Shell sort uses a series of increments and then sorts the list by repeatetly comparing $i$ and $i - inc$. Its comparisons push larger elements to the top, and its $O(n \cdot q)$ ($q$ average number of swaps) time comparison cycles. with $O(\log n)$ cycles gives $O(qn \log n)$ time, uses $O(1)$ auxilary memory, but is unstable. Now, how about $q$? It'll land in the middle of the elements on average (due to each cycle being a set of repeated insertion sorts). So, it's roughly $O(n/s)$, $s$ being the cycle size. This means it's actually $O(n^2)$. Its grade is **C** due to this time complexity.