|  |  |  |
| --- | --- | --- |
| Morris Code | From? | Message |
| **.......---.----.-.....------..-.-.** | Jerry McGuire |  |
| **...-...-..-....-....--.-.-.-.** | The Terminator | I’ll Be Back |
| **.-......-........-....-.-...--...----..----..-.-.-.....----**  **.-.---.-...--....** | Forrest Gump |  |
| **-----....---.-.-.---------....** | Hamlet | To be or not to be |
| **-.-.---..--.-..--.-.....--.-...-...-.....-.-...--....** | A Few Good Men |  |
| **...----.-.---..-.-...-..--......-.** | Star Wars | I am your father |

The Decoded Message

IF[cs] YOU[ppy] CAN[vm] READ[vvi] THIS[dd1]

THE[at] CODE[nn1] IS[vbz] WORKING[vvg]

WELL[rr] DONE[vdn]

AND[cc] PARTY[nn1] ON[ii]

The Explanation

The algorithm converts the Morris code into English by a list of helper methods. The algorithm starts by checking if there is any code to decipher. If there is any code to translate, the Morris code will be broken into individual charaters until there is no more Morris code to decipher. Each character will be checked for coherence with a character that was last checked. If the letter is the first in the string, then that letter will be the last word.

After that, there is a coherency check for the two letter that are currently in question. If the two letters in question fail the coherency check to see if the two letters will be likely to be together, the algorithm steps back and reevaluates what the two letters could have been. Once the all the letters have passed a coherency check to be turned into words, the algorithm goes one layer higher and will check the coherency of two words being together. Once all the words are check for coherency, the sentence as a whole will be evaluated for coherency. If the level or coherency (frequency of bigrams being next to each other) has is met, then the sentence will be entered into a list of possible translations that the word can be.

Once all the permeations for possible translations have been exhausted, the list of possible translations will be sorted in ascending order. Then the top twenty from the list of possible translations will be placed in an empty list of strings that will be presented to the user in order of overall frequency.

People that helped me

Bastion Auxer – Showed examples of how to conceptualize the project as a whole and helped in the basic understanding of recursive backtracking.

Joseph Juarez – Helped in debugging the values of the translations that were being skewed by factor of frequency.