

COMP 304 Assignment 4

Daniel Braithwaite

September 23, 2016

1 Example Query's

1. **answer([i, feel, bad], R).** will unify R to be [what, makes, you, feel, bad, qm].
2. The previous query also works in reverse **answer(Q, [what, makes, you, feel, bad, qm])** will unify Q to be [i, feel, bad].

Similarly works the same for all other types of query's that the program can receive

2 Working Backwards

In my solution given a reply string it is able to find a question string. A few tweaks had to be made for this to work.

Firstly the match predicate is used to take a question string to a reply string. Originally it just matched the first words of the query string. The problem here was that when working backwards pro-log didnt know which definition of match to use, it would pick the first option. Here it would get stuck as part of creating the reply involves appending two lists together and, it would be unable to find the correct original list because the words dont line up. To solve this the match predicate will match on the first words of question and reply.

There is still an issue with working backwards, it will successfully find a solution but then if you ask for another it will enter an infinite loop and eventually run out of memory. It seems to create an infinitely big list while trying to reverse the append predicate, im not sure why this happens