CSC 421/Applied Algorithms and Structures  
Problem set 4

Exercises

Name: Kavana Manvi Krishnamurthy

ID: 2158984

*Exercise Edit Distance*

**Using the edit distance algorithm given on page 114 in the textbook**, draw and fill out the *Edit* table used in it for the strings ***DARING*** and ***DISCARD***.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | **D** | **I** | **S** | **C** | **A** | **R** | **D** |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| **D** | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
| **A** | 2 | 1 | 1 | 2 | 3 | 3 | 4 | 5 |
| **R** | 3 | 2 | 2 | 2 | 3 | 4 | 3 | 4 |
| **I** | 4 | 3 | 2 | 3 | 3 | 4 | 4 | 4 |
| **N** | 5 | 4 | 3 | 3 | 4 | 4 | 5 | 5 |
| **G** | 6 | 5 | 4 | 4 | 4 | 5 | 5 | 6 |

*Exercise Corner to Corner Path*

You are given an 𝑛 × 𝑛 table 𝑝 with natural numbers in each entry representing a profit. As with the chessboard traversal problem discussed in class, the goal is to find a maximum profit path, subject to these conditions.

* The path must start in the upper left corner (that is, the square at position 1,1) and end at the lower right corner (position 𝑛, 𝑛).
* A move from a square to the next on a path must go either to the right or down.
* A path’s profit is the total of the profits for the squares followed by the path.

Answer each of the following:

1. Define a function 𝑞 as a recurrence relation where 𝑞(𝑖, 𝑗) is the maximum profit attainable for every path that ends at entry 𝑖, 𝑗.

Assuming index starts from 1 to n:

1. As we’ve seen, one computes values for 𝑞 by using an 𝑛 × 𝑛 table called 𝑞. For an  
   entry 𝑞[𝑖, 𝑗] in that table, specify which entries you need filled in before you can fill in 𝑞[𝑖, 𝑗].

We have to fill the entries 𝑞[𝑖, 𝑗-1] (top cell) and 𝑞[𝑖-1, 𝑗] (left cell) before we can proceed to fill the entry

𝑞[𝑖, 𝑗].

1. Specify an order for how to fill in the entries in 𝑞 given the previous answer.

Assuming index starts from 0 to n.

First step is to fill the first cell in q:

𝑞[0][0] = p[0][0]

Second, fill the first row and first coloumn of q:

𝑞[0, 𝑗] = *q*[0][*j*−1]+*p*[0][*j*] for all *j* from 1 to *n*−1.

𝑞[i,0] = *q*[i-1][*0*]+*p*[i][*0*] for all *i* from 1 to *n*−1.

Fill rest of the table:

*q*[*i*][*j*]=max(*q*[*i*−1][*j*],*q*[*i*][*j*−1])+*p*[*i*][*j*] for all *i* from 1 to *n*−1 and *j* from 1 to *n*−1.

1. Given the following values for the 𝑝 table, fill out the 𝑞 table. Write down the maximum achievable path profit.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **12** | **22** | **25** | **29** | **36** |
| **20** | **24** | **40** | **50** | **58** |
| **27** | **33** | **44** | **53** | **70** |
| **30** | **53** | **61** | **70** | **80** |
| **44** | **69** | **81** | **92**  C | **99** |