1 Convex Hull

1.1 The Problem

The convex hull of a set $Q$ of points denoted $CH(Q)$ is the smallest convex polygon $P$ for which each point in $Q$ is either on the boundary of $P$ or in its interior. See Figure 1 for an example.

![Figure 1: A set of points and its convex hull.](image)

1.2 Graham’s Scan

Sketch of the algorithm:

- Maintain a stack $S$ of candidate points
- Each point in $Q$ is pushed once onto the stack, and wrong points are eventually popped from the stack.
- At the end, stack $S$ contains exactly the vertices of $CH(Q)$ in counterclockwise order.

Pseudocode:

```
Graham-Scan(Q):
1 Let p_0 be the point in Q with the minimum y-coordinate or the leftmost point in case of a tie
2 Let \{p_1, p_2, \ldots, p_m\} be the remaining points in Q, sorted by polar angle in CCW order around p_0 (if more than one point has the same angle, remove all but the one that is farthest from p_0)
3 PUSH(p_0, S)
4 PUSH(p_1, S)
5 PUSH(p_2, S)
6 for i <- 3 to m
7     do while the angle formed by points NEXT-TO-TOP(S), TOP(S), and p_i makes a non-left turn (straight or to the right)
8         do POP(S)
9     PUSH(p_i, S)
10 return S
```
Example of Graham’s Scan:

At the beginning of the for loop, $S$ contains $p_0$, $p_1$, and $p_2$.

When considering $p_3$, we first pop $p_2$ (because $\angle p_1 p_2 p_3$ makes a non-left turn) and then push $p_3$ onto the stack.

1.3 Analysis of Graham’s Scan

1.3.1 Correctness

**Theorem 1.** If Graham-Scan is run on a set $Q$ of points, where $|Q| \geq 3$, then at termination, the stack $S$ consists of, from bottom to top, exactly the vertices of $CH(Q)$ in counterclockwise order.

**Proof.** We shall use the following loop invariant to prove this theorem:

The stack $S$ consists of, from bottom to top, exactly the vertices of $CH(Q_{i-1})$ in counterclockwise order, where $Q_i$ is the set of points $\{p_0, p_1, \ldots, p_i\}$.

- **Initialization:** At the beginning of the loop, $i = 3$, and stack $S$ consists of $p_0$, $p_1$, and $p_2$, which is exactly the contents of $Q_{i-1}$, or $Q_2$. The convex hull $CH(Q_2)$ is clearly the set of points in $S$ because the convex hull of three points is going to be the triangle formed by the three points.
• **Maintenance:** Assuming that the loop invariant hold for all previous iterations of the loop, we show that on the $i$-th iteration of the loop, if the loop invariant is true at the beginning of the loop, it is therefore true at the beginning of the next one as well.

Let $p_j$ and $p_k$ be the two points at the top of the stack right before $p_i$ is pushed onto the stack.

**Claim 1.** After $p_i$ is pushed onto the stack, $S$ consists of $CH(Q_j \cup \{p_i\})$.

*Proof.* By the loop invariant, we know that $S$ consists of $CH(Q_j)$ right before we push $p_i$ onto the stack. This is because the stack is in the same state as directly after pushing $p_j$ onto the stack. This is because no point is ever pushed more than once onto the stack, and thus no points beneath $p_j$ could have been removed without popping $p_j$ off the stack.

Secondly, because we exited the while loop, we know that $\angle p_k p_j p_i$ is a left turn. Since $p_0$ is the bottom and leftmost point, we know that $\angle p_j p_i p_0$ is also a left turn. Thus, once we push $p_i$ onto the stack, we get the convex hull of $Q_j$ and the point $p_i$.

We still need to show that this convex hull is also the convex hull of $Q_i$, i.e. including the points between $p_j$ and $p_i$ that were popped off in the while loop.

**Claim 2.** $CH(Q_j \cup \{p_i\}) = CH(Q_i)$.

*Proof.* Let $p_t$ be a point that was popped off and $p_r$ be the point directly below it in the stack at that time. To have popped $p_t$, then $\angle p_r p_t p_i$ is a non-left turn. If it was a right turn, that would make the polygon not convex since we are going counterclockwise around the set of points, and $p_t$ is not going to be on the perimeter of the convex hull. If it was not a turn at all (the angle is $180^\circ$), then $p_t$ is on the perimeter of the convex hull but is not going to be a vertex of the polygon. In either case, we can remove $p_t$ from consideration and still have the same convex hull, i.e. $CH(Q_i - p_t) = CH(Q_i)$. This is true for all members of the set $P_i$ of all points popped off during the $i$-th iteration of the loop, so we have $CH(Q_i - P_i) = CH(Q_i)$. Because we are using a stack in which all the points popped off had to be contiguous in the stack and we stopped popping when the top point in the stack was $p_j$, we have $CH(Q_i - P_i) = CH(Q_j \cup \{p_i\}) = CH(Q_i)$.

• **Termination:** When the loop terminates, $i = m + 1$, so $S$ consists of $CH(Q_m)$ by the loop invariant, which is equivalent to $CH(Q)$. This is because any points that were removed going from $Q$ to $Q_m$ in line 2 were not going to be in $CH(Q)$. (You should be able to see why this is pretty easily; if not, draw a picture.)

1.3.2 **Runtime**

- Line 1: $O(n)$
- Line 2: $O(n \lg n)$ for sorting
- Lines 3-5: $O(1)$
- Lines 6-9, calls to Push: $O(n)$ (each point only gets pushed once)
- Lines 6-9, calls to Pop: $O(n)$ (each point gets popped at most one time)
- Total running time: $O(n \lg n)$
1.4 Lower Bound

We get a lower bound of \(\Omega(n \lg n)\) time by using the solution to a convex hull problem to sort numbers.

1. Given a list of numbers \((x_1, x_2, \ldots, x_n)\), construct the set with the points: \(((x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n))\) where \(y_i = x_i^2\). This takes \(O(n)\) time. The parabola is a convex function, which means that if all of our points are on the parabola (which they will be), they must all be part of the convex hull.

2. Find a convex hull on those points. This takes ??? time.

3. Walk along points on the hull and get them in sorted order in \(O(n)\) time.

4. We must have taken \(\Omega(n \lg n)\) to sort in the comparison model, so since everything else only took \(O(n)\) time, finding the convex hull had to take at least \(\Omega(n \lg n)\).