```python
#!/usr/bin/env python3

import csv, sys

# Set up a dictionary for mapping the initialisms onto file handles
for output.
initialisms_of_interest = dict()

# Set up a set for storing the unique initialisms that we read in
from a file
initialisms = set()

if len(sys.argv) < 2:
        print('Usage: python3 search_engine.py
initialisms_file.txt')
        print('  where the initialisms file contains initialisms of
interest (all of the same length)')
        exit(1)

# Read the initialisms in
for line in open(sys.argv[1]):
        initialisms.add(line.rstrip())

# Open output files for storing matches, and keep the file handles
in a dictionary for ease of reference later
for initialism in initialisms:
        initialisms_of_interest[initialism] =
open('matches_'+initialism+'.txt','w')

# Read tab delimited file from stdin
reader = csv.reader(sys.stdin, delimiter="\t") # delimiter takes the
one character string \t representing tab to specify the field
separator

# Handle it line by line:
for row in reader:
        # The ngram = the first entry in the row, which we remove
trailing whitespace from and then split on whitespace
        ngram = row[0].rstrip().split()
        initialism = ''
        # Take each word and generate the initialism
        for word in ngram:
                initialism += word[0].upper() # Take the first
letter of each word and convert to a capital letter
        if initialism in initialisms_of_interest:
                # The match is an array, so convert it back to a
string and print it out to the right output file
                print(' '.join(str(word) for word in
ngram),file=initialisms_of_interest[initialism])

# close all the files
for file_handle in initialisms_of_interest.items():
        file_handle[1].close()
```