Algorithm 3: Divide-and-Conquer

---

**Divide-and-Conquer Convex Hull**

1. If $|P| \leq 3$, then compute the convex hull by brute force in $O(1)$ time and return.
2. Otherwise, partition the point set $P$ into two sets $A$ and $B$, where $A$ consists of half the points with the lowest $x$-coordinates and $B$ consists of half of the points with the highest $x$-coordinates.
3. Recursively compute $H_A = CH(A)$ and $H_B = CH(B)$.
4. Merge the two hulls into a common convex hull, $H$, by computing the upper and lower tangents for $H_A$ and $H_B$ and discarding all the points lying between these two tangents.

---

**Finding the Lower Tangent**

**LowerTangent($H_A$, $H_B$):**

1. Let $a$ be the rightmost point of $H_A$.
2. Let $b$ be the leftmost point of $H_B$.
3. While $ab$ is not a lower tangent for $H_A$ and $H_B$ do
   a. While $ab$ is not a lower tangent to $H_A$ do $a = a - 1$ (move $a$ clockwise).
   b. While $ab$ is not a lower tangent to $H_B$ do $b = b + 1$ (move $b$ counterclockwise).
4. Return $ab$.

---

More questions about Convex Hull:

1. **Polygon Containment**: one common question we might ask is for a given point $q$ and convex polygon $P$, is $q \in P$. Assume we already have the convex hull of $P$ computed and properly organized. The organization required is to have the points on the convex hull divided into two subsets: the upper convex hull and the lower convex hull. Furthermore, the points are sorted by $x$-coordinate.

   Once we have the upper and lower convex hull subsets organized, we are ready to determine if we have $q \in P$. If so, then a vertical line drawn straight up from $q$’s should cross exactly one edge on the upper convex hull. Similarly, a vertical line drawn down from $q$ should cross exactly one edge on the lower convex hull. We can perform binary search on the vertices in the upper hull and on the vertices in the lower hull to find if such edges exist. This takes only $O(logh)$ ($h$ is the convex points of $P$) time if we already have the upper and lower hulls organized.
2. **Adding a Point to a Polygon:** For a point q that is outside of the convex polygon P, we want to add that point q to the convex hull of P. To do this, think of the point q as a light source shining onto the polygon P. The edges on the polygon that are lighted by q fall between two vertices that are called **supporting tangent vertices** v₁ and v₂ - vertices that the two **tangent vectors** t₁ and t₂ intersect. We can also apply binary search with $O(\log h)$ time complexity (h is the convex points of P) if we already have the upper and lower hulls organized.

Combine the two questions together, we have a new algorithm to compute convex hull - Randomized Incremental Construction, which can be generalized to higher dimensions.

The basic idea is that we start from a randomly chosen three points. They form a triangle that is a convex hull. We then randomly add a point from the left of the given set of points into the current convex hull. We first determine whether this point locates inside the current convex hull or not (question 1). If yes, we add another point. Otherwise, we add the point into the current convex hull (question 2).