chooses the first opened bin and does not try to optimize the choice. This might give rise to bins filled only partly and, therefore, to an increased number of bins (the division of the available space into many small fragments is called *fragmentation*).

Figure 2.4
An example of (a) an optimal packing that uses $3n/2$ bins and (b) a packing produced by FFD that uses $11n/6$ bins

An apparently better algorithm for MINIMUM BIN PACKING is the *Best Fit Decreasing* algorithm. Like *First Fit Decreasing*, *Best Fit Decreasing* initially sorts the items in non-increasing order with their respect their weight and then considers them sequentially. The difference between the two algorithms is the rule used for choosing the bin in which the new item a_i is inserted: while trying to pack a_i, *Best Fit Decreasing* chooses one bin whose empty space is minimum. In this way, the algorithm tries to reduce the fragmentation by maximizing the number of bins with large available capacity.

It is possible to see that the quality of the solution found by *Best Fit Decreasing* may be worse than the quality of the solution found by *First Fit Decreasing* (see Exercise 2.9). On the other hand, as shown in the following example, it is possible to define instances for which *Best Fit Decreasing* finds an optimal solution while *First Fit Decreasing* returns a non-optimal solution.

◀ Example 2.6

For any $n > 0$, let us us consider the following instance x_n. The instance includes $6n$ items: n items of size $7/10$, $2n$ items of size $2/5$, n items of size $1/5$, and $2n$ items of size $3/20$. As can be easily seen, $m_{BFD}(x_n) = 2n$, while $m_{FFD}(x_n) = 2n + \lceil n/10 \rceil$.