Problem 6.1 [Cache-oblivious Maximal Points in 3D].

Describe a cache-oblivious algorithm which takes $N$ distinct points in 3D space and returns a list of all maximal points. A point $(x, y, z)$ is maximal if there is no other point $(x', y', z')$ such that $x' \geq x$, $y' \geq y$, and $z' \geq z$; in other words, $(x, y, z)$ is not dominated by any other point. Your algorithm should run in $O(\text{sort}(N, M, B)) = O(\frac{N}{B} \log_{M/B} \frac{N}{B})$ memory transfers, under the tall-cache assumption.