H — Hiring Help

Problem

Given productivity values ℓ_i, f_i of n coders and productivity ℓ, f of consultant for t-hour long project, is there a weighted average of coders such that $\ell_{\text{avg}} \geq \ell/t$ and $f_{\text{avg}} \geq f/t$? Handle many queries like this, interleaved with some of the n coders leaving.

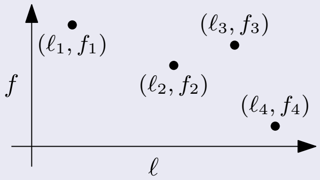
H — Hiring Help

Problem

Given productivity values ℓ_i, f_i of n coders and productivity ℓ, f of consultant for t-hour long project, is there a weighted average of coders such that $\ell_{\text{avg}} \geq \ell/t$ and $f_{\text{avg}} \geq f/t$? Handle many queries like this, interleaved with some of the n coders leaving.

Geometric View

1. The coders are a set of points P in 2D.



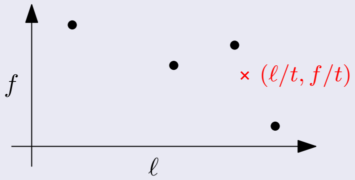
H — Hiring Help

Problem

Given productivity values ℓ_i, f_i of n coders and productivity ℓ, f of consultant for t-hour long project, is there a weighted average of coders such that $\ell_{\text{avg}} \geq \ell/t$ and $f_{\text{avg}} \geq f/t$? Handle many queries like this, interleaved with some of the n coders leaving.

Geometric View

1. The coders are a set of points P in 2D.
2. The consultant query is another point q in 2D.



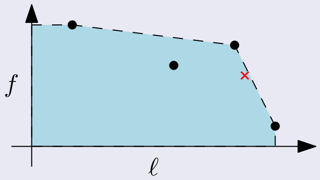
H — Hiring Help

Problem

Given productivity values ℓ_i, f_i of n coders and productivity ℓ, f of consultant for t-hour long project, is there a weighted average of coders such that $\ell_{\text{avg}} \geq \ell/t$ and $f_{\text{avg}} \geq f/t$? Handle many queries like this, interleaved with some of the n coders leaving.

Geometric View

1. The coders are a set of points P in 2D.
2. The consultant query is another point q in 2D.
3. The weighted average exists if and only if q is below the upper side of the convex hull of P.



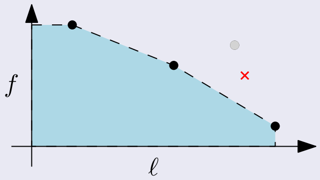
H — Hiring Help

Problem

Given productivity values ℓ_i, f_i of n coders and productivity ℓ, f of consultant for t-hour long project, is there a weighted average of coders such that $\ell_{\text{avg}} \geq \ell / t$ and $f_{\text{avg}} \geq f / t$? Handle many queries like this, interleaved with some of the n coders leaving.

Geometric View

1. The coders are a set of points P in 2D.
2. The consultant query is another point q in 2D.
3. The weighted average exists if and only if q is below the upper side of the convex hull of P.
4. Coders leaving corresponds to points being removed, leading to the convex hull changing.



H — Hiring Help

Reformulated Problem

Given set of points (x, y), maintain upper side of its convex hull, under removals of points and queries about whether other points (x^*, y^*) are below the hull.

H — Hiring Help

Reformulated Problem

Given set of points (x, y), maintain upper side of its convex hull, under removals of points and queries about whether other points (x^*, y^*) are below the hull.

Solution

1. The hull is a piecewise linear function, represent it as a sorted set of points $(x_1, y_1), (x_2, y_2), \ldots, (x_t, y_t)$ where $x_1 < x_2 < \ldots x_t$ and $y_1 > y_2 > \ldots y_t$.

H — Hiring Help

Reformulated Problem

Given set of points (x, y), maintain upper side of its convex hull, under removals of points and queries about whether other points (x^*, y^*) are below the hull.

Solution

1. The hull is a piecewise linear function, represent it as a sorted set of points $(x_1, y_1), (x_2, y_2), \ldots, (x_t, y_t)$ where $x_1 < x_2 < \ldots x_t$ and $y_1 > y_2 > \ldots y_t$.
2. For a query (x^*, y^*), find index i such that $x_{i-1} < x^* \leq x_i$ and check if (x^*, y^*) is below line from (x_{i-1}, y_{i-1}) to (x_i, y_i).

H — Hiring Help

Reformulated Problem

Given set of points (x, y), maintain upper side of its convex hull, under removals of points and queries about whether other points (x^*, y^*) are below the hull.

Solution

1. The hull is a piecewise linear function, represent it as a sorted set of points $(x_1, y_1), (x_2, y_2), \ldots, (x_t, y_t)$ where $x_1 < x_2 < \ldots x_t$ and $y_1 > y_2 > \ldots y_t$.
2. For a query (x^*, y^*), find index i such that $x_{i-1} < x^* \leq x_i$ and check if (x^*, y^*) is below line from (x_{i-1}, y_{i-1}) to (x_i, y_i).
3. Handling removals can be done,

H — Hiring Help

Reformulated Problem

Given set of points (x, y), maintain upper side of its convex hull, under *additions* of points and queries about whether other points (x^*, y^*) are below the hull.

Solution

1. The hull is a piecewise linear function, represent it as a sorted set of points $(x_1, y_1), (x_2, y_2), \ldots, (x_t, y_t)$ where $x_1 < x_2 < \ldots x_t$ and $y_1 > y_2 > \ldots y_t$.
2. For a query (x^*, y^*), find index i such that $x_{i-1} < x^* \leq x_i$ and check if (x^*, y^*) is below line from (x_{i-1}, y_{i-1}) to (x_i, y_i).
3. Handling removals can be done, but if we instead run the events in reverse order, the removals become *additions*, which are easier to handle.

H — Hiring Help

Reformulated Problem

Given set of points (x, y), maintain upper side of its convex hull, under *additions* of points and queries about whether other points (x^*, y^*) are below the hull.

Handling additions

1. If point to add is outside current hull, add it to our current set of points.

H — Hiring Help

Reformulated Problem

Given set of points (x, y), maintain upper side of its convex hull, under *additions* of points and queries about whether other points (x^*, y^*) are below the hull.

Handling additions

1. If point to add is outside current hull, add it to our current set of points.
2. Remove any concavities formed to left and right of new point.

H — Hiring Help

Reformulated Problem

Given set of points (x, y), maintain upper side of its convex hull, under *additions* of points and queries about whether other points (x^*, y^*) are below the hull.

Handling additions

1. If point to add is outside current hull, add it to our current set of points.
2. Remove any concavities formed to left and right of new point.
3. Issue(?): addition may take a long time because many old points could be discarded from hull.

H — Hiring Help

Reformulated Problem

Given set of points (x, y), maintain upper side of its convex hull, under *additions* of points and queries about whether other points (x^*, y^*) are below the hull.

Handling additions

1. If point to add is outside current hull, add it to our current set of points.
2. Remove any concavities formed to left and right of new point.
3. Issue(?): addition may take a long time because many old points could be discarded from hull.
4. Not an issue: once a point is discarded, it can never be added back, so total number of removals for all events is $\leq n$.

H — Hiring Help

Reformulated Problem

Given set of points (x, y), maintain upper side of its convex hull, under *additions* of points and queries about whether other points (x^*, y^*) are below the hull.

Handling additions

1. If point to add is outside current hull, add it to our current set of points.
2. Remove any concavities formed to left and right of new point.
3. Issue(?): addition may take a long time because many old points could be discarded from hull.
4. Not an issue: once a point is discarded, it can never be added back, so total number of removals for all events is $\leq n$.
5. $O((n + e) \log n)$ total time complexity.

H — Hiring Help

Reformulated Problem

Given set of points (x, y), maintain upper side of its convex hull, under *additions* of points and queries about whether other points (x^*, y^*) are below the hull.

Handling additions

1. If point to add is outside current hull, add it to our current set of points.
2. Remove any concavities formed to left and right of new point.
3. Issue(?): addition may take a long time because many old points could be discarded from hull.
4. Not an issue: once a point is discarded, it can never be added back, so total number of removals for all events is $\leq n$.
5. $O((n + e) \log n)$ total time complexity.

Statistics at 4-hour mark: 24 submissions, 7 accepted, first after 01:42