```
# reverse 2 points
def reverse(s):
  return s[-1] if len(s) == 1 else s[-1]+reverse(s[:-1])

# strip 3 points
def _elide(ch, chars):
  return '' if ch in chars else ch

def strip(s, chars):
  return _elide(s[0], chars) \
     if len(s) == 1 else _elide(s[0], chars)+strip(s[1:],chars)

# split 4points
def _embed(res, token):
  return res if len(token) == 0 else res+[token]

def _split(s, seps, res, token):
  return _embed(res, token) if len(s) == 0 else (\
     _split(s[1:], seps, _embed(res, token), '') \
     if s[0] in seps else _split(s[1:], seps, res, token+s[0]))

def split(s, seps):
  return _split(s, seps, [], '')

# find 5 points
def _find(s, ch, cnt):
  return -1 if len(s)==0 else (\
    cnt if (s[0] == ch) else _find(s[1:], ch, cnt+1))

def find(s, ch):
  if (find.state[2] != ch or find.state[1] != s or find.state[0] == -1):
    find.state = [0, s, ch]

  find.state[0] = _find(s[find.state[0]:], ch, find.state[0]+1)
  return -1 if find.state[0] == -1 else find.state[0]-1

find.state = [0, "", '']
```