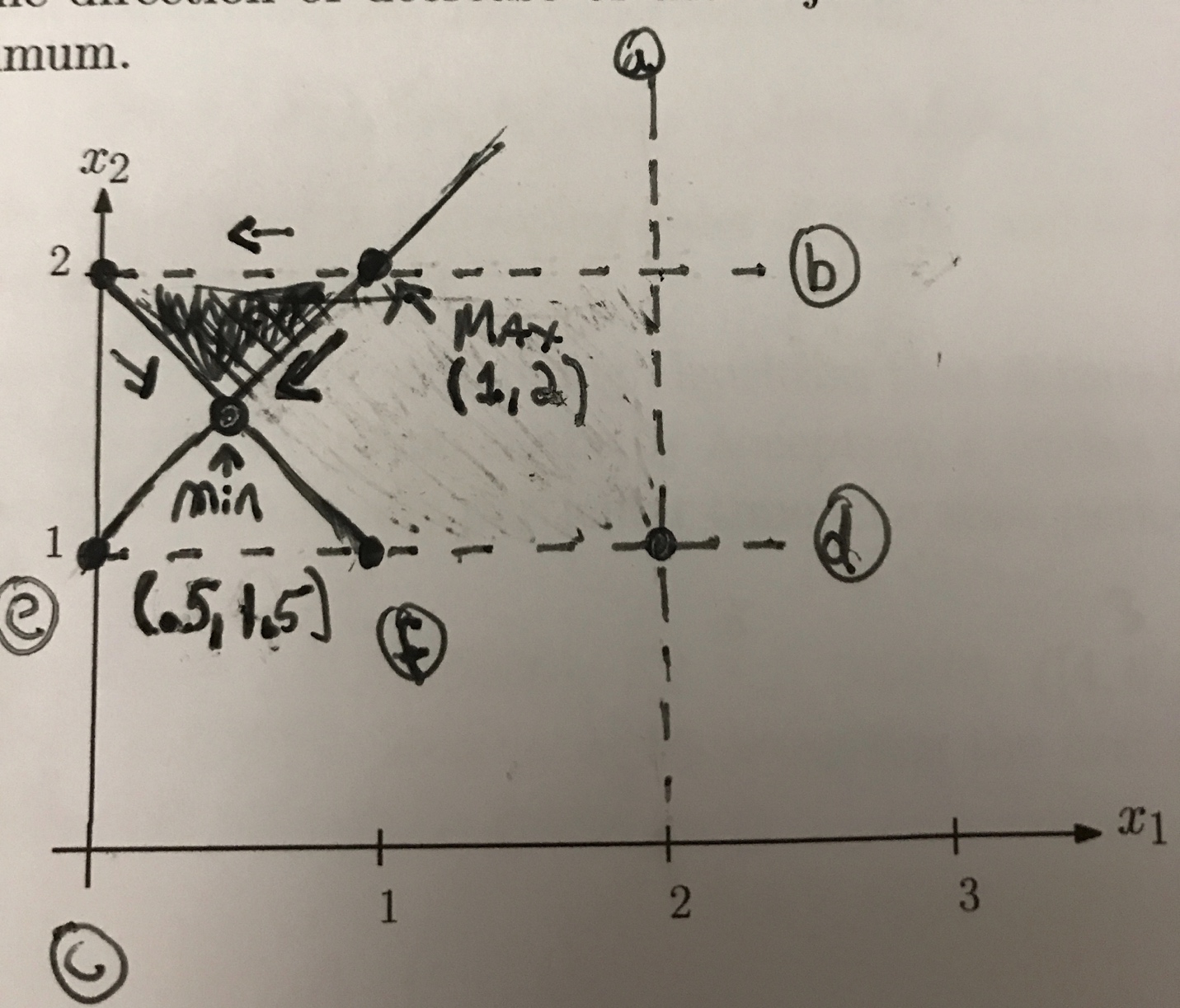
Homework 9

Jeff Wagner jpw373 and Jake Goldstein jsg525

Professor Hellerstein

Section B



Min = (.5, 1.5) and max = (1,2)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | E | D | I | T |
|  | 0 | 1 | 2 | 3 | 4 |
| D | 1 | 1 | 1 | 2 | 3 |
| I | 2 | 2 | 2 | 1 | 2 |
| S | 3 | 3 | 3 | 2 | 2 |
| T | 4 | 4 | 4 | 3 | 2 |
| A | 5 | 5 | 5 | 4 | 3 |
| N | 6 | 6 | 6 | 5 | 4 |
| C | 7 | 7 | 7 | 6 | 5 |
| E | 8 | 7 | 8 | 7 | 6 |

The edit distance between the two strings is 6

3.

For i = 0,1,2,…,m:

Iedit(i,0) = i

For j = 0,1,2,…,n:

Iedit(0,j) = j

For i = 1,2,...m:

For j = 1,2,...n:

Iedit(i,j) = min{ ledit(i-1, j)+1, ledit(i, j-1)+1, ledit(i-1, j-1) + idiff(i,j)

Return Iedit(m,n)

idiff(i, j) will return the difference between x[i] and y[j]

4.

1. The dimensions of the table are: 1 by n

I.e.,

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| max(0) | max(1) | max(2) |  |  | max(n) |

          0   1    2            ……………………………. n

Where max(0) = 0 //base case

Where the number 1,....,n are all the subscripts of b

    b)   Stored in the table is the max profit made by airing the commercial segments

    c)   max(i) = { max(i-1), max(k) + (xi)}

Where k = all the max values that make sure bi < ai+1

    d)   The idea behind the recurrence is you either use the current bids offer which is xi and that means that if you use it then you need to work backwards through the array and find a max kth value that makes sure that bi-1 < ai or you don’t use the current bids offering which means you can just use the max(i-1).

    e)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| max(1) | max(2) | max(3) |  |  | max(n) |

          0   1    2            ……………………………. n

The table/array is filled from left to right starting with b1 and ending at bn

Where max(0) = 0 //base case

Where the number 1,....,n are all the subscripts of b

   f) The value returned is the max profit you can get from running segments that don’t overlap and that is stored in the last index in the array which is max(n).

5.

1. S = time in hour spent sight-seeing

F = time in hours spent fishing

R = time in hours spent running

H = time in hours spent hacking

K = time in hours spent karaoking

    b)   Alexa’s constraint is: 1 <= F + H + S

Bob’s constraint is: 1 <= F +R + K

Carol’s constraint is: 1 <= F + H

Dudley’s constraint is: 1 <= H + S + K

   c)  The objective function is: 6(S) + 5(F) + 3(R) + 1(H) + 7(K) ≥ 0

We are minimizing it