

**TABLE 1—COMMON ALLOCATION ALGORITHMS**

<b>Allocation algorithm</b>	<b>Method</b>	<b>Upside</b>	<b>Downside</b>
<b>First-fit</b>	Searches for the first block that offers the requested size.	Easy to implement Works well at the beginning	Long and unpredictable High degree of fragmentation
<b>Best-fit</b> (almost never used)	Searches for the block that is closest to the requested size	Uniform time for allocating blocks regardless of size	Slow High fragmentation
<b>Worst-fit</b> (rarely used)	Searches for the block that is the worst match for the requested size	Faster than best-fit Less fragmentation than best-fit	Not enough benefits to justify the cost
<b>Buddy</b>	Blocks are split from and joined to "buddy blocks" based on data structure	Info on data Limits fragmentation to a certain degree	Can be difficult to write Properties may vary
<b>Fixed size</b>	Usually searches multiple lists of identically sized blocks	Easy to implement Counteracts fragmentation when there are few block sizes Consistently fast	Large amounts of internal fragmentation