

7. Policies

goals: minimize fragmentation, minimize search time

ordering: address-based vs. increasing size vs. decreasing size

Best Fit, Worst Fit, First Fit, Next Fit



Assume an allocation request of size 15. A best-fit approach would search the entire list and find that 20 was the best fit, as it is the smallest free space that can accommodate the request. The resulting free list:



As happens in this example, and often happens with a best-fit approach, a small free chunk is now left over. A worst-fit approach is similar but instead finds the largest chunk, in this example 30. The resulting list:



The first-fit strategy, in this example, does the same thing as worst-fit, also finding the first free block that can satisfy the request. The difference is in the search cost; both best-fit and worst-fit look through the entire list; first-fit only examines free chunks until it finds one that fits, thus reducing search cost.

Other: tree vs. list, separate lists for different sizes, custom block allocation, ...