Hidden Markov Models for POS tagging

Using the frequencies from tutorial 1 (files called `german-tag-freqs.txt`, `german-POS-bigram-freqs.txt` and `german-word-POS-freqs.txt`), compute the probabilities needed by a Markov-model-based tagger (initial probabilities, transition matrix, and emission probabilities).

We here want to estimate these probabilities from the relative frequencies of events observed in the corpus, i.e. based on the frequency counts that we have stored in the above mentioned three files.

a. Calculate the initial probabilities – for estimating the probabilities from the frequency counts of the POS tags, you can use R, or a spreadsheet.
b. Compute the relative frequencies of the part-of-speech bigrams by using `join` to combine the necessary files, importing the data into R or your spreadsheet and doing appropriate calculations.
c. Do the same with the word-tag pairs to compute the emission probabilities.
d. Give the probabilities used to tag the sentence *"Das behauptet das Publikum"*. Here you only need to retrieve the elevant probabilities from your files generated during a-c (use grep), you do not need to calculate the probabilities of the different POS tag assignments for the sentence.

(note: for the sort function to work correctly for sorting e.g. the second column, you need to both specify the start of sort and end of sort position, such as `sort -k 2,2 FILE`. Otherwise, the default sorting to the end of the line will be applied which may lead to undesired effects due to sorting position of the tab or space.)