Lab04-Dynamic Programming

CS214-Algorithm and Complexity, Xiaofeng Gao, Spring 2019.

$*$ If there is any problem, please contact TA Jiahao Fan.

$*$ Name: 田雪飞 Student ID: 515030910347 Email: 13487426939@qq.com

1. **Solution.** The following is solution:

(a) The recurrence for OPT(a):

$$OPT(a) = \begin{cases} 0 & a = 0 \\ min\{OPT(a - i^2) + 1\} & a \geq 1 \quad and \quad i\epsilon\mathbf{N} \quad and \quad i^2 \leq a \end{cases}$$

(b) Pseudo code by recurrence:

□

Algorithm 1: Dynamic algorithm

```
Input: A positive integer n
Output: The least number of perfect
        square numbers OPT(n)
$OPT(0) \leftarrow 0$
for $a \leftarrow 1$ to $n$ do
    min $\leftarrow$ $\mathbf{N}$;
    for $i \leftarrow 1$ to $a$ do
        if $OPT(a - i^2)$+1< $min$ then
            $min \leftarrow OPT(a - i^2) + 1$;
        $i \leftarrow i^2$;
    $OPT(a) \leftarrow min$;
return $OPT(n)$;
```

2. **Solution.** The following is solution:

(a) A recurrence for ANS(i,j):

$$ANS(i,j) = \begin{cases} \mathbf{true} & n = 0\&\&(m = 0 \, ||\forall j, p[j] =' *') \\ ANS(i-1, j-1)\&\&(s[i] == p[j]||p[j] ==' ?') & p[j]! =' *' \\ ANS(i-1, j-1)||ANS(i-1, j)||ANS(i-1, j) & p[j] =' *' \end{cases}$$

(b) Pseudo code by recurrence:

Algorithm 2: Dynamic Algorithm

```
Input: A string s and a pattern p.
Output: true if s matches p,or false otherwise.
$ANS(0,0) \leftarrow 0$;
for $j \leftarrow 0$ to $m$ do
    if $p[j] ==' *'$ then
        $ANS(0,j) = $ true;
    else
        return false;
for $i \leftarrow 0$ to n-1 do
    for $j \leftarrow 0$ to m-1 do
        if $p[j]! =' *'$ then
            $ANS(i,j) \leftarrow ANS(i-1,j)\&\&(s[i] == p[j]||p[j] ==' ?')$
        else
            $ANS(i,j) \leftarrow ANS(i-1,j-1)||ANS(i-1,j)||ANS(i,j-1)$
return $ANS(n,m)$;
```

(c) According to above,we use $ANS(i,j)$ represent $ANS[i][j]$;so $ANS(i,j)$ can be remembered. so main time cost on two loop,and space cost on $ANS[n][m]$. so,
Time complexity: is $o(nm)$;
Space complexity: is $o(nm)$. □

3. **Solution.** The following is solution.
(a) The code is in Code-Sequence Alignment.cpp file.
(b) According to the **(a)**,we can compute the edit distance between
the following two distance strings is 352.
(c) You can enter two strings to create a graphics by($txf\_homework\_04.py$).
and there has a example of picture in zip. □

Remark: You need to include your .cpp, .pdf and .tex files in your uploaded .rar or .zip file.