HWK8. 卞宏宇 2023202269

15.2-4 顶点数：如果子问题不包含一个矩阵的情况

则有 $\frac{n(n+1)}{2}$ 个.

边：$\sum_{i=2}^{n}(C_i^2-1)(n-i+1)=\sum_{i=2}^{n}\frac{i(i-1)(n-i+1)}{2}$

将 $A_iA_{i+1}\cdots A_k$ 的子问题表示为顶点 D_{ik}.

则边从 D_{AB} 指向 D_{ab} $(1\le A\le a<b\le B$，且 A,a,b,B 不能同时相等)

15.3-3. 有的，唯一的区别只是求解公式变成了：

$$m[i,j]=\begin{cases}0 & \text{if } i=j.\\ \max\limits_{i\le k<j}\{m[i,k]+m[k+1,j]+p_{i-1}p_kp_j\} & \text{if } i<j\end{cases}$$

15.4-5 定义子问题为以 A[k] 结尾的最长子序列，算法如下：

```
FIND(A)
  n= A.length.
  B[n]
  B[0]=1
  mmax=0.
  for i=1 to n-1
      max=1
      for j=0 to i-1
          if A[j]<A[i] and B[j]+1>max
              max= B[j]+1
      B[i]=max
      if max>mmax
```

```
        mmax = max
    return mmax
```

```
以下a.EDIT(i,j,x,y)
    n = x.length
    m = y.length.
  if i == n
    return (m - j) * cost(insert)
  if j == m
    return min(cost(terminal), (n-i)*cost(delete))
  if x[i] == y[j]
    return EDIT(i+1, j+1, x, y) + cost(copy)
  else:
    e1 = EDIT(i+1, j+1, x, y) + cost(substitute)
    e2 = EDIT(i+1, j, x, y) + cost(delete)
    e3 = EDIT(i, j+1, x, y) + cost(insert)
    e4 = INTMAX
    if (x[i] == y[j+1] and x[i+1] == y[j])
      e4 = EDIT(i+2, j+2, x, y) + cost(spin)
    return min{e1, e2, e3, e4}.
```

b. cost(copy) = -1

cost(insert) = 2

cost(delete) = 2.

cost(substitute) = 1

cost(spin) = +∞

cost(terminal) = +∞

这样的分EDIT算法得到的是单词相似度的负值