ECEN 475 Fall 2020 Lab2

Due 11:59pm, October 14, Tuesday

Stockmeyer algorithm is used for determining optimal orientation of the blocks. Please write a C/C++/MATLAB program to implement Stockmeyer's placement algorithm. The objective of this lab is to find the optimal orientation of the blocks in the floorplan for the minimal area.

Input: A *text file* describing an original slicing tree and the dimensions of the block. Polish expression is used for describing the connection of the internal nodes in the slicing tree.

For example, shown below is an original slicing tree and its floorplan. The polish expression for the example is 3-7-H-5-1-V-8-2-H-V-4-V-6-V-H. The block dimension in this example is (2,4), (1,3), (3,3), (3,5), (3,2), (5,3), (1,2), (2,4).
(2,4) means block 1's height is 4 and width is 2. (1,3) means that block 2's height is 3 and width is 1, etc.

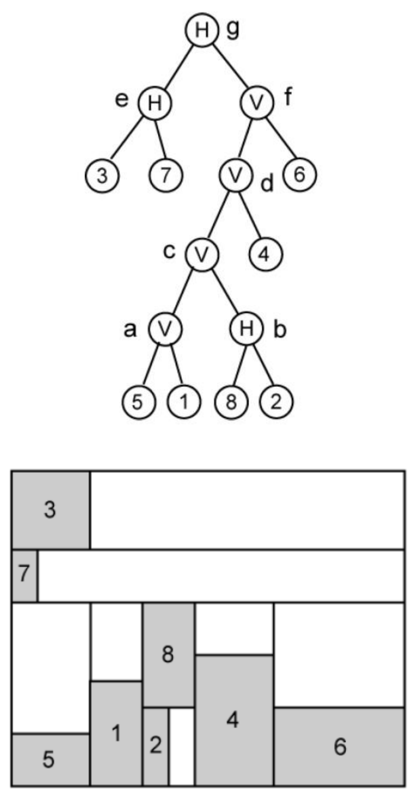

Figure 1. A slicing tree and its floorplan. (Note that the lower left corner of each block is placed at the lower left corner of its room.)

Input file format:

• The first line is the polish expression.
• Each of the following lines has two numbers which indicate the block dimension.
The input file for the slicing tree would look like:
3-7-H-5-1-V-8-2-H-V-4-V-6-V-H
2 4
1 3
3 3
3 5
3 2
5 3
1 2
2 4

Output:
• Height and width of the optimized floorplan.

Requirements:
• The input filename should be taken as the ***argument at command line*** in Linux/UNIX (ignore if using MATLAB).
• Please submit source code and the executable code for Linux/UNIX (ignore if using MATLAB).
• The output should be displayed to screen in a self-clear format.

Grading:
Your executable code will be evaluated by test cases different from the given example. The given test case could be used for testing but your code should be able to deal with general conditions. The grading will consider the solution correctness. If the answer is not totally correct, the notes in your code which indicate your understanding would help you get more points.

Appendix: The solution to the sample test case

Bottom-up Tree Traversal

visit node a: Since the cut orientation is vertical;

$$L = \{(2,3),(3,2)\}$$
$$R = \{(2,4),(4,2)\}$$

i join $l_1 = (2,3)$ and $r_1 = (2,4)$: we get $(2+2, \max\{3,4\}) = (4,4)$. Since the maximum is from R, we join l_1 and r_2 next.
ii join $l_1 = (2,3)$ and $r_2 = (4,2)$: we get $(2+4, \max\{3,2\}) = (6,3)$. Since the maximum is from L, we join l_2 and r_2 next.
iii join $l_2 = (3,2)$ and $r_2 = (4,2)$: we get $(3+4, \max\{2,2\}) = (7,2)$.

Thus, the resulting dimensions are $\{(4,4),(6,3),(7,2)\}$.

visit node b: Since the cut orientation is horizontal;

$$L = \{(4,2),(2,4)\}$$
$$R = \{(3,1),(1,3)\}$$

i join $l_1 = (4,2)$ and $r_1 = (3,1)$: we get $(\max\{4,3\}, 2+1) = (4,3)$. Since the maximum is from L, we join l_2 and r_1 next.

ii join $l_2 = (2,4)$ and $r_1 = (3,1)$: we get $(\max\{2,3\}, 4+1) = (3,5)$. Since the maximum is from R, we join l_2 and r_2 next.

iii join $l_2 = (2,4)$ and $r_2 = (1,3)$: we get $(\max\{2,1\}, 4+3) = (2,7)$.

Thus, the resulting dimensions are $\{(4,3),(3,5),(2,7)\}$.

visit node g: Since the cut orientation is horizontal;

$$L = \{(3,4)\}$$
$$R = \{(20,3),(18,4),(13,5),(12,7)\}$$

i join $l_1 = (3,4)$ and $r_1 = (20,3)$: we get $(\max\{3,20\}, 4+3) = (20,7)$. Since the maximum is from R, we join l_1 and r_2 next.

ii join $l_1 = (3,4)$ and $r_2 = (18,4)$: we get $(\max\{3,18\}, 4+4) = (18,8)$. Since the maximum is from R, we join l_1 and r_3 next.

iii join $l_1 = (3,4)$ and $r_3 = (13,5)$: we get $(\max\{3,13\}, 4+5) = (13,9)$. Since the maximum is from R, we join l_1 and r_4 next.

iv join $l_1 = (3,4)$ and $r_4 = (12,7)$: we get $(\max\{3,12\}, 4+7) = (12,11)$.

Thus, the resulting dimensions are $\{(20,7),(18,8),(13,9),(12,11)\}$. The minimum area floorplan is $13 \times 9 = 117$.

Area reduced from 15 × 12 to 13 × 9