ifis
Institut für Informationssysteme
Technische Universität Braunschweig

Technische Universität Braunschweig
Institut für Informationssysteme
http://www.ifis.cs.tu-bs.de

Karl Neumann, Sarah Tauscher

Spatial Databases and GIS

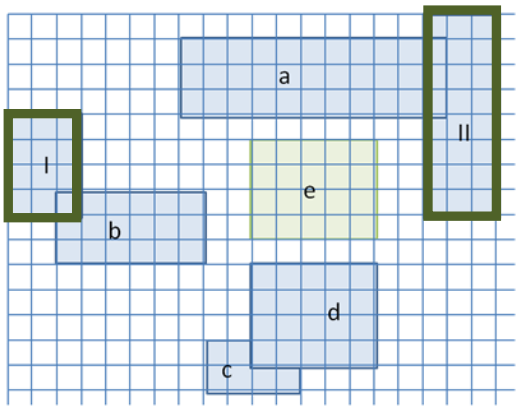
Solutions for Sheet 7

Exercise 1 (R-tree)

The blue rectangles below had been in one r-tree node. After the insertion of the green rectangle a split is necessary. Use the algorithm with linear complexity to determine the seeds and distribute the rectangles into two nodes.

Determine the normalized distance between the smallest right/upper boundary and the biggest left/lower boundary of all rectangles:

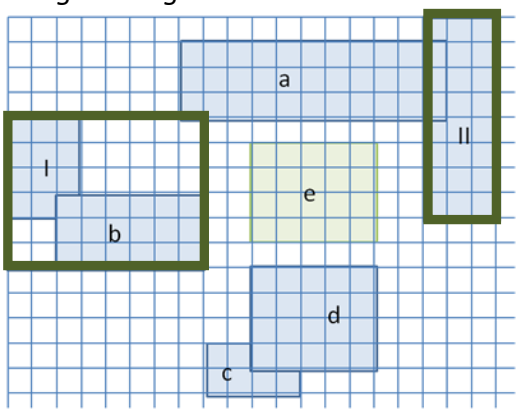
x-direction: 7/10; y-direction: 3/5; choose (I, II) as seed nodes



Calculate the increase in volume if you add the remaining rectangles either to I or II.

	a	b	c	d	e
I	114	36	120	138	63
II	80	156	156	116	66
Diff.	**34**	**120**	**36**	**22**	**3**

Add the rectangle first, where the difference between the two choices is the biggest(b). Recalculate the values and repeat these two steps until all rectangles are added to one of the nodes.



ifis
Institut für Informationssysteme
Technische Universität Braunschweig

Technische Universität Braunschweig
Institut für Informationssysteme
http://www.ifis.cs.tu-bs.de

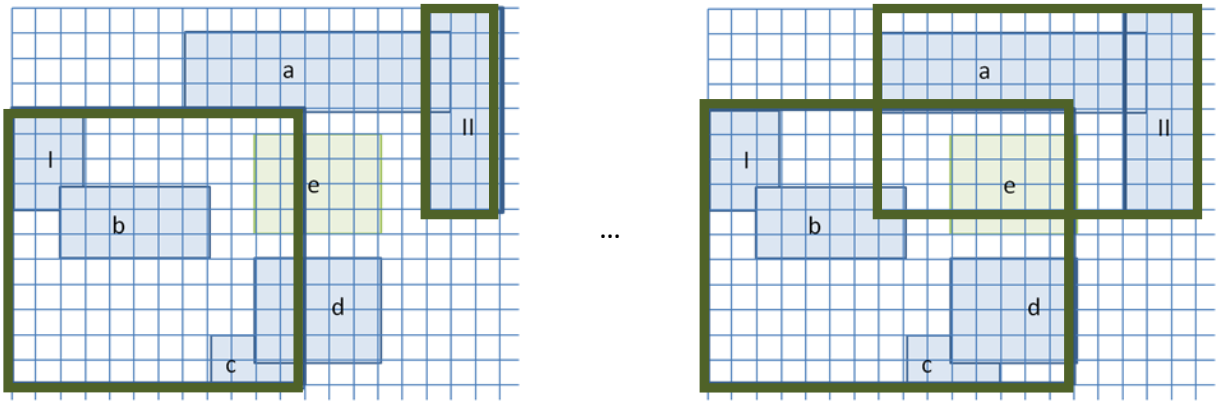
Karl Neumann, Sarah Tauscher

	a	c	d	e
I+b	114	84	102	27
II	80	156	116	66
Diff.	34	72	14	39

	a	d	e
I+b+c	120	33	33
II	80	116	66
Diff.	40	83	33

	a	e
I+b+c+d	87	0
II	80	66
Diff.	7	66

	a
I+b+c+d+e	87
II	80
Diff.	7



Exercise 2 (BSP-tree)

1. Construct the BSP tree using the heuristic (slide 498) for the example on slide 502. In contrast to the solution presented during the lecture, objects that lie on the same splitline should be stored in one node.
   Calculate the value for all objects (see table below). As 2, 3, 5 and 6 all have the smallest value, you may choose one of them. As object 6 could be splitted by two other objects it is reasonable to choose 6, although this decision has nothing to do with the heuristic.

object	# objects in front of	# objects behind	# splits	result
0	3	1	2	32
1	2	2	1	15
2	0	6	0	6
3	6	0	0	6
4	2	2	1	15
5	6	0	0	6
6	0	6	0	6

Technische Universität Braunschweig
Institut für Informationssysteme
http://www.ifis.cs.tu-bs.de

Karl Neumann, Sarah Tauscher

As all objects lie on the same side of 6 just recalculate the values for the other objects.

object	# objects in front of	# objects behind	# splits	result
0	2	1	2	31
1	2	2	0	0
2	0	5	0	5
3	5	0	0	5
4	2	2	0	0
5	5	0	0	5

Objects 1 and 4 have the same value but they also lie on the same splitline, so both are added next as one node. Now you have to group the remaining objects according to their position with respect to the splitline defined by 1 and 4.

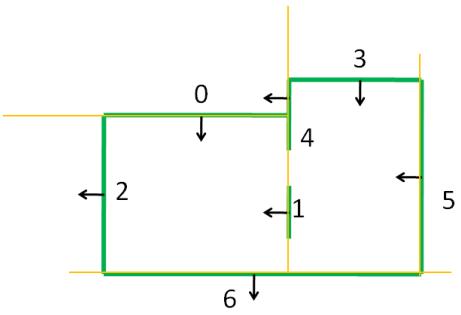
Objects in front of 1 and 4

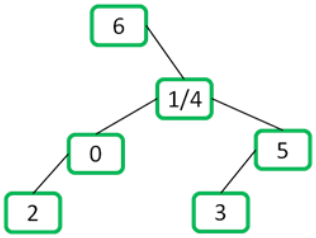
object	# objects in front of	# objects behind	# splits	result
0	1	0	0	1
2	0	1	0	1

Objects behind 1 and 4

object	# objects in front of	# objects behind	# splits	result
3	1	0	0	1
5	1	0	0	1

→ as all objects get the same values you may choose any order, e.g.:





Technische Universität Braunschweig
Institut für Informationssysteme
http://www.ifis.cs.tu-bs.de

Karl Neumann, Sarah Tauscher

2. What influence does the orientation of the split lines have, i.e. how would the tree change if you change the orientation of some of them?
   You have to switch the subtrees of that node, i.e. if you change the orientation of line1/4 :