```
from copy import *

initials = {
  'a': ['albania','andorra','austria'],
  'b': ['belarus','belgium','bosnia and herzegovina','bulgaria'],
  'c': ['croatia','czech republic'],
  'd': ['denmark'],
  'e': ['estonia'],
  'f': ['finland','france'],
  'g': ['germany','greece'],
  'h': ['hungary'],
  'i': ['iceland','ireland','italy'],
  'j': [],
  'l': ['latvia','liechtenstein','lithuania','luxembourg'],
  'k': [],
  'm': ['macedonia','malta','moldova','monaco','montenegro'],
  'n': ['netherlands','norway'],
  'o': [],
  'p': ['poland','portugal'],
  'q': [],
  'r': ['romania','russia'],
  's': ['san marino','serbia','slovakia','slovenia','spain','sweden', \
                  'switzerland'],
  't': [],
  'u': ['ukraine','united kingdom'],
  'v': ['vatican city'],
  'w': [],
  'x': [],
  'y': [],
  'z': [],
}

def update_env(env, initial, nextn):
  env[initial].remove(nextn)
  return env

def path2(n, env, sol):
  c = n[len(n)-1]
  if len(env[c])==0: return sol
  else:
    return max([path2(nextn, update_env(deepcopy(env), c, nextn), sol+[nextn])
             for nextn in env[c]], key=len)

def path(name):
  return path2(name, update_env(deepcopy(initials), name[0], name), [name])
```