;

W = [com, pew, tay, shun, uhl, lingg, wis, ticks] ;

W = [com, pew, tay, shun, uhl, lingg, wis, tics] ;

W = [com, pew, tay, shun, uhl, lingg, wyss, ticks] ;

W = [com, pew, tay, shun, uhl, lingg, wyss, tics] ;

W = [com, pew, tay, shun, uhl, lingg, wiss, ticks] ;

W = [com, pew, tay, shun, uhl, lingg, wiss, tics] ;

W = [com, pew, tay, shun, ul, linguistics] ;

W = [com, pew, tay, shun, ul, lingg, wis, ticks] ;

…

Each result represents a sequence of words with the same pronunciation as “computational linguistics”. Some of the words that prolog found I had no idea even existed. The results show the power and reach of the prolog program. The results kept going, but I stopped since I realized there must have been thousands of phrases that sound exactly like the term “computational linguistics”.