Presentation writer: Tigran Hayrapetyan

Lecturer | Programmer | Researcher

www.linkedin.com/in/tigran-hayrapetyan-cs/

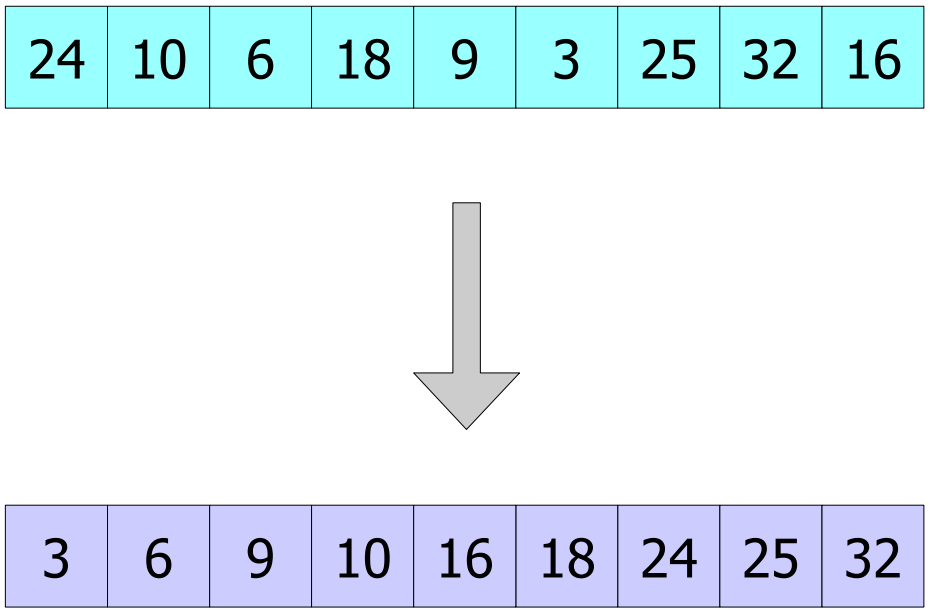
Convex hull

(divide and conquer algorithm)

Prerequisites:

- *recursion.*

Divide and conquer

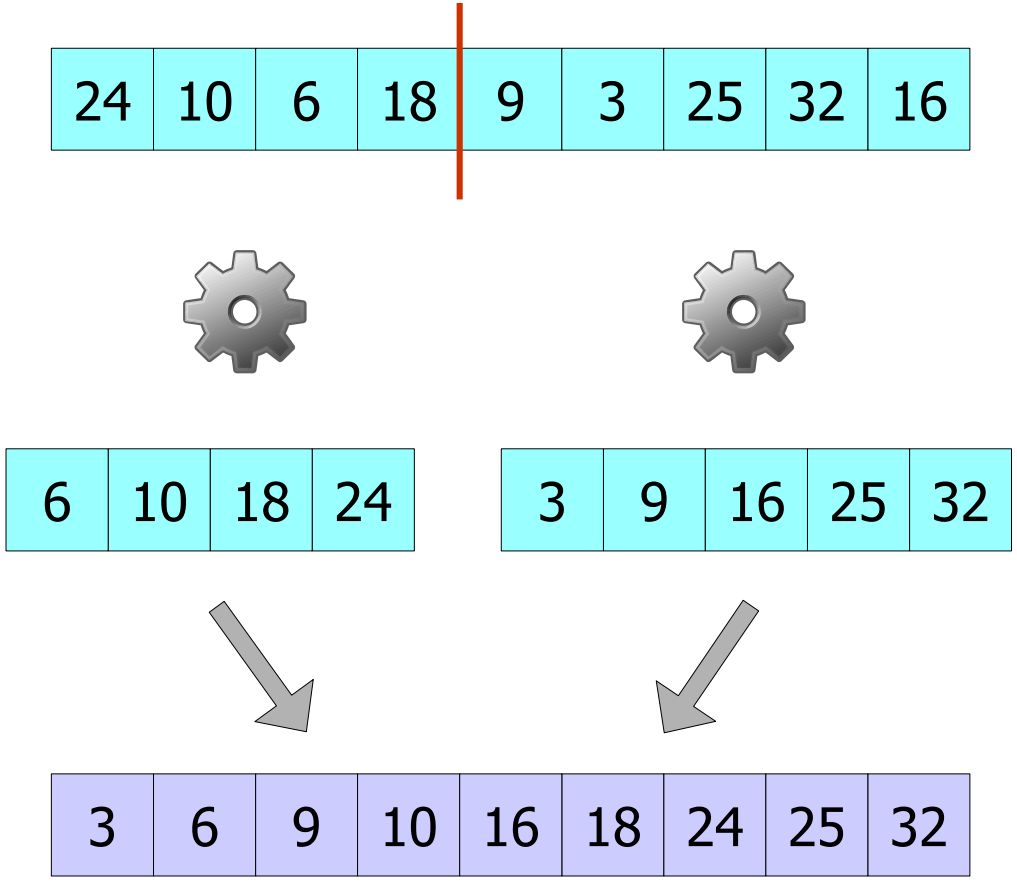


Many techniques which work for sorting, also work for finding convex hull.

Merge sort uses divide and conquer technique.

... let's apply it for finding the convex hull too.

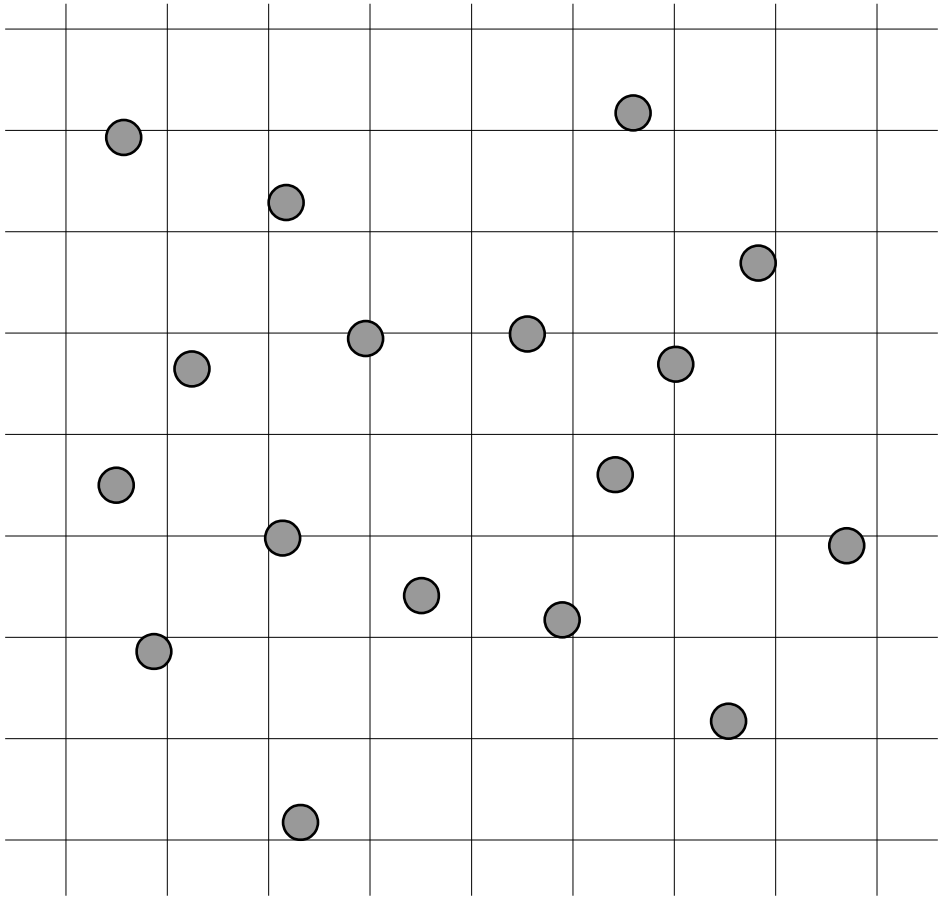
Divide and conquer



Let's recall how Merge sort works:

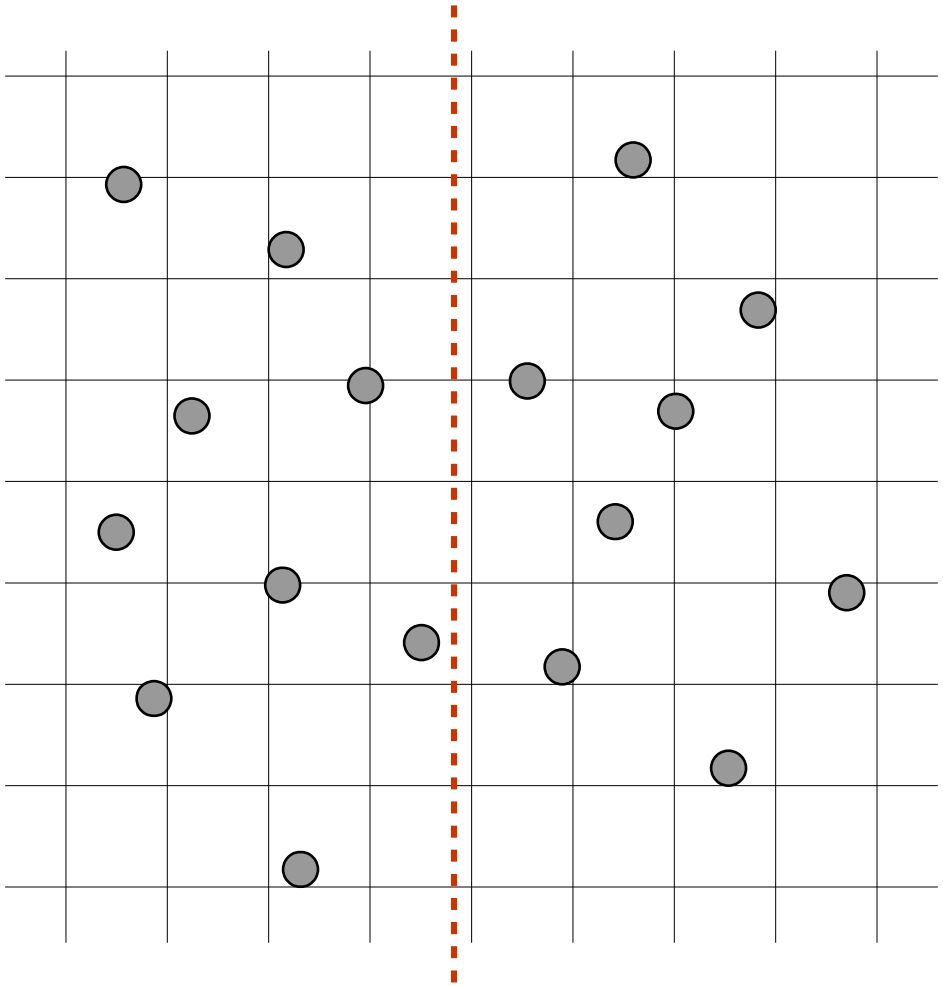
- Divide the array in **2** equal halves,
- Sort them independantly (recursively),
- Merge them.

Divide and conquer



Question: How can we apply the same technique to convex hull finding?

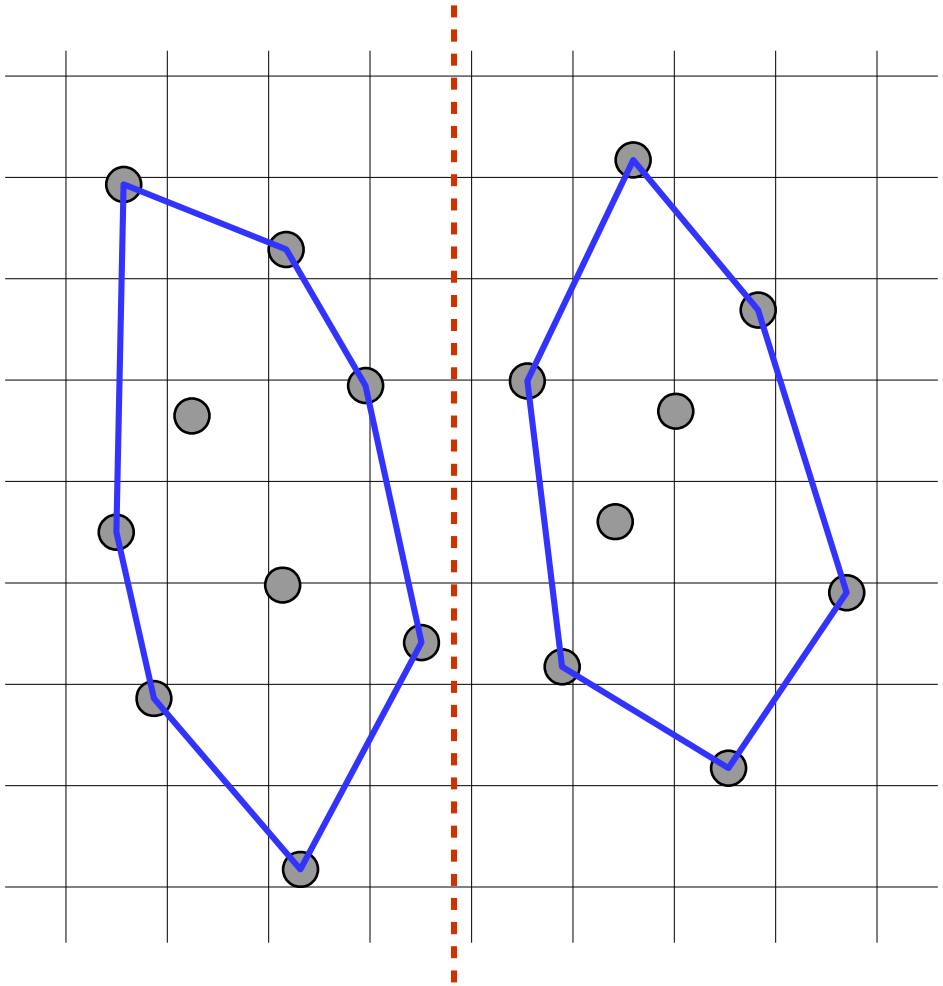
Divide and conquer



Answer:

- Divide the points into 2 equal parts,

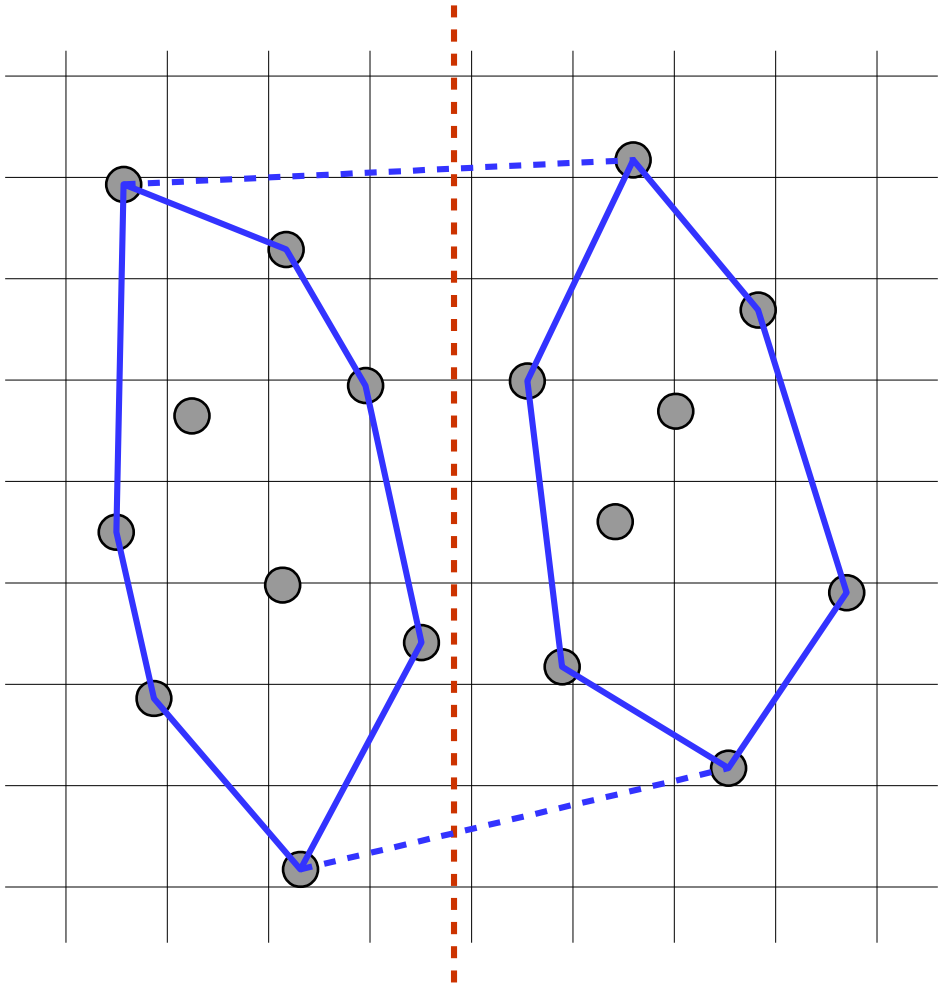
Divide and conquer



Answer:

- Divide the points into 2 equal parts,
- Construct convex hulls for every part (recursively),

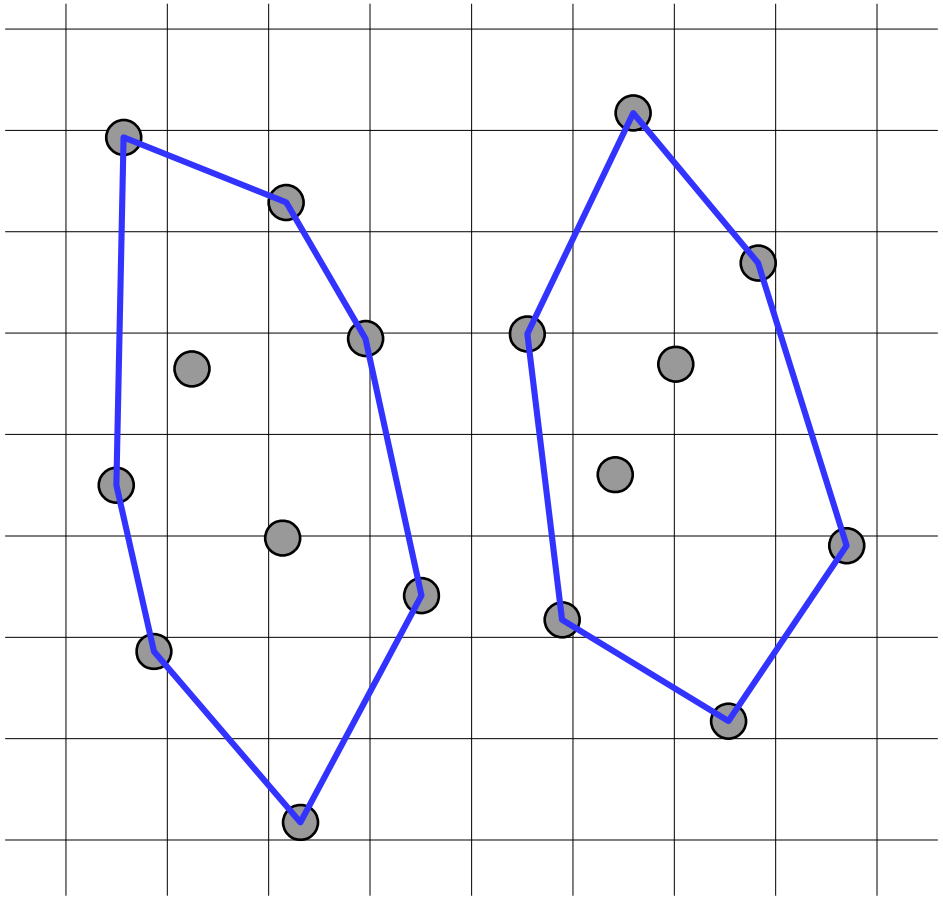
Divide and conquer



Answer:

- Divide the points into 2 equal parts,
- Construct convex hulls for every part (recursively),
- Merge them.

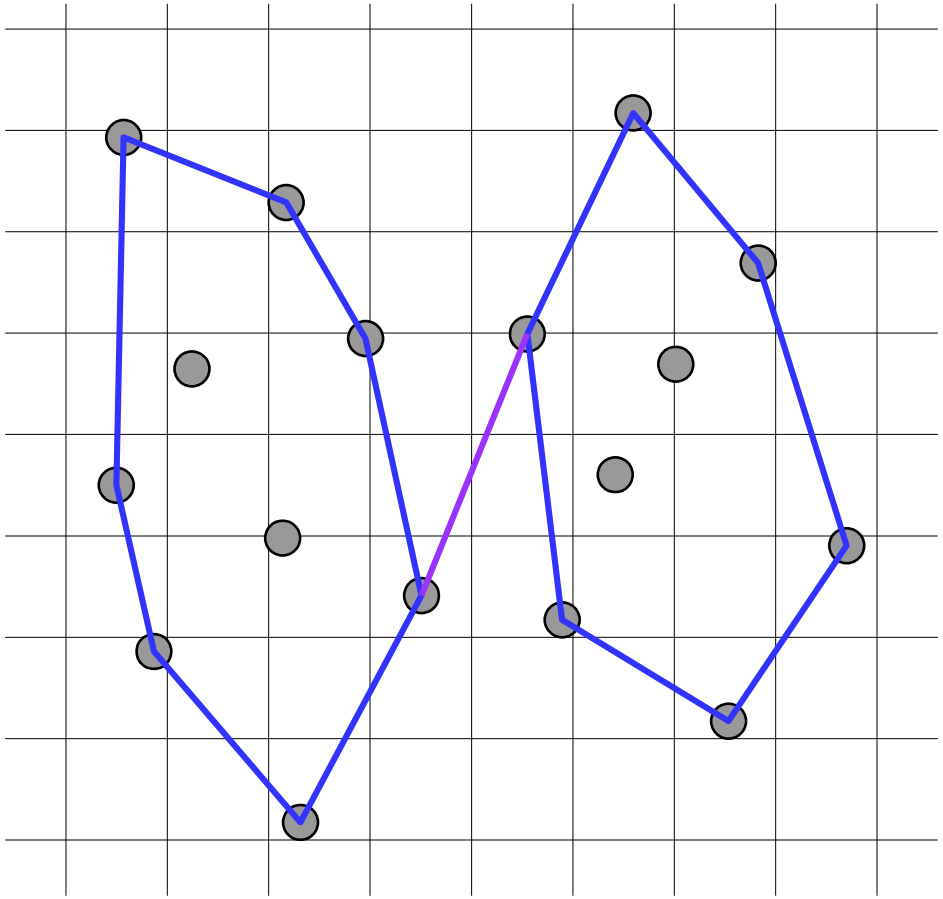
Divide and conquer



How to merge the **2** hulls?

... note, they always don't intersect.

Divide and conquer



How to merge the **2** hulls?

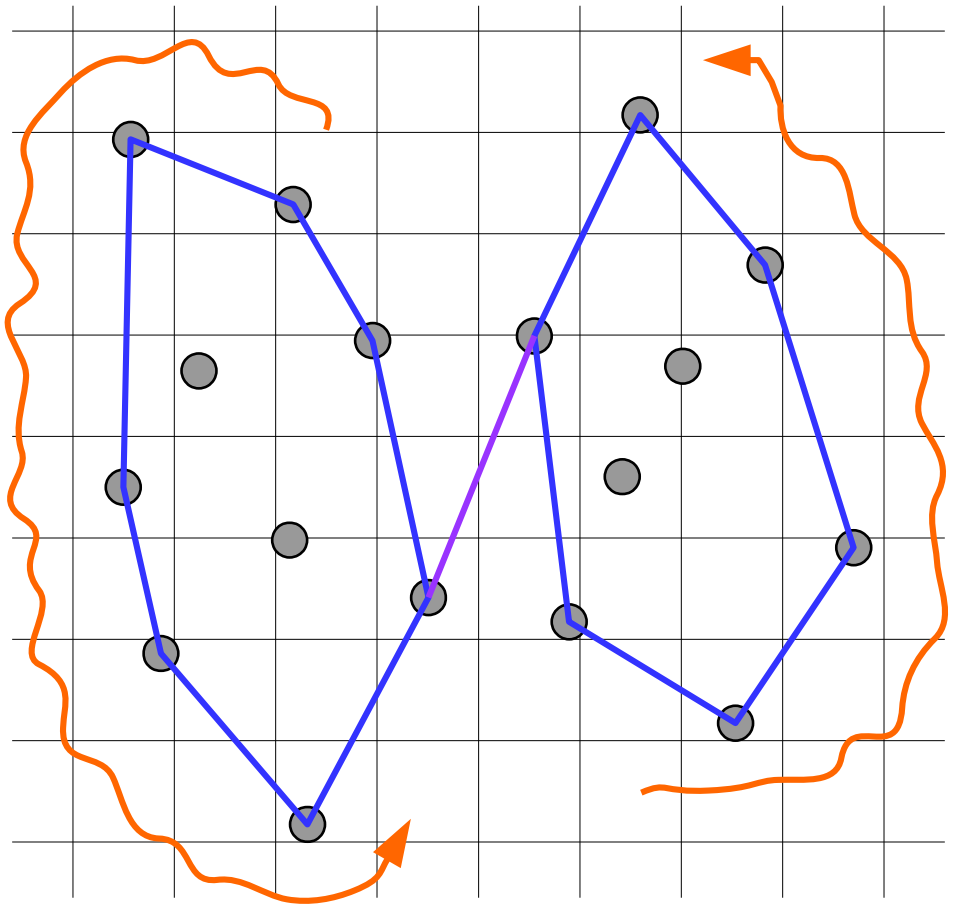
... note, they always don't intersect.

Let's connect them:

-> rightmost point of left hull,

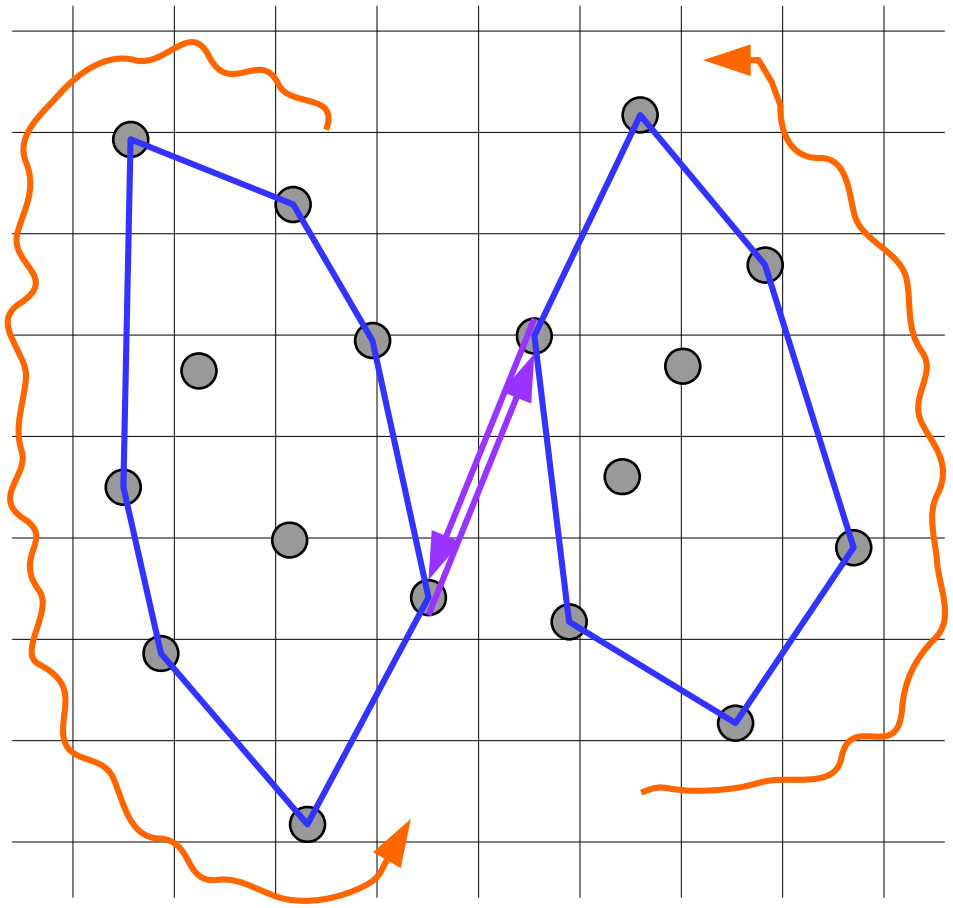
-> leftmost point of right hull.

Divide and conquer



Remember, that every hull can be interpreted also as cycling around all the input points.

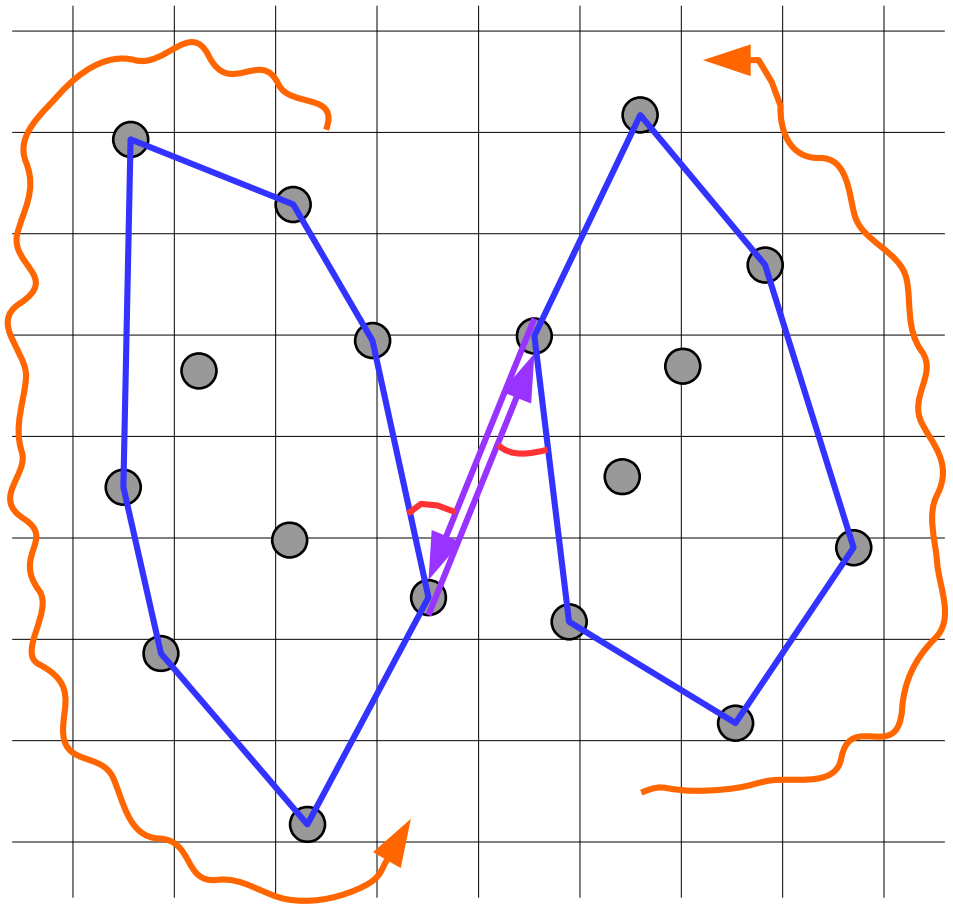
Divide and conquer



Remember, that every hull can be interpreted also as cycling around all the input points.

Now, adding the "bridge" can be interpreted this way:

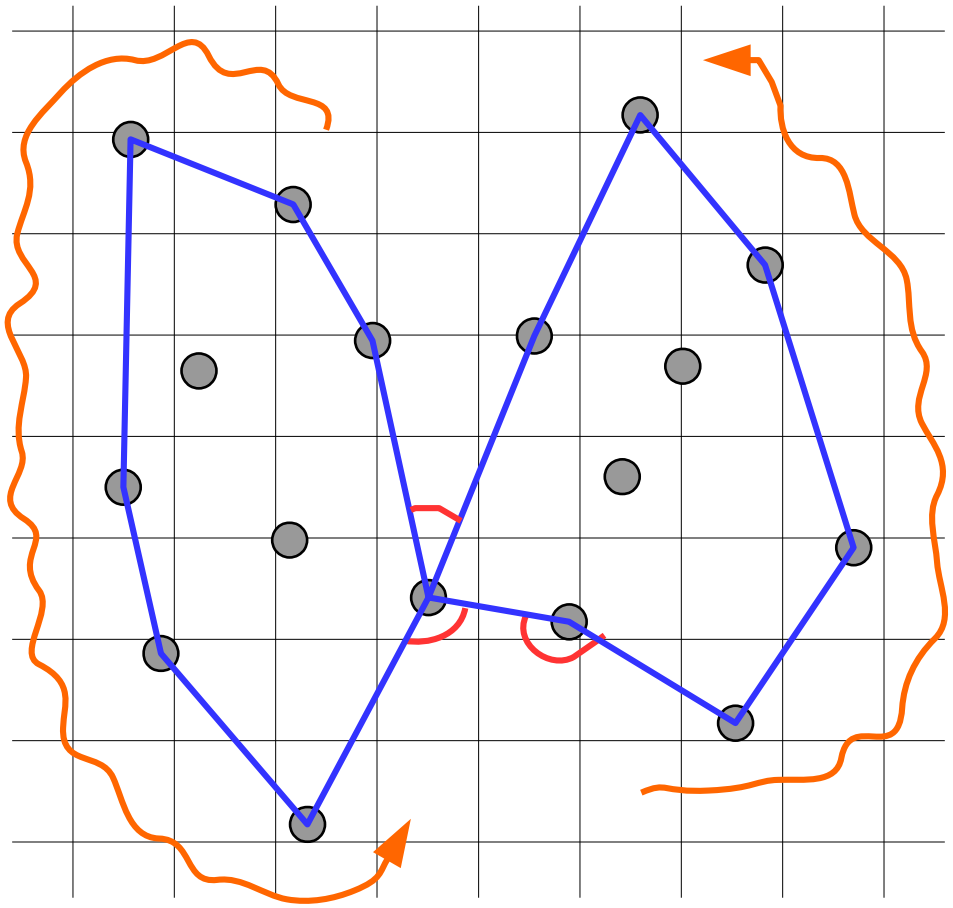
Divide and conquer



Having this, it will remain only to exclude "right turns",

... the way we were doing in the previous algorithm.

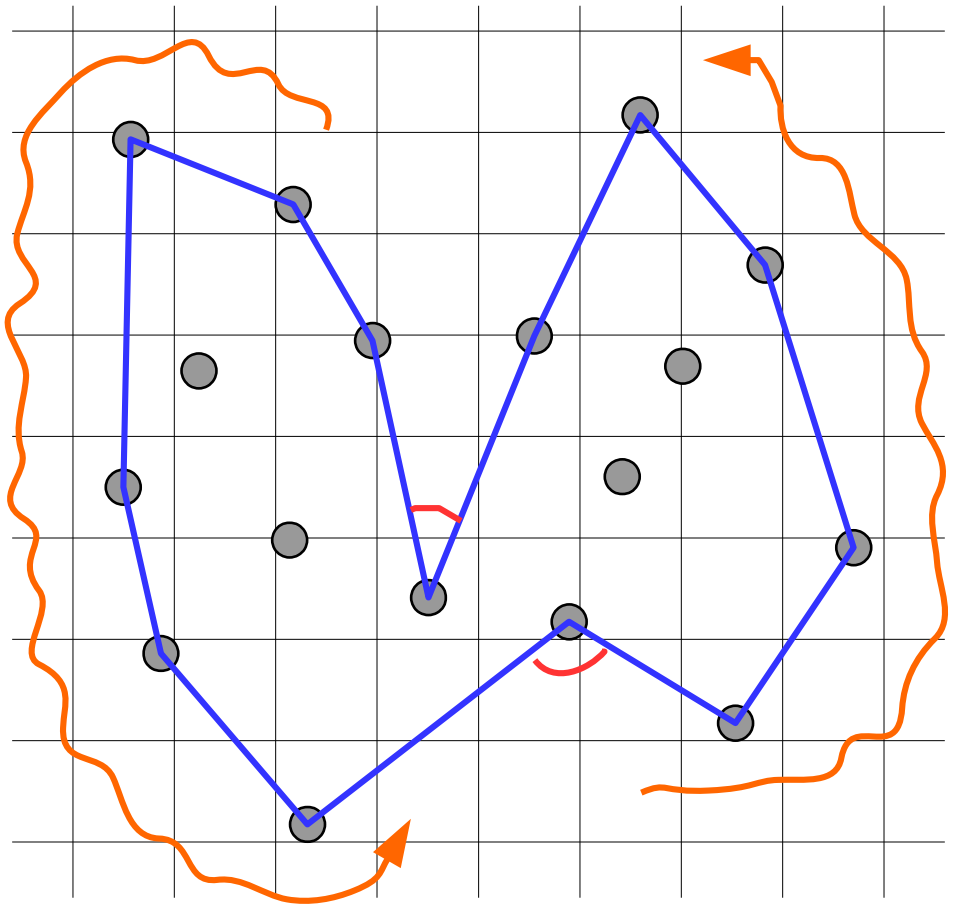
Divide and conquer



Having this, it will remain only to exclude "right turns",

... the way we were doing in the previous algorithm.

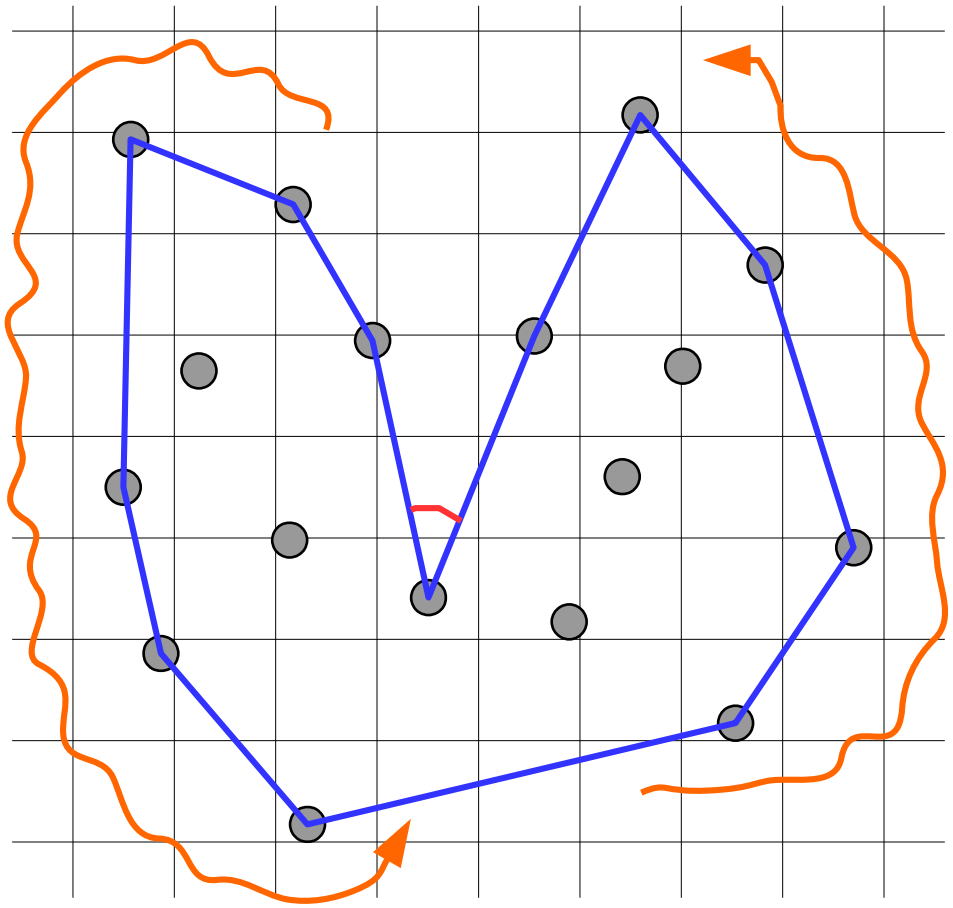
Divide and conquer



Having this, it will remain only to exclude "right turns",

... the way we were doing in the previous algorithm.

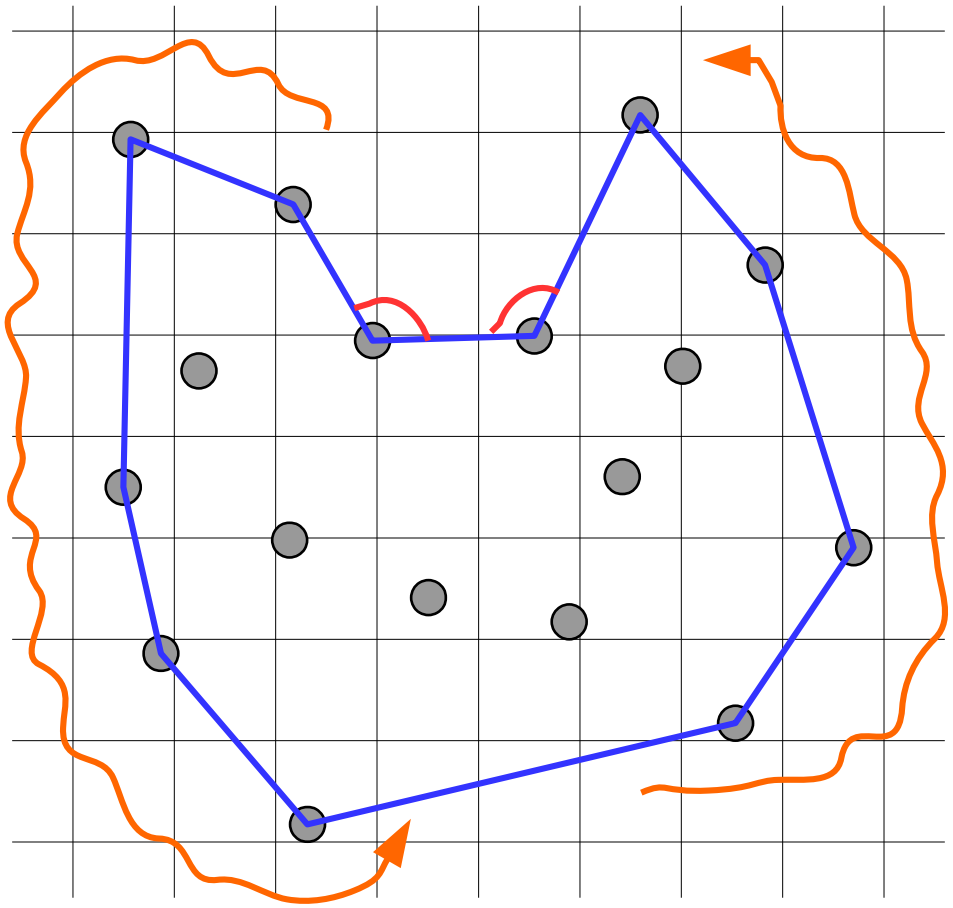
Divide and conquer



Having this, it will remain only to exclude "right turns",

... the way we were doing in the previous algorithm.

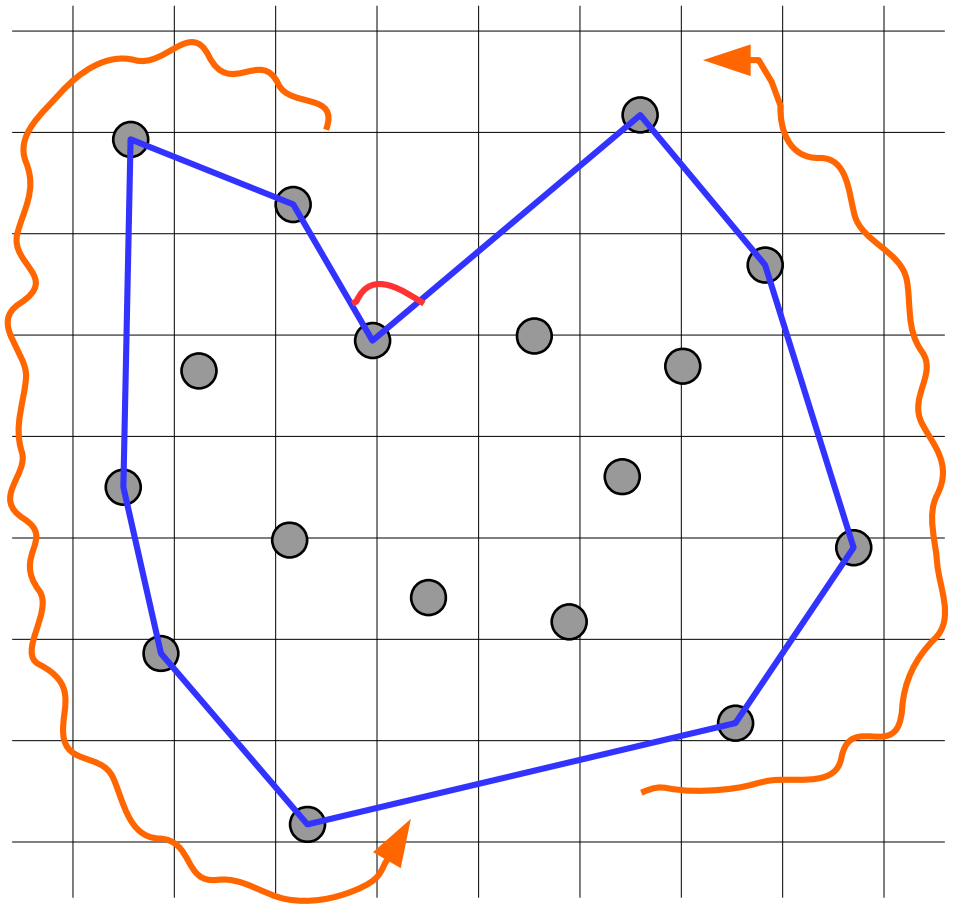
Divide and conquer



Having this, it will remain only to exclude "right turns",

... the way we were doing in the previous algorithm.

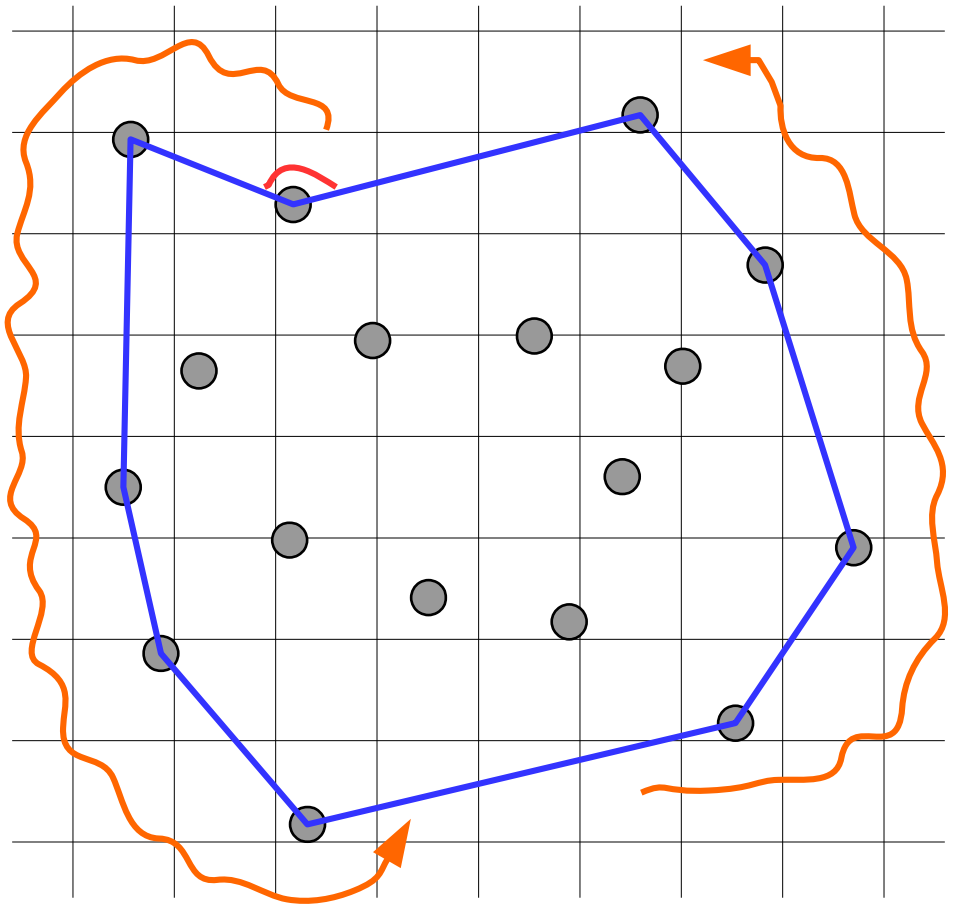
Divide and conquer



Having this, it will remain only to exclude "right turns",

... the way we were doing in the previous algorithm.

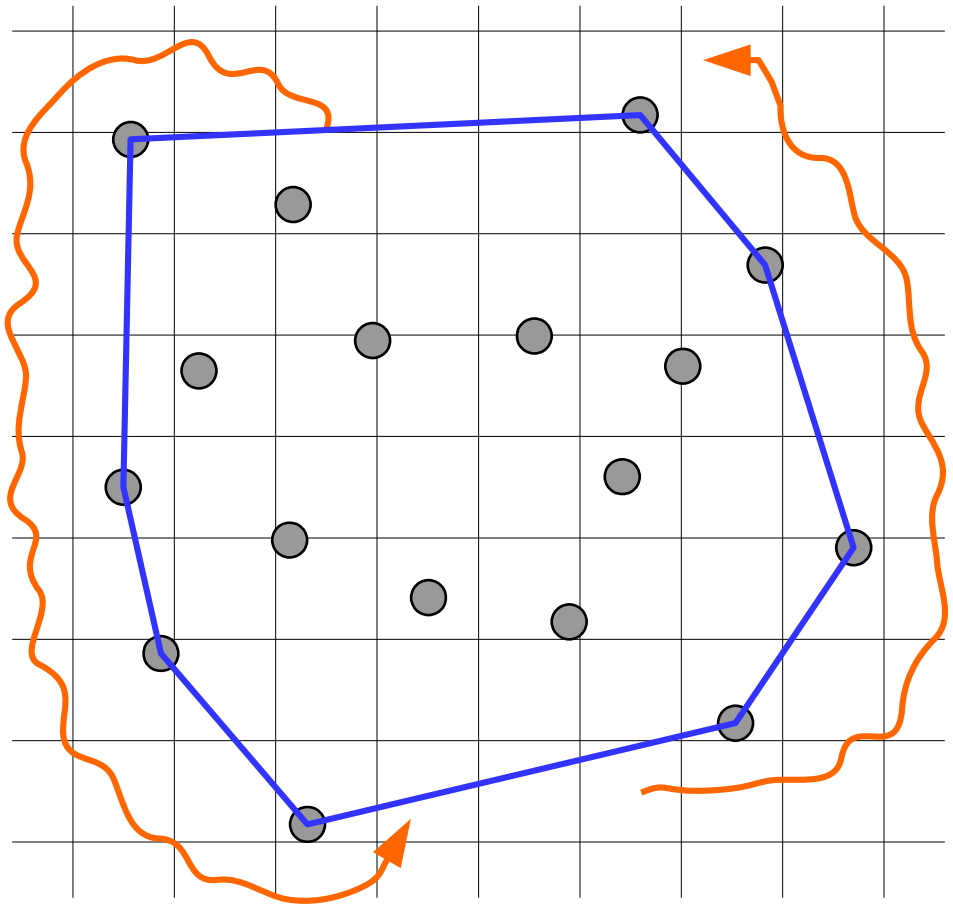
Divide and conquer



Having this, it will remain only to exclude "right turns",

... the way we were doing in the previous algorithm.

Divide and conquer



Having this, it will remain only to exclude "right turns",

... the way we were doing in the previous algorithm.

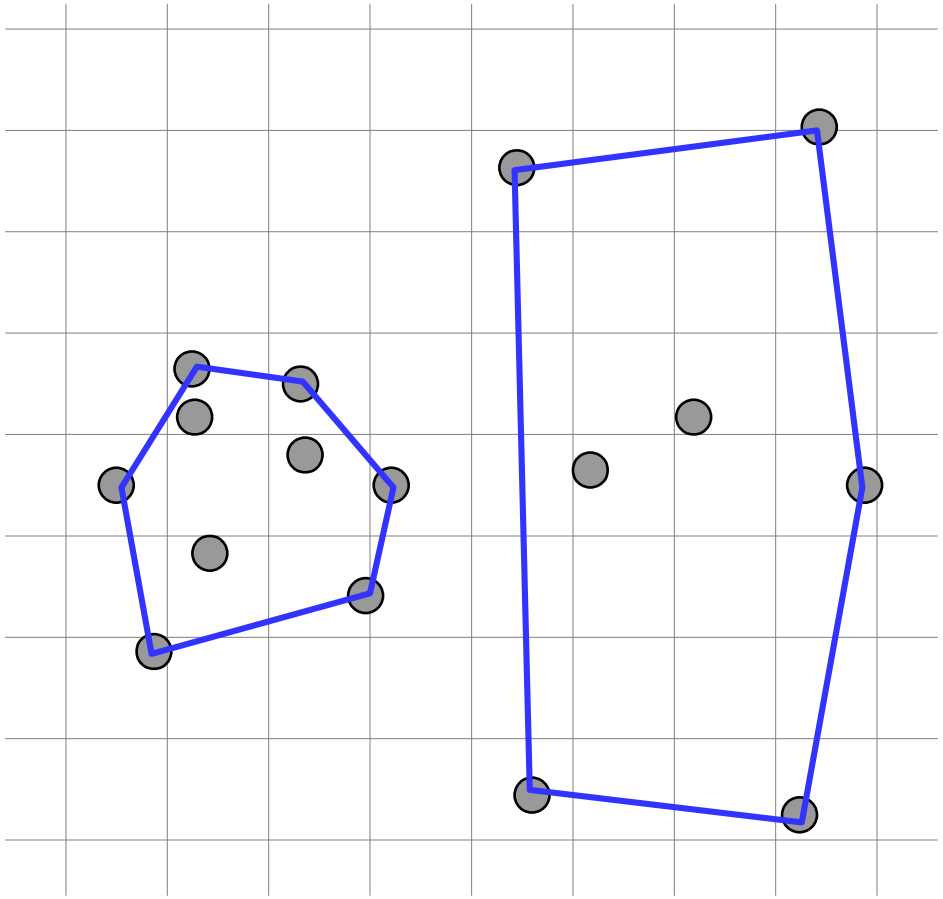
Time complexity

Calculating time complexity:

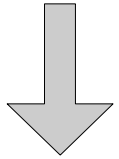
$$f(n) = f(n/2) + f(n/2) + n = 2*f(n/2) + n =$$
$$= 2*(2*f(n/4)+n/2) + n = 4*f(n/4) + 2n =$$
$$= 4*(2*f(n/8)+n/4) + 2n = 8*f(n/8) + 3n =$$
$$= \ldots =$$
$$= n * \log n$$

So here the logarithm originates not from sorting, but from partitioning.

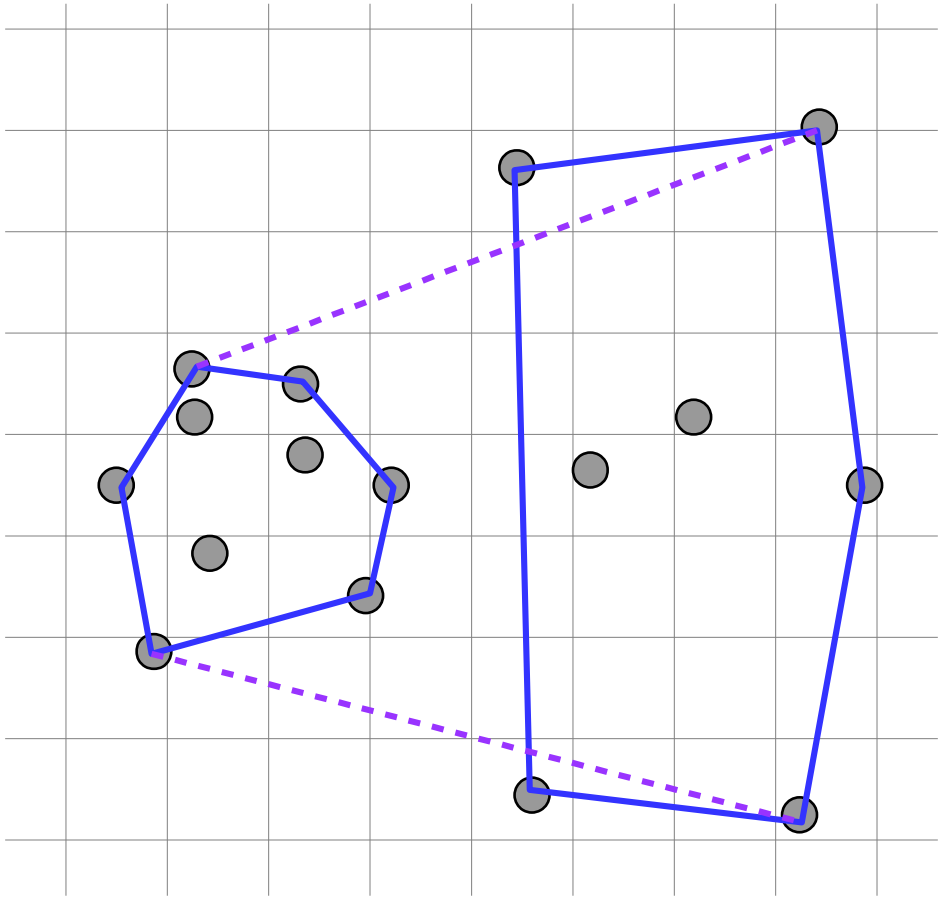
Divide and conquer



Note, in order to merge the **2** convex hulls, we can't just take the bottom-most pair and the top-most pair.

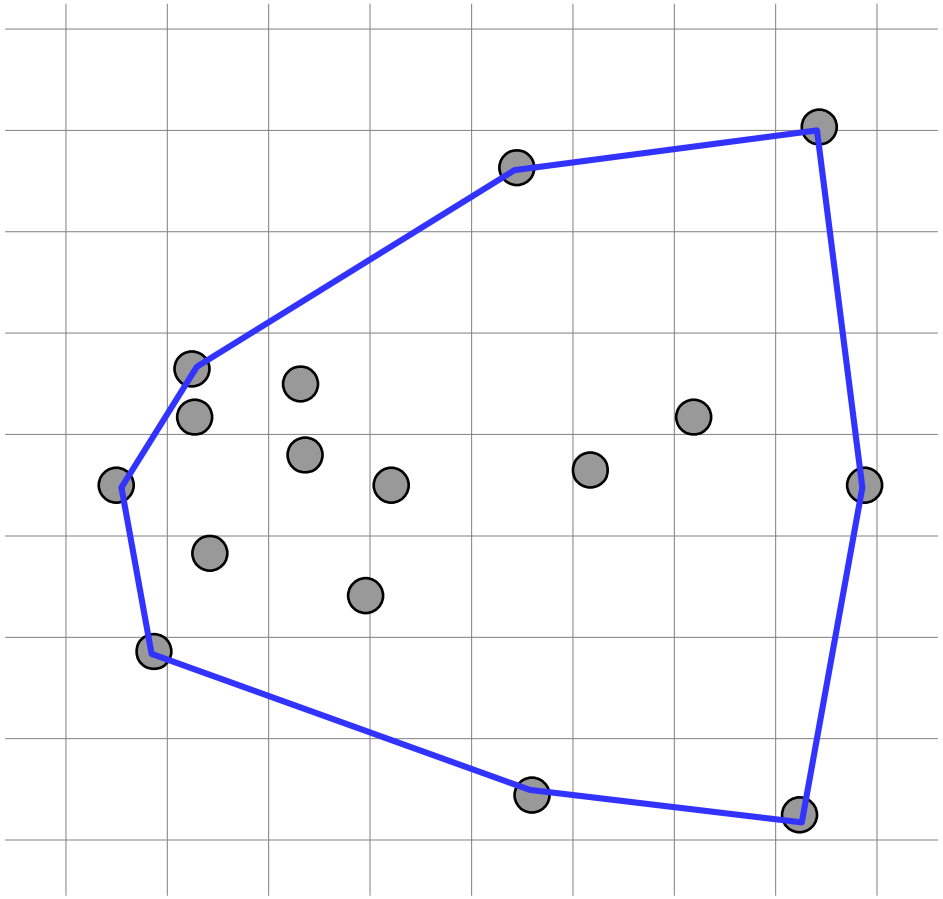


Divide and conquer



Note, in order to merge the **2** convex hulls, we can't just take the bottom-most pair and the top-most pair.

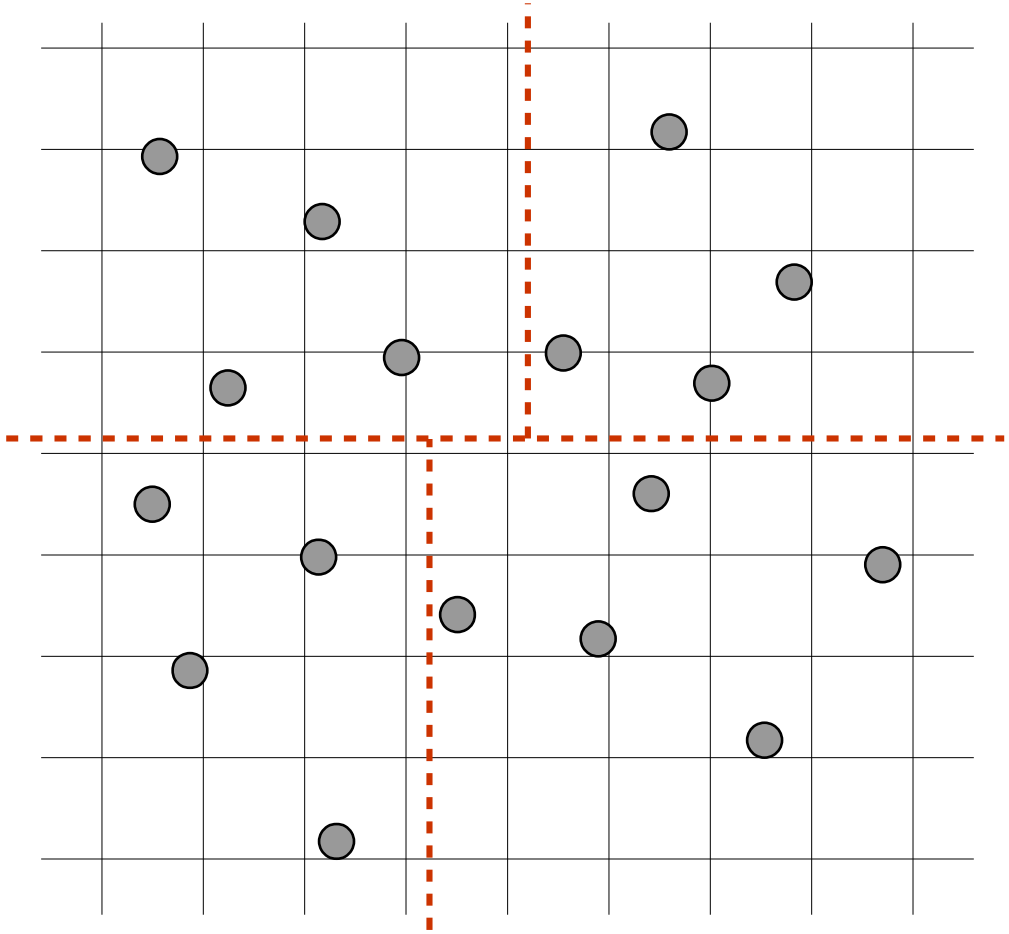
Divide and conquer



Note, in order to merge the **2** convex hulls, we can't just take the bottom-most pair and the top-most pair.

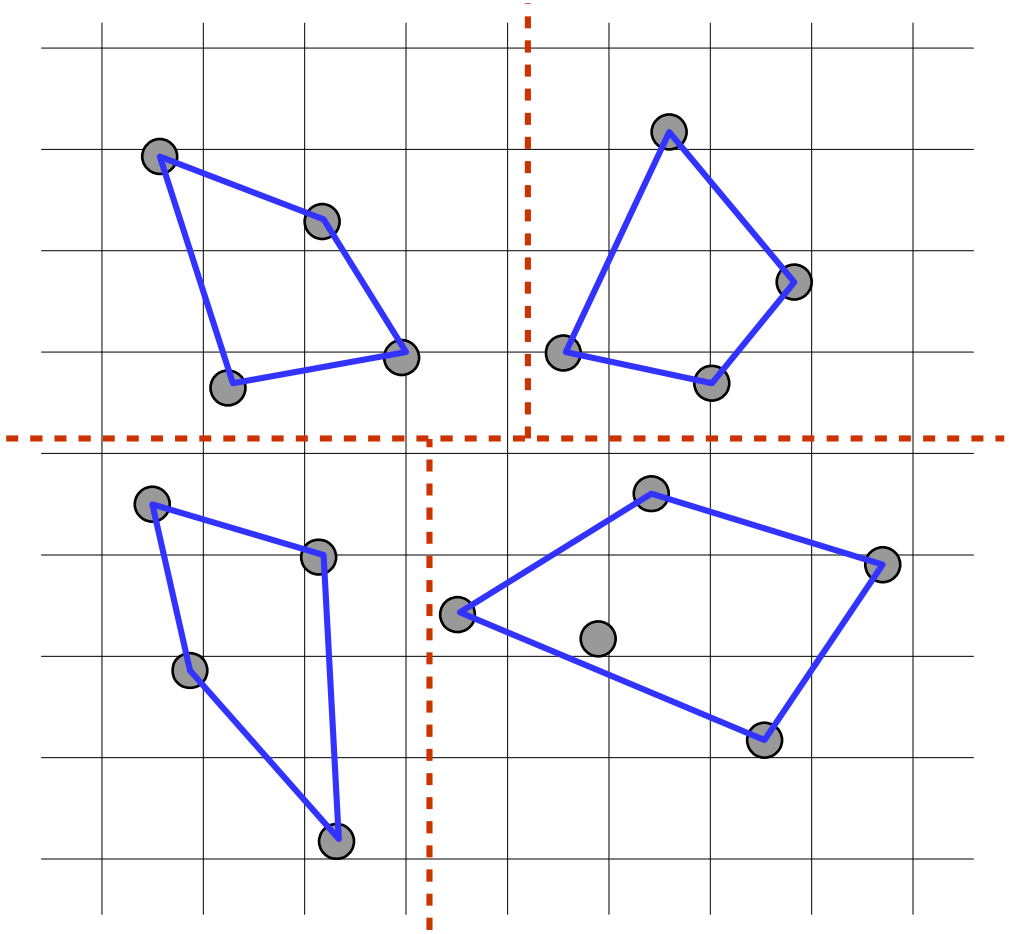
... actual result of merging is this.

Divide and conquer



Question: Can we interleave directions of the split: once horizontally, then vertically, and so on...

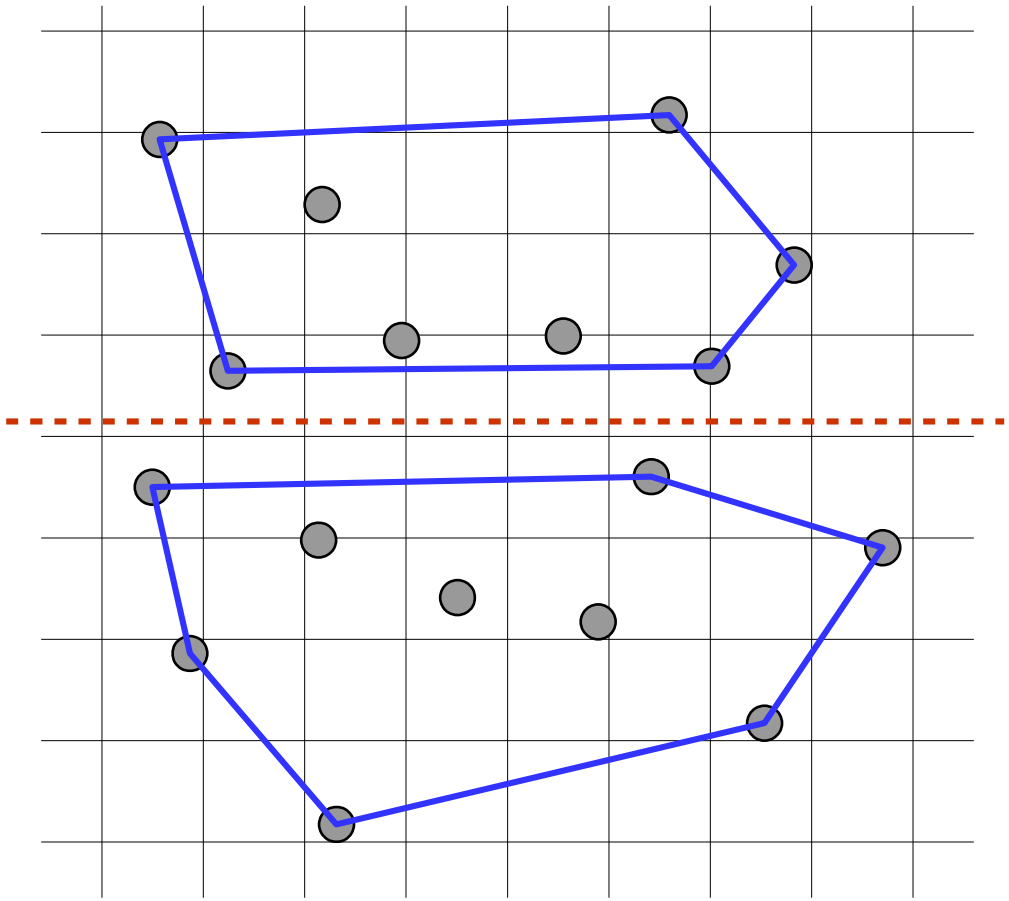
Divide and conquer



Question: Can we interleave directions of the split: once horizontally, then vertically, and so on...

Answer: Yes.

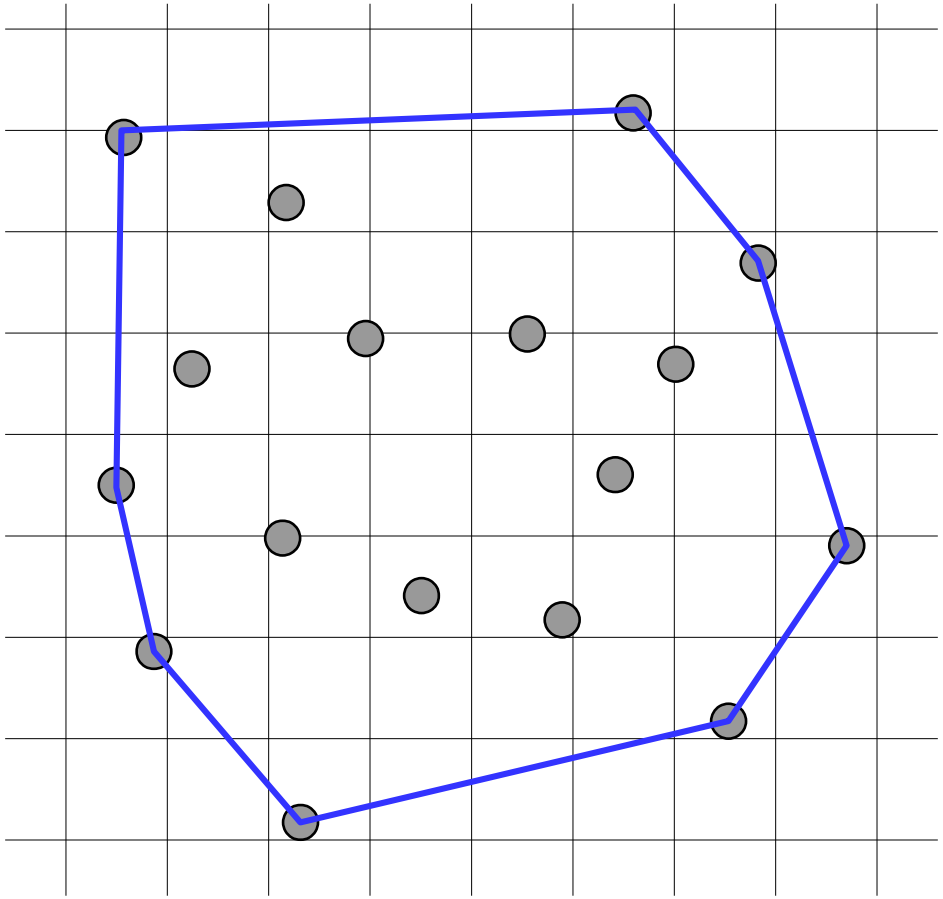
Divide and conquer



Question: Can we interleave directions of the split: once horizontally, then vertically, and so on...

Answer: Yes.

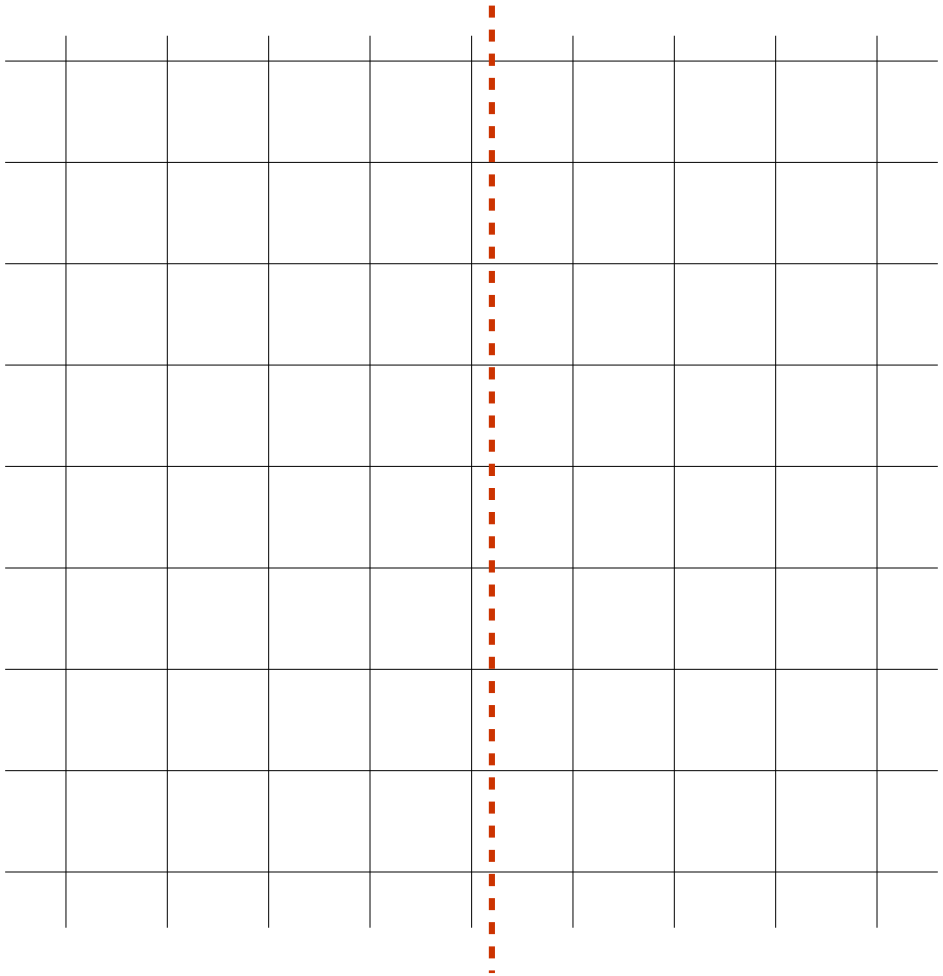
Divide and conquer



Question: Can we interleave directions of the split: once horizontally, then vertically, and so on...

Answer: Yes.

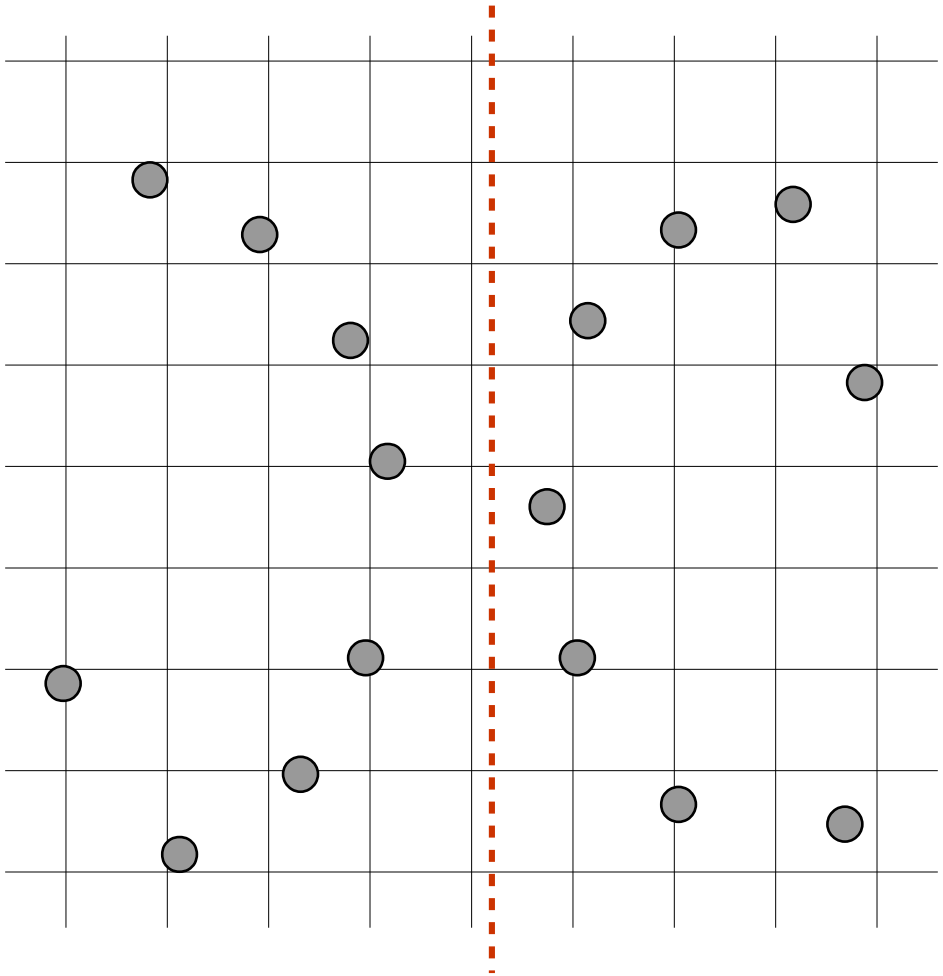
Divide and conquer



Question: Describe a case, when:

- final '**h**' is small,
- much time is spent on the final merge.

Divide and conquer

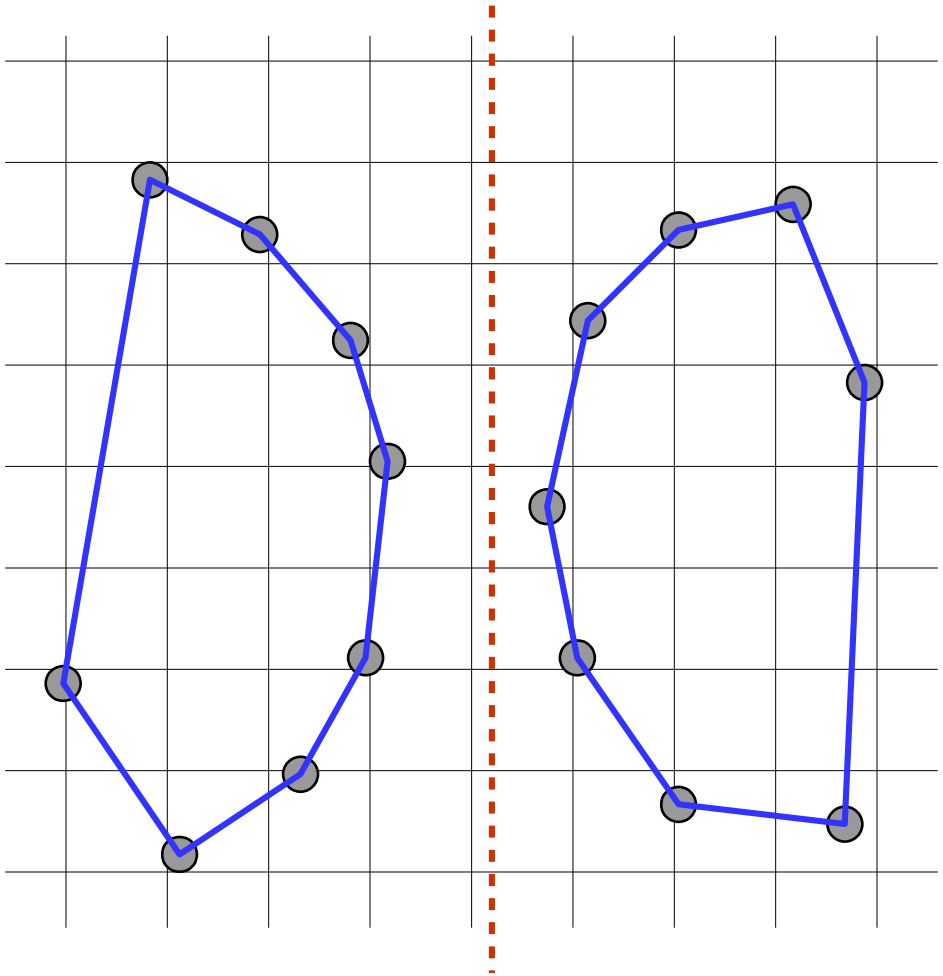


Question: Describe a case, when:

- final '**h**' is small,
- much time is spent on the final merge.

Answer: <---

Divide and conquer



Question: Describe a case, when:

- final '**h**' is small,
- much time is spent on the final merge.

Answer: <---

Exercise

Draw on paper **n=16** points, and perform first **2** levels of the algorithm.

Presentation writer: Tigran Hayrapetyan

Lecturer | Programmer | Researcher

www.linkedin.com/in/tigran-hayrapetyan-cs/

Thank you!

Convex hull

(divide and conquer algorithm)