**Algorithm 2021 Spring HW3**

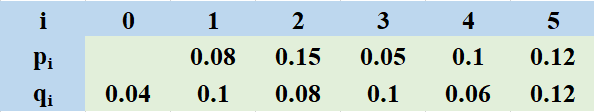
(Chapter 9 and Chapter 15)

1. (25pts) There are two sorted arrays, each containing numbers. Please give an algorithm to find the median of all elements, and it’s running time should be bounded by .
2. Compare Median of array 1 and 2.
3. If median 1 = median 2, the median of all 2n element is it.
4. If median 1 > median 2, find median of all 2n element in array1 [ n/2 , n ], array2 [ 0 , n/2 ]
5. Else, find median of all 2n element in array1 [ 0 , n/2 ], array2 [ n/2 , n]
6. Repeat the process above until the subarray remain 2 elements in each array. ( we will let them be array 1[0,1] and array 2[0,1] ).
7. Compare array1 [0] and array2 [0] to get the larger one.

Compare array1 [1] and array2 [1] to get the smaller one.

Add the above two elements and divided by 2, then we get the median.

1. (25pts) Determine the cost of an optimal binary search tree for a set of n = 5 keys with the following probabilities:



(A) 2.85

(B) 1.98

(C) 2.12

(D) 1.92

(note: level starts at 1.)

(note2: need to construct the tree)

Ans:

qi:直到搜尋leaf還是沒找到的點(參照資結課本第10章)

w[i,j] = qi-1 +

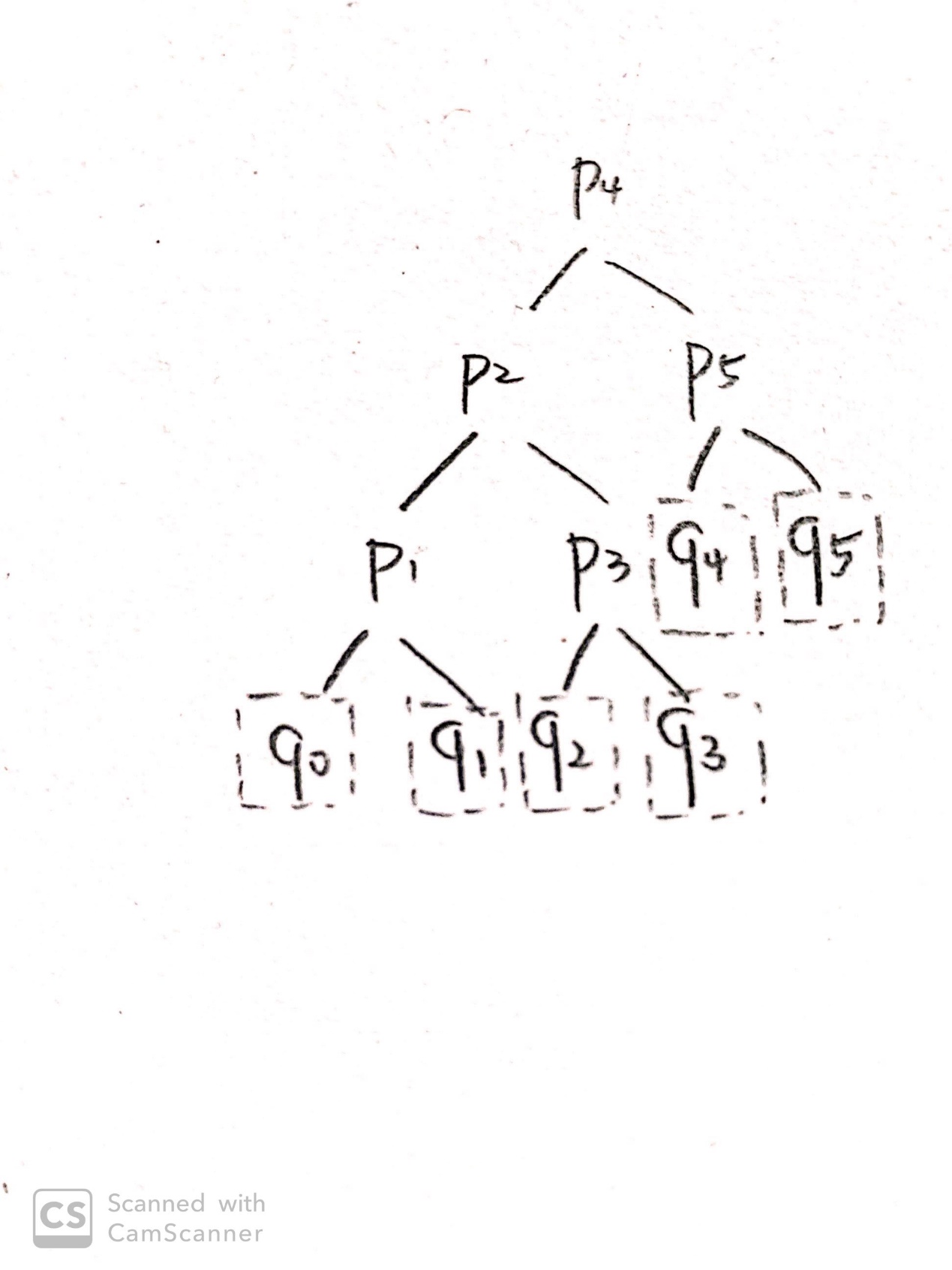
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| w (j→) | 0 | 1 | 2 | 3 | 4 | 5 |
| 1 | 0.04 | 0.22 | 0.45 | 0.6 | 0.76 | 1 |
| 2 |  | 0.1 | 0.33 | 0.48 | 0.64 | 0.88 |
| 3 |  |  | 0.08 | 0.23 | 0.39 | 0.63 |
| 4 |  |  |  | 0.1 | 0.26 | 0.5 |
| 5 |  |  |  |  | 0.06 | 0.3 |
| 6 |  |  |  |  |  | 0.12 |

e[i,j] = min(w[i,j] + e[i,k-1] + e[k+1,j])

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| e (j→) | 0 | 1 | 2 | 3 | 4 | 5 |
| 1 | 0.04 | 0.36 | 0.89 | 1.37 | 1.98 | 2.85 |
| 2 |  | 0.1 | 0.51 | 0.99 | 1.57 | 2.35 |
| 3 |  |  | 0.08 | 0.41 | 0.86 | 1.52 |
| 4 |  |  |  | 0.1 | 0.42 | 1.04 |
| 5 |  |  |  |  | 0.06 | 0.48 |
| 6 |  |  |  |  |  | 0.12 |

r

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| r (j→) | 1 | 2 | 3 | 4 | 5 |
| 1 | 1 | 2 | 2 | 2 | 4 |
| 2 |  | 2 | 2 | 3 | 4 |
| 3 |  |  | 3 | 4 | 4 |
| 4 |  |  |  | 4 | 5 |
| 5 |  |  |  |  | 5 |



1. (25pts)



Please find the longest common subsequence of string “ACBDSABBE” and “CADDSBEAB” by using the above algorithm. You need to draw the table to show your process.

Let X = “ACBDSABBE”, Y = “CADDSBEAB”

i is the index about with X, j is the index with Y.

below is table of c[i,j]

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | A | C | B | D | S | A | B | B | E |
|  | **0↑** | **0←** | 0← | 0← | 0← | 0← | 0← | 0← | 0← | 0← |
| C | 0↑ | 0↑ | **1↖** | **1**← | 1← | 1← | 1← | 1← | 1← | 1← |
| A | 0↑ | 1↖ | 1↑ | **1↑** | 1← | 1← | 2↖ | 2← | 2← | 2← |
| D | 0↑ | 1↑ | 1↑ | 1↑ | **2↖** | 2← | 2↑ | 2↑ | 2↑ | 2↑ |
| D | 0↑ | 1↑ | 1↑ | 1↑ | **2↑** | 2↑ | 2↑ | 2↑ | 2↑ | 2↑ |
| S | 0↑ | 1↑ | 1↑ | 1↑ | 2↑ | **3↖** | **3←** | **3←** | 3← | 3← |
| B | 0↑ | 1↑ | 1↑ | 2↖ | 2↑ | 3↑ | 3↑ | 4↖ | **4↖** | 4← |
| E | 0↑ | 1↑ | 1↑ | 2↑ | 2↑ | 3↑ | 3↑ | 4↑ | 4↑ | **5↖** |
| A | 0↑ | 1↑ | 1↑ | 2↑ | 2↑ | 3↑ | 4↖ | 4↑ | 4↑ | **5↑** |
| B | 0↑ | 1↑ | 1↑ | 2↑ | 2↑ | 3↑ | 4↑ | 5↖ | 5↖ | **5↑** |

The LCS = “CDSBE”

1. (25pts) Matrix-chain

Consider the multiplication of five matrices , and the dimensions are as follows:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| matrix | A1 | A2 | A3 | A4 | A5 |
| dimension |  |  |  |  |  |

Please find the minimum number of scalar multiplications, optimal parenthesization and fill up the two tables.

Table m:

I

1

2

3

4

5

J 1 2 3 4 5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 0 | 7500 | 8750 | 12750 | 20750 |
|  | 0 | 3750 | 8750 | 17250 |
|  |  | 0 | 3000 | 10500 |
|  |  |  | 0 | 6000 |
|  |  |  |  | 0 |

Table s:

I

1

2

3

4

J 2 3 4 5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1 | 1 | 3 | 3 |
|  |  | 2 | 3 | 3 |
|  |  |  | 3 | 3 |
|  |  |  |  | 4 |
|  |  |  |  |  |

Calculate m:

P = <20,25,15,10,20,30>

m[1,2] = P0 P1 P2 = 20\*25\*15 = 7500

m[2,3] = P1 P2 P3 = 25\*15\*10 = 3750

m[3,4] = P2 P3 P4 = 15\*10\*20 = 3000

m[4,5] = P3 P4 P5 = 10\*20\*30 = 6000

m[1,3] = m[1,1] + m[2,3] + P0 P1 P3 = 0 + 3750 + 20\*25\*10 = 8750

or = m[1,2] + m[3,3] + P0 P2 P3 = 7500 + 3000 + 20\*15\*10 = 13500

m[2,4] = m[2,2] + m[3,4] + P1 P2 P4 = 0 + 3000 + 25\*15\*20 = 10500

or = m[2,3] + m[4,4] + P1 P3 P4 = 3750 + 0 + 25\*10\*20 = 8750

m[3,5] = m[3,3] + m[4,5] + P2 P3 P5 = 0 + 6000 + 15\*10\*30 = 10500

or = m[3,4] + m[5,5] + P2 P4 P5 = 3000+ 0 + 15\*20\*30 = 12000

m[1,4] = m[1,1] + m[2,4] + P0 P1 P4 = 0 + 8750 + 20\*25\*20 = 18750

or = m[1,2] + m[3,4] + P0 P2 P4 = 7500 + 0 + 20\*15\*20 = 13500

or = m[1,3] + m[4,4] + P0 P3 P4 = 8750 + 0 + 20\*10\*20 = 12750

m[2,5] = m[2,2] + m[3,5] + P1 P2 P5 = 0 + 10500 + 25\*15\*30 = 21750

or = m[2,3] + m[4,5] + P1 P3 P5 = 3750 + 6000 + 25\*10\*30 = 17250

or = m[2,4] + m[5,5] + P1 P4 P5 = 8750 + 0 + 25\*20\*30 = 23750

m[1,5] = m[1,1] + m[2,5] + P0 P1 P5 = 0 + 17250 + 20\*25\*30 = 32250

or = m[1,2] + m[3,5] + P0 P2 P5 = 7500 + 10500 + 20\*15\*30 = 27000

or = m[1,3] + m[4,5] + P0 P3 P5 = 8750 + 6000 + 20\*10\*30 = 20750

or = m[1,4] + m[5,5] + P0 P4 P5 = 12750 + 0 + 20\*20\*30 = 24750