Computational Geometry

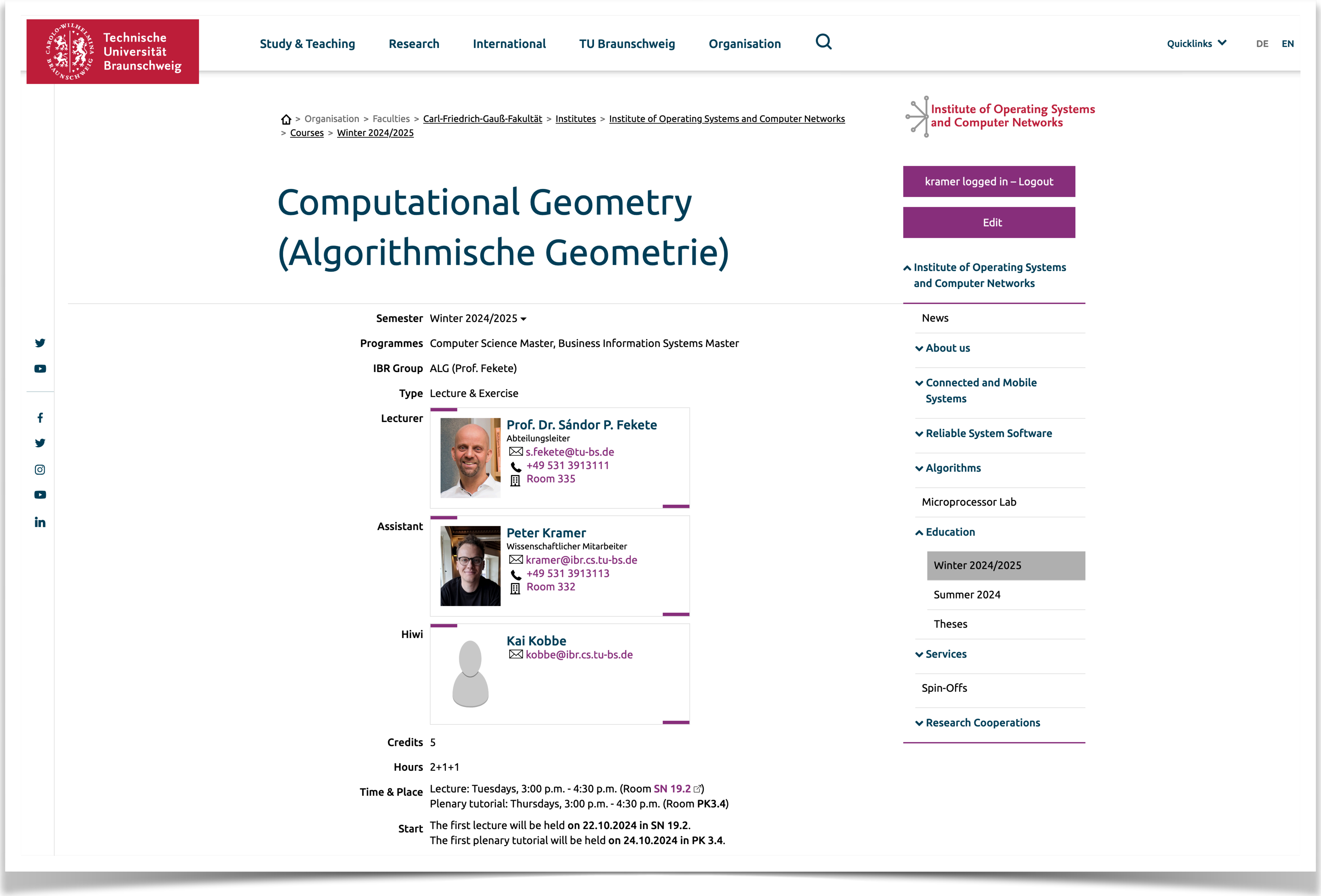
Tutorial #1 — Organisation and Convex Hulls

Institut für Betriebssysteme
und Rechnerverbund
Algorithmik

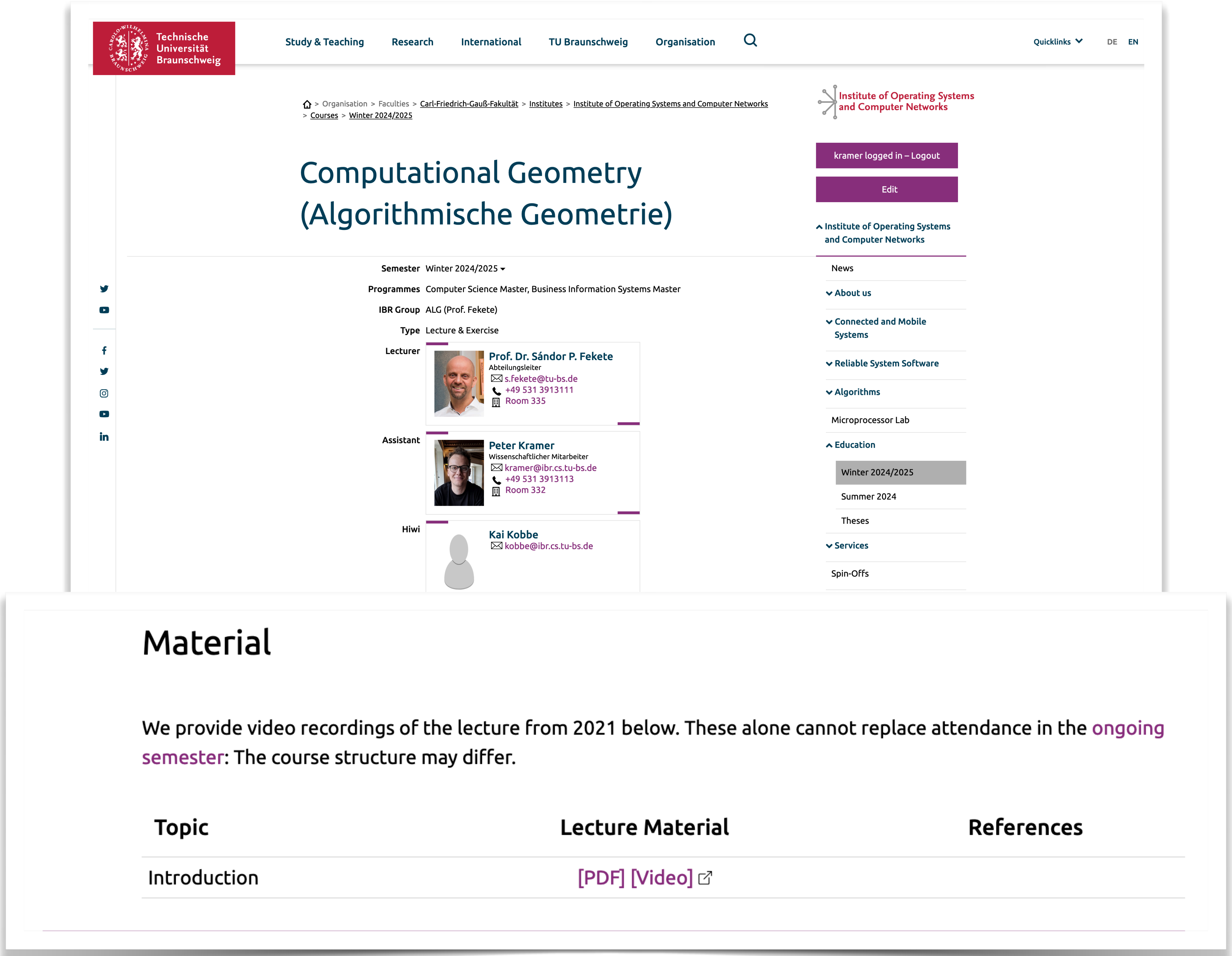
Peter Kramer | *October 24th, 2024*

Organisation

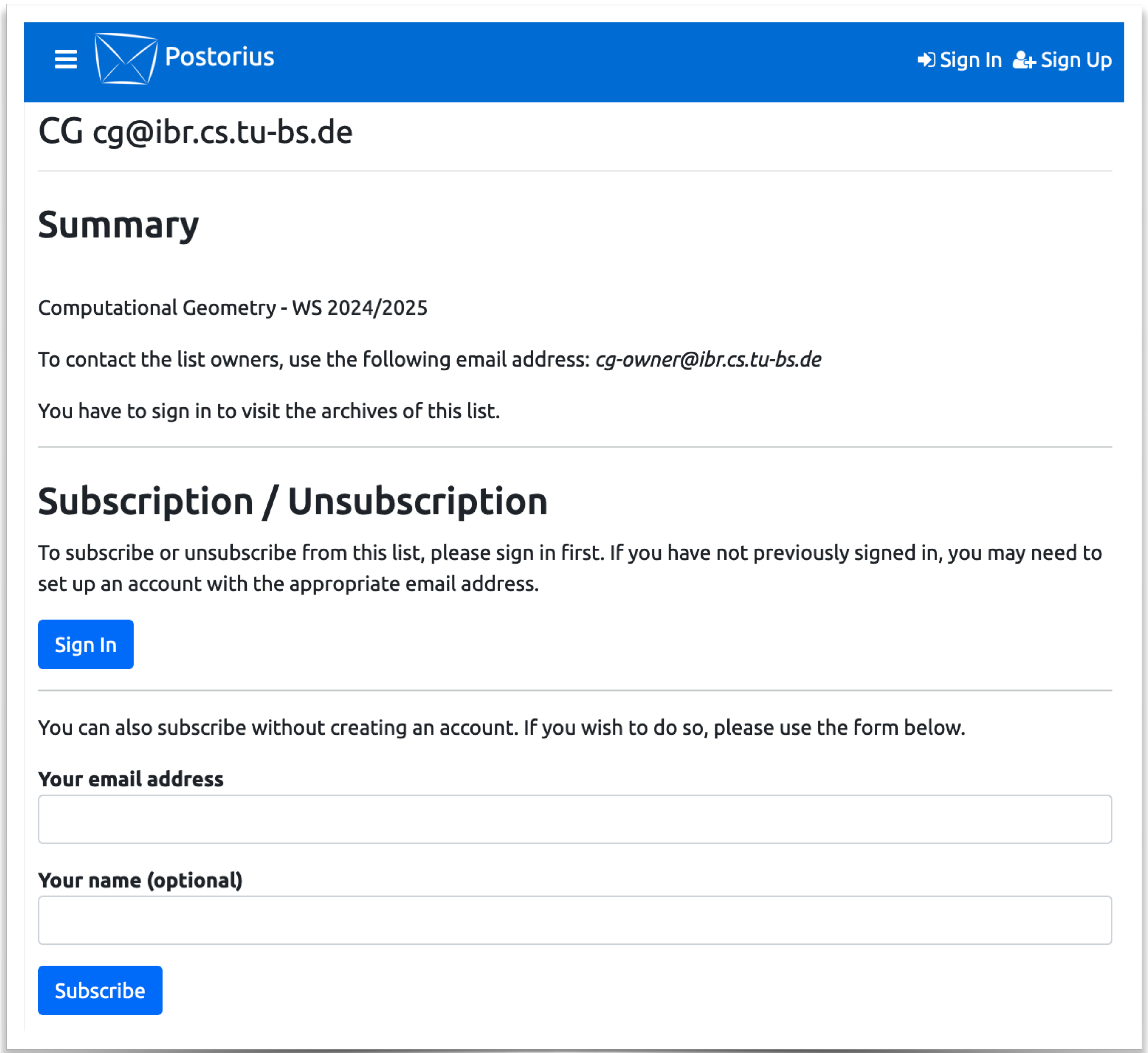
Website


Technische Universität Braunschweig
Study & Teaching
Research
International
TU Braunschweig
Organisation
Quicklinks
DE EN
> Organisation > Faculties > Carl-Friedrich-Gauß-Fakultät > Institutes > Institute of Operating Systems and Computer Networks > Courses > Winter 2024/2025
Institute of Operating Systems and Computer Networks
kramer logged in – Logout
Edit
Computational Geometry (Algorithmische Geometrie)
Semester Winter 2024/2025
Programmes Computer Science Master, Business Information Systems Master
IBR Group ALG (Prof. Fekete)
Type Lecture & Exercise
Lecturer
Prof. Dr. Sándor P. Fekete
Abteilungsleiter
s.fekete@tu-bs.de
+49 531 3913111
Room 335
Assistant
Peter Kramer
Wissenschaftlicher Mitarbeiter
kramer@ibr.cs.tu-bs.de
+49 531 3913113
Room 332
Hiwi
Kai Kobbe
kobbe@ibr.cs.tu-bs.de
Credits 5
Hours 2+1+1
Time & Place Lecture: Tuesdays, 3:00 p.m. - 4:30 p.m. (Room SN 19.2)
Plenary tutorial: Thursdays, 3:00 p.m. - 4:30 p.m. (Room PK3.4)
Start The first lecture will be held on 22.10.2024 in SN 19.2.
The first plenary tutorial will be held on 24.10.2024 in PK 3.4.
Institute of Operating Systems and Computer Networks
News
About us
Connected and Mobile Systems
Reliable System Software
Algorithms
Microprocessor Lab
Education
Winter 2024/2025
Summer 2024
Theses
Services
Spin-Offs
Research Cooperations

Website



Technische Universität Braunschweig
Study & Teaching
Research
International
TU Braunschweig
Organisation
Quicklinks
DE EN
> Organisation > Faculties > Carl-Friedrich-Gauß-Fakultät > Institutes > Institute of Operating Systems and Computer Networks > Courses > Winter 2024/2025
Institute of Operating Systems and Computer Networks
kramer logged in – Logout
Edit
Computational Geometry (Algorithmische Geometrie)
Semester Winter 2024/2025
Programmes Computer Science Master, Business Information Systems Master
IBR Group ALG (Prof. Fekete)
Type Lecture & Exercise
Lecturer
Prof. Dr. Sándor P. Fekete
Abteilungsleiter
s.fekete@tu-bs.de
+49 531 3913111
Room 335
Assistant
Peter Kramer
Wissenschaftlicher Mitarbeiter
kramer@ibr.cs.tu-bs.de
+49 531 3913113
Room 332
Hiwi
Kai Kobbe
kobbe@ibr.cs.tu-bs.de
Institute of Operating Systems and Computer Networks
News
About us
Connected and Mobile Systems
Reliable System Software
Algorithms
Microprocessor Lab
Education
Winter 2024/2025
Summer 2024
Theses
Services
Spin-Offs
Material
We provide video recordings of the lecture from 2021 below. These alone cannot replace attendance in the ongoing semester: The course structure may differ.
Topic
Lecture Material
References
Introduction
[PDF] [Video]



Postorius
Sign In
Sign Up
CG cg@ibr.cs.tu-bs.de
Summary
Computational Geometry - WS 2024/2025
To contact the list owners, use the following email address: cg-owner@ibr.cs.tu-bs.de
You have to sign in to visit the archives of this list.
Subscription / Unsubscription
To subscribe or unsubscribe from this list, please sign in first. If you have not previously signed in, you may need to set up an account with the appropriate email address.
Sign In
You can also subscribe without creating an account. If you wish to do so, please use the form below.
Your email address
Your name (optional)
Subscribe

Tutorial

- **Every Thursday at 3pm** in this room, either a big or a small tutorial.
  - Depending on attendance, we might switch rooms.

- **Big tutorial:** Expand upon and put concepts from the lecture to use.
- **Small tutorial:** Homework discussion.

October						
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

November						
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

13 Lectures

7 Big Tutorials

4 Small Tutorials

4 Sheets

December						
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

January						
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

Exam prep?

Tutorial

- **Every Thursday at 3pm** in this room, either a big or a small tutorial.
  - Depending on attendance, we might switch rooms.

- **Big tutorial:** Expand upon and put concepts from the lecture to use.
- **Small tutorial:** Homework discussion.

- **Studienleistung:** Four homework sheets.
  - Biweekly assignments to solve in pairs.
  - 50% of total possible points on the sheets; roughly 70 - 75 points.

Exam...

The form of exam depends on the number of attending students.

Announcement in an upcoming lecture.

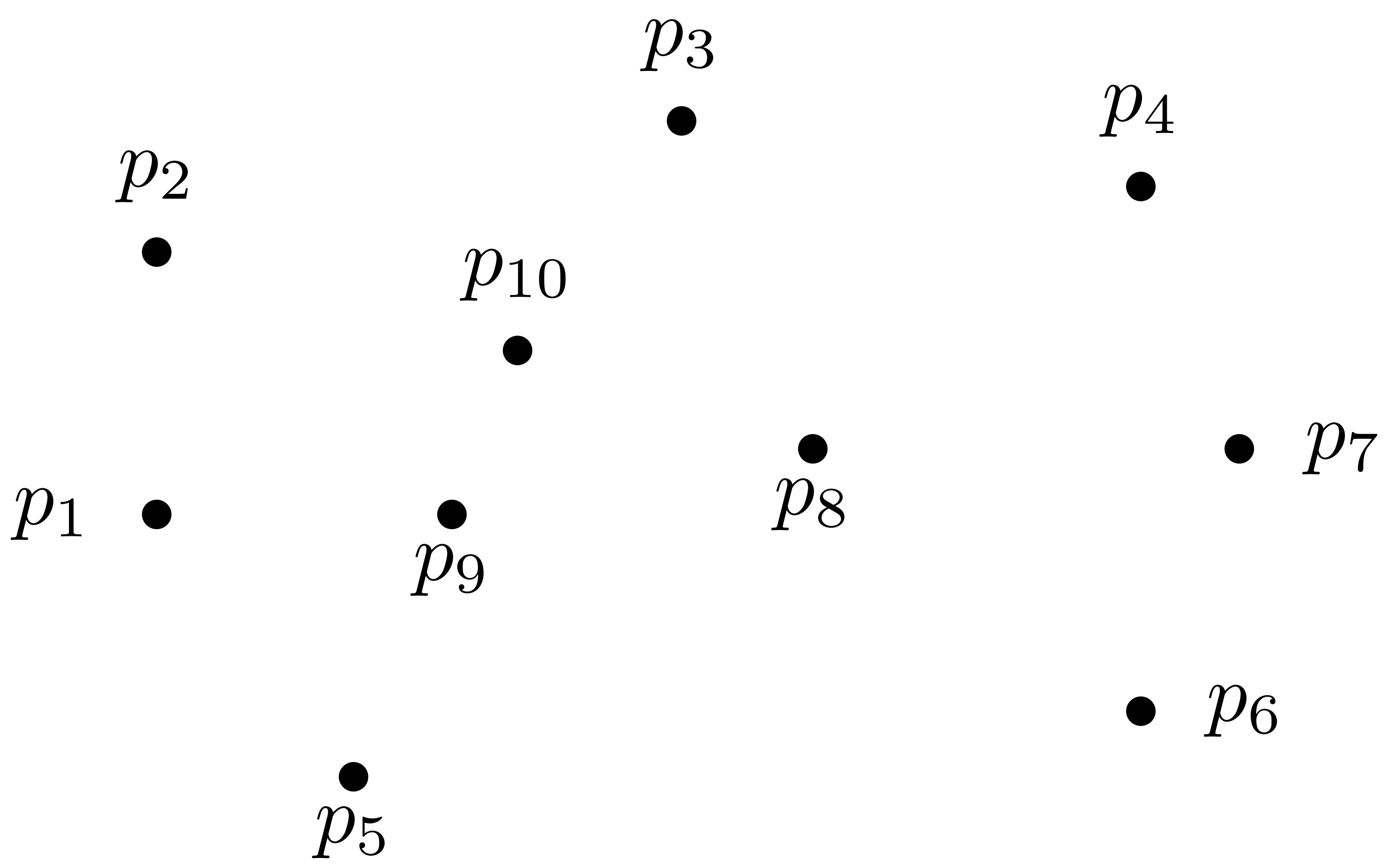
Convex Hulls

Peter Kramer | October 24th, 2024

The convex hull

… is a point set? A convex polygon? A subset?

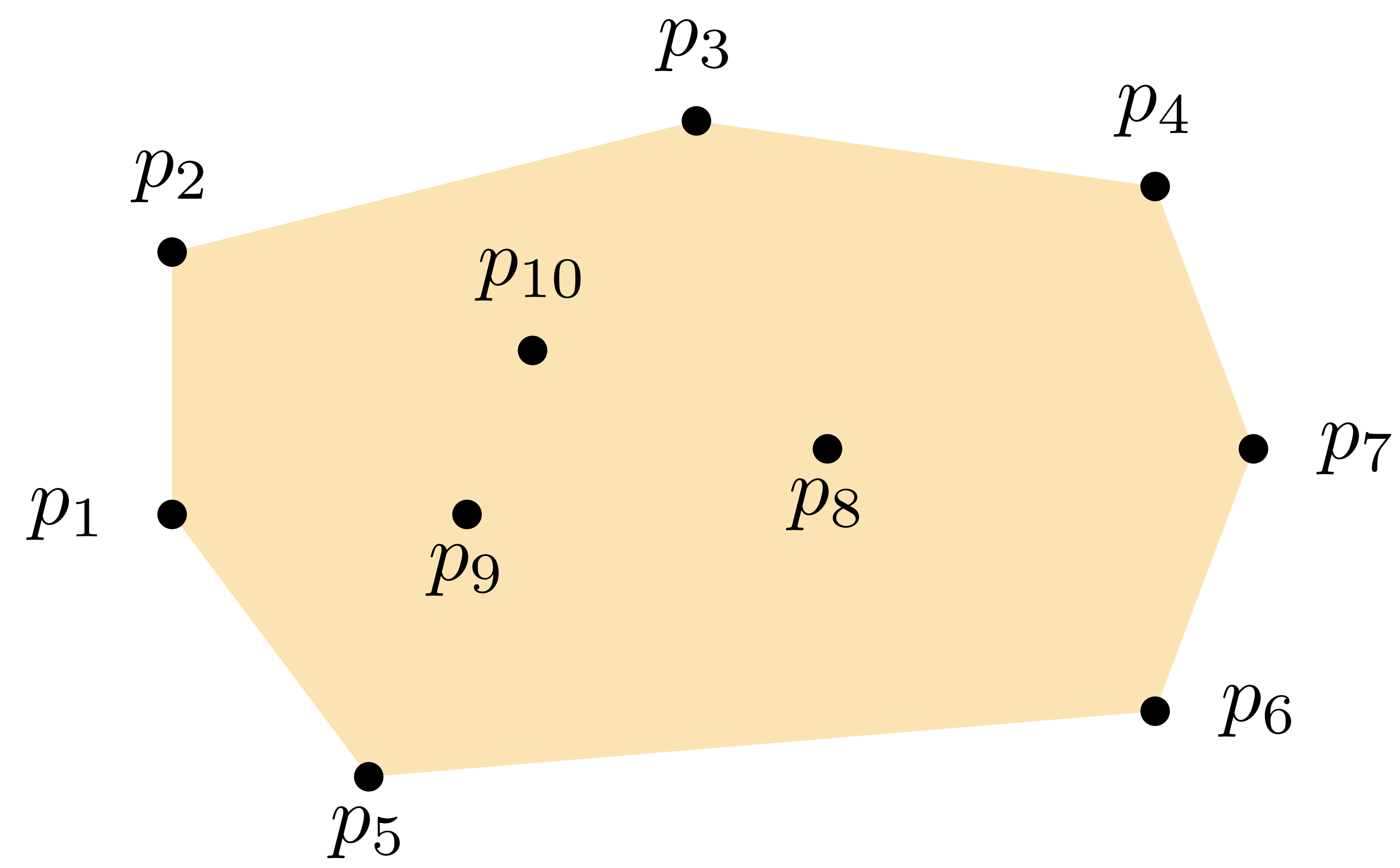
- For a finite point set $\mathscr{P} := \{p_1, p_2, \ldots, p_n\} \subset \mathbb{R}^2$, $\operatorname{conv}(\mathscr{P}) = \ldots ?$



The convex hull

… is a point set? A convex polygon? A subset?

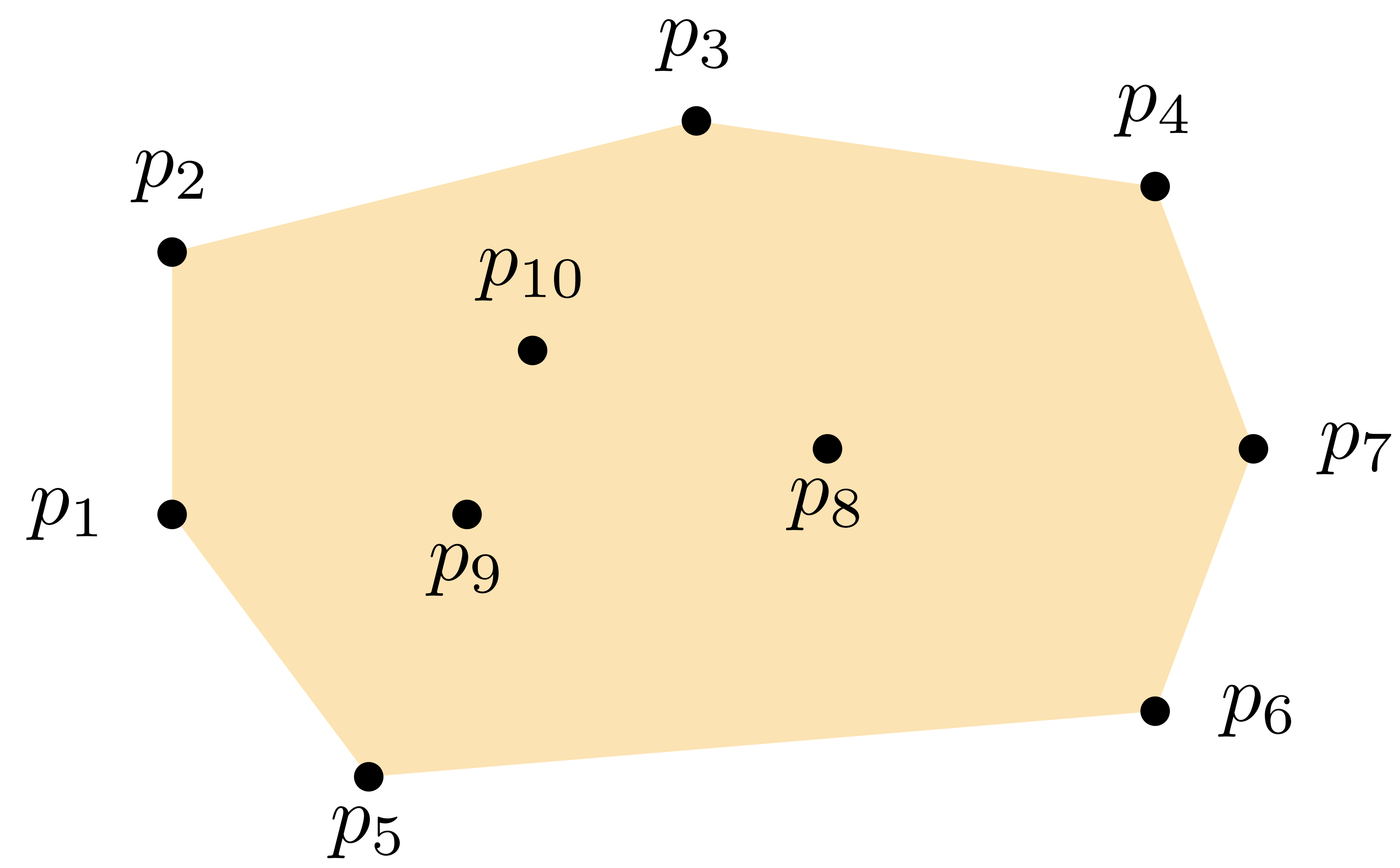
- For a finite point set $\mathscr{P} := \{p_1, p_2, \ldots, p_n\} \subset \mathbb{R}^2$, $\operatorname{conv}(\mathscr{P}) = \ldots$?



The convex hull

… is a point set? A convex polygon? A subset?

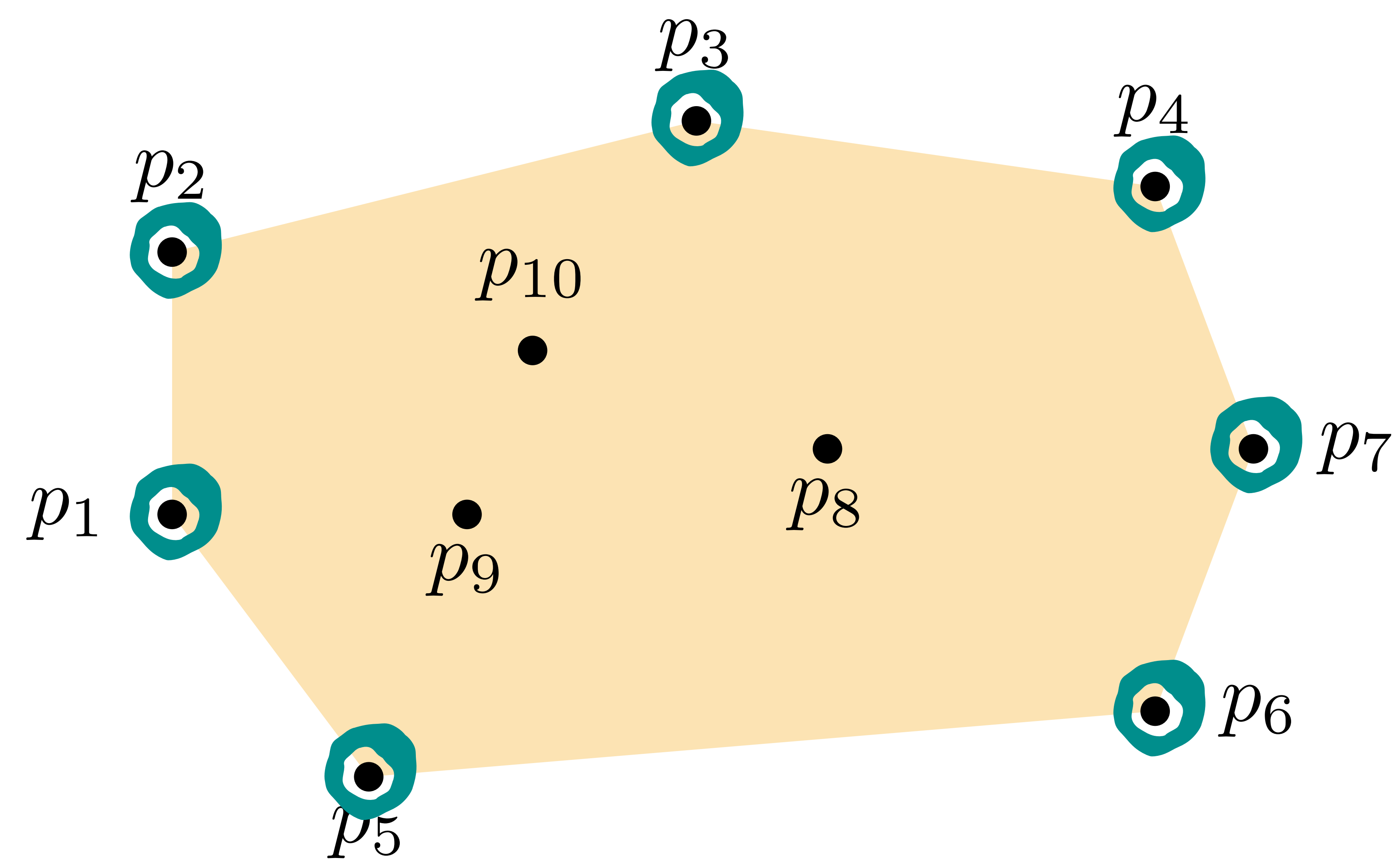
- For a finite point set $\mathscr{P} := \{p_1, p_2, \ \dots, p_n\} \subset \mathbb{R}^2$, $\mathrm{conv}(\mathscr{P}) = \ \dots$?
- *Definition 2.3:* $\mathrm{conv}(\mathscr{P}) = \{x \in \mathbb{R}^2 \mid x$ is a convex combination of $\mathscr{P}\}$.



The convex hull

… is a point set? A convex polygon? A subset?

- For a finite point set $\mathscr{P} := \{p_1, p_2, \ldots, p_n\} \subset \mathbb{R}^2$, $\mathrm{conv}(\mathscr{P}) = \ldots$?
- *Definition 2.3:* $\mathrm{conv}(\mathscr{P}) = \{x \in \mathbb{R}^2 \mid x$ is a convex combination of $\mathscr{P}\}$.
- *This is an infinite set, but our algorithm(s) only give us points from $\mathscr{P}$!?*

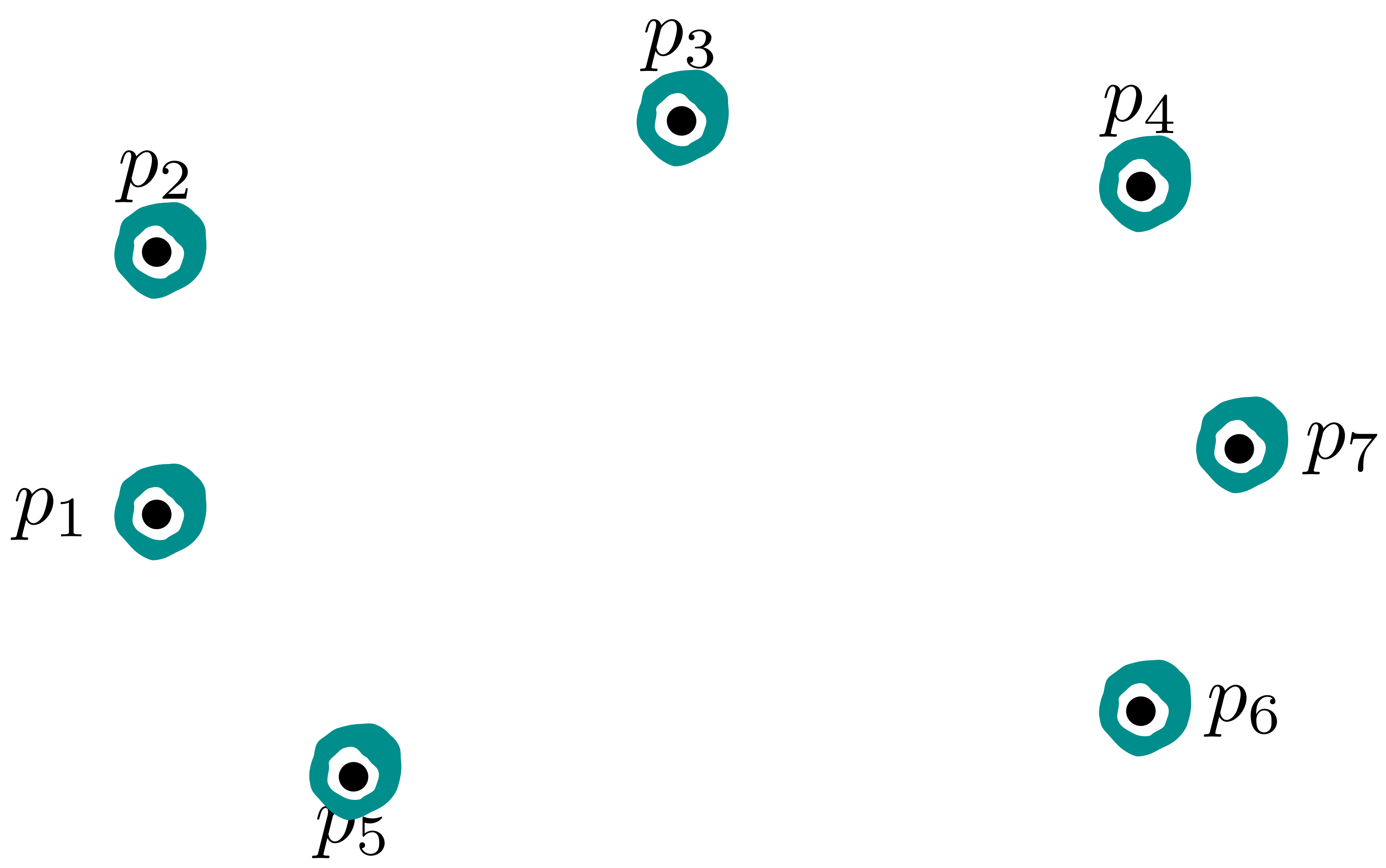


The convex hull

… is a point set? A convex polygon? A subset?

- For a finite point set $\mathscr{P} := \{p_1, p_2, \ldots, p_n\} \subset \mathbb{R}^2$, $\mathrm{conv}(\mathscr{P}) = \ldots$?
- *Definition 2.3:* $\mathrm{conv}(\mathscr{P}) = \{x \in \mathbb{R}^2 \mid x$ is a convex combination of $\mathscr{P}\}$.
- *This is an infinite set, but our algorithm(s) only give us points from $\mathscr{P}$!?*
- $\mathrm{conv}(\mathscr{P}) = \mathscr{P}$ *???*

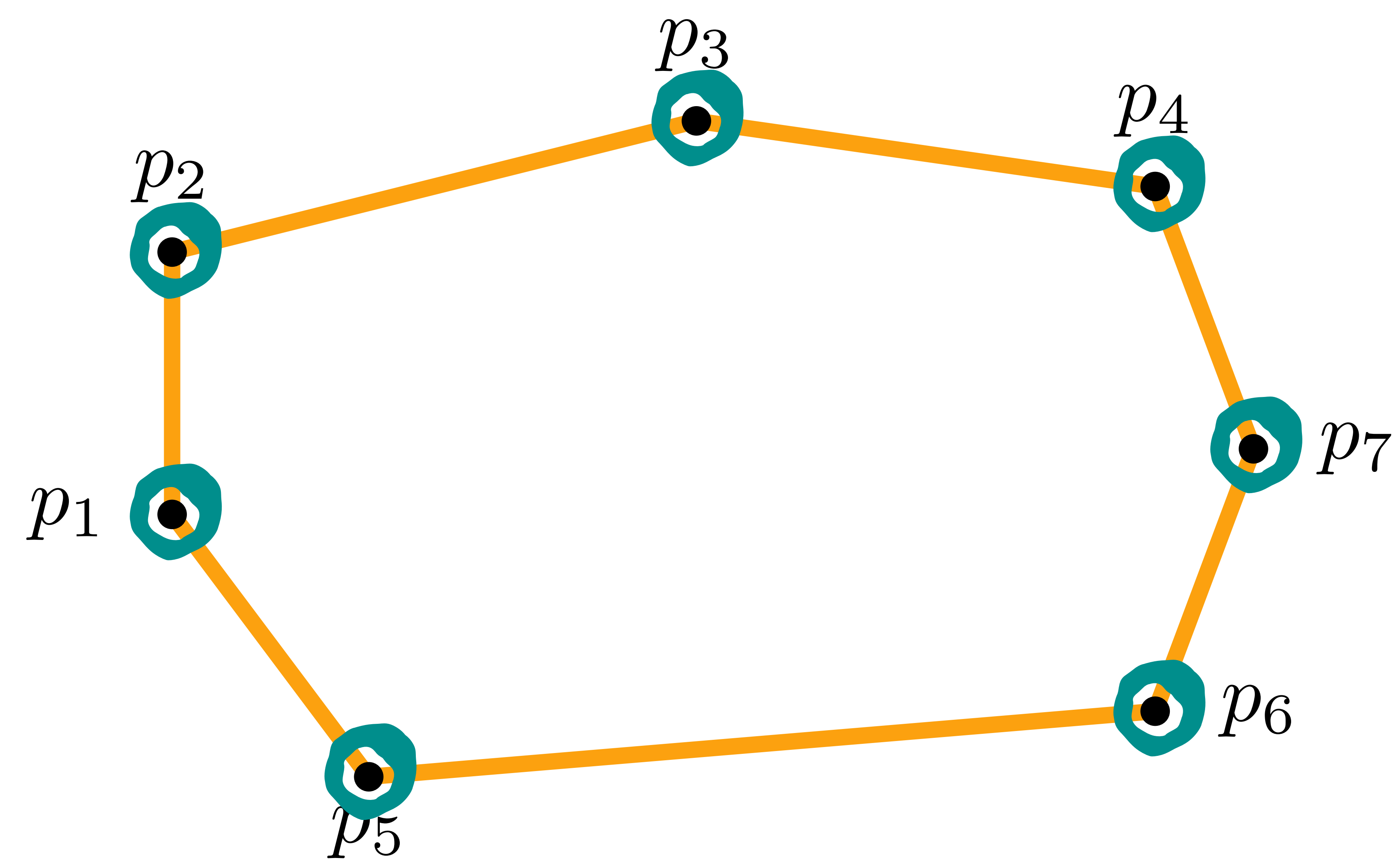
Point sets like this one are referred to as being in "convex position".



The convex hull

… is a point set? A convex polygon? A subset?

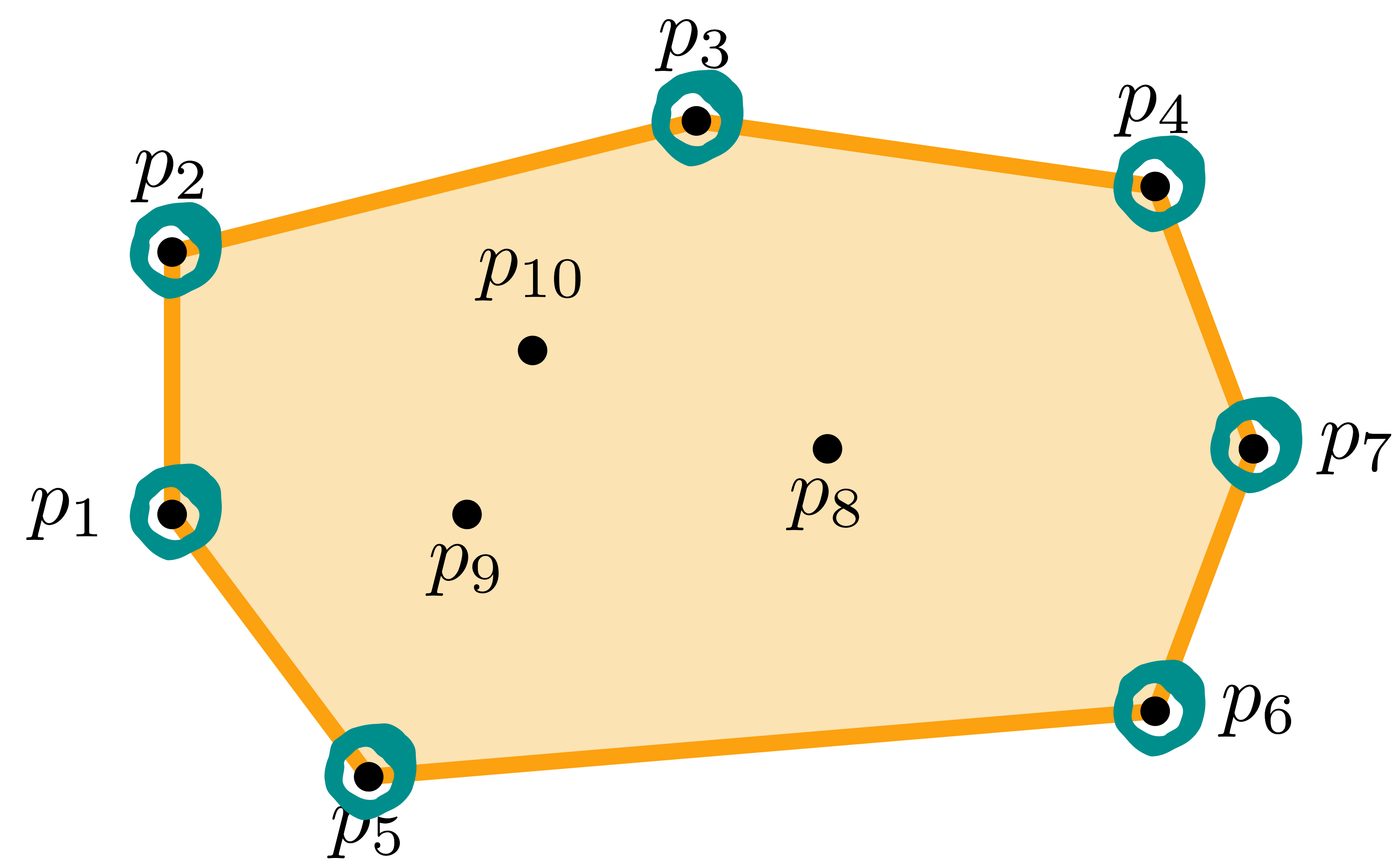
- For a finite point set $\mathscr{P} := \{p_1, p_2, \ \ldots, p_n\} \subset \mathbb{R}^2$, $\text{conv}(\mathscr{P}) = \ \ldots$?
- *Definition 2.3:* $\text{conv}(\mathscr{P}) = \{x \in \mathbb{R}^2 \mid x$ is a convex combination of $\mathscr{P}\}$.
- *This is an infinite set, but our algorithm(s) only give us points from $\mathscr{P}$!?*
- ~~$\text{conv}(\mathscr{P}) = \mathscr{P}$~~ *???*



The convex hull

… is a (usually infinite) point set, described by a convex polygon.

- Given a finite point set $\mathscr{P} := \{p_1, p_2, \ldots, p_n\} \subset \mathbb{R}^2$.
- *Definition 2.3:* $\text{conv}(\mathscr{P}) = \{x \in \mathbb{R}^2 \mid x$ is a convex combination of $\mathscr{P}\}$
- *Our algorithms compute the boundary set* $\delta\,\text{conv}(\mathscr{P})$ *as a convex polygon!*



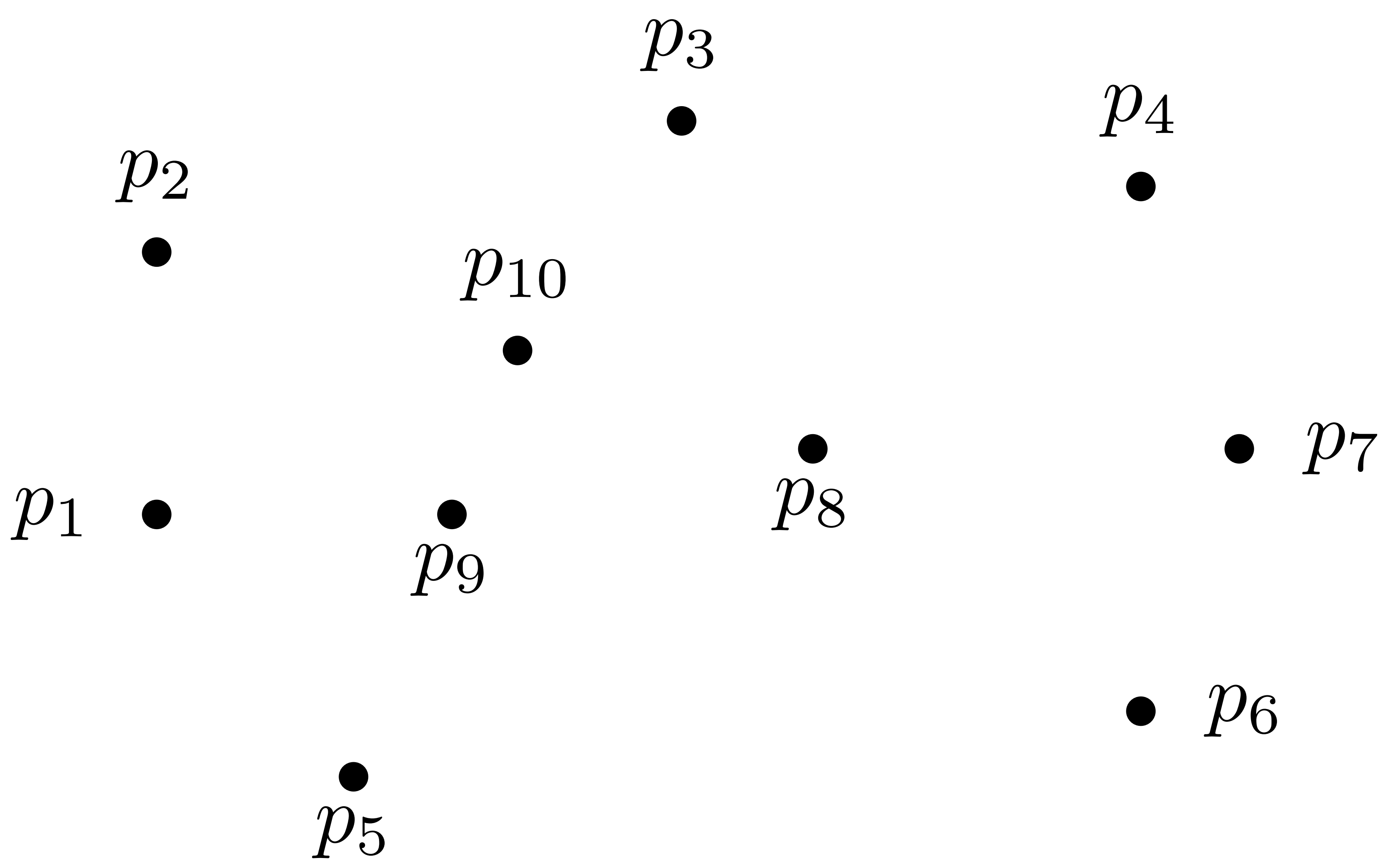
Detour: Point sets & Polygons

Peter Kramer | October 24th, 2024

Point sets & Polygons

What's the difference?

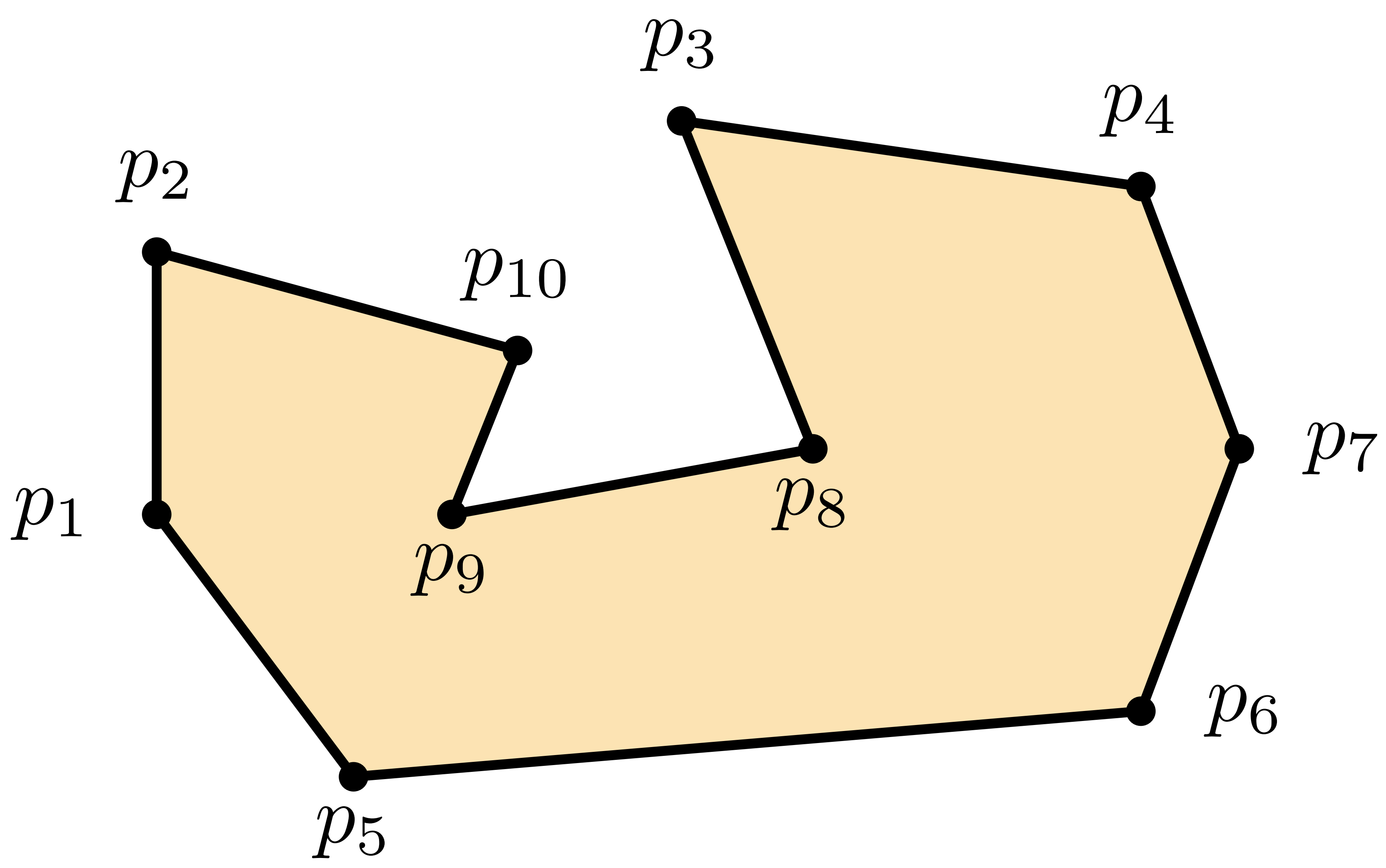
- Consider a finite point set $\mathscr{P} := \{p_1, p_2, \ldots, p_n\} \subset \mathbb{R}^2$.
- What's a polygon P using these points?



Point sets & Polygons

What's the difference?

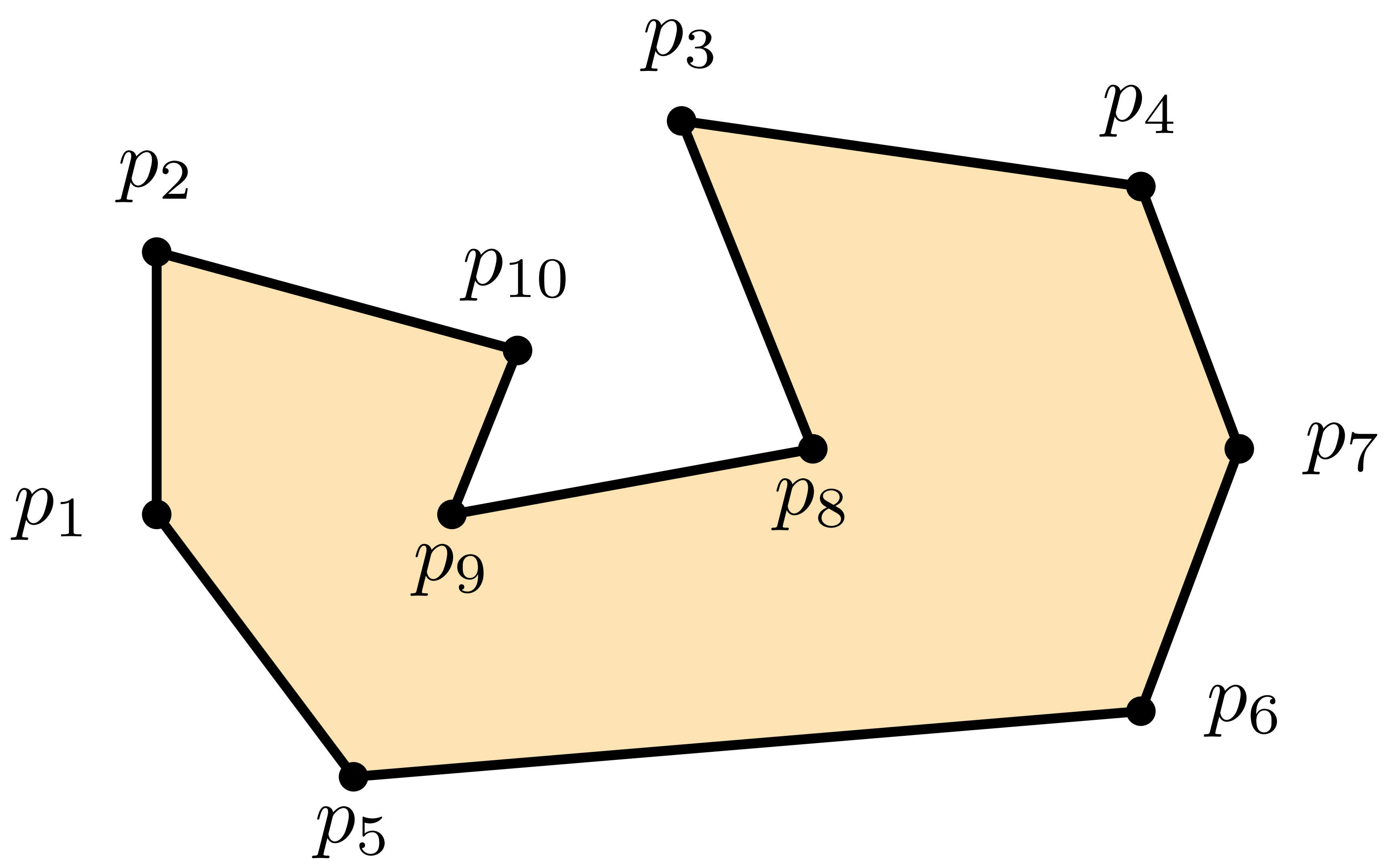
- Consider a finite point set $\mathscr{P} := \{p_1, p_2, \ldots, p_n\} \subset \mathbb{R}^2$.
- What's a polygon P using these points?



Point sets & Polygons

What's the difference?

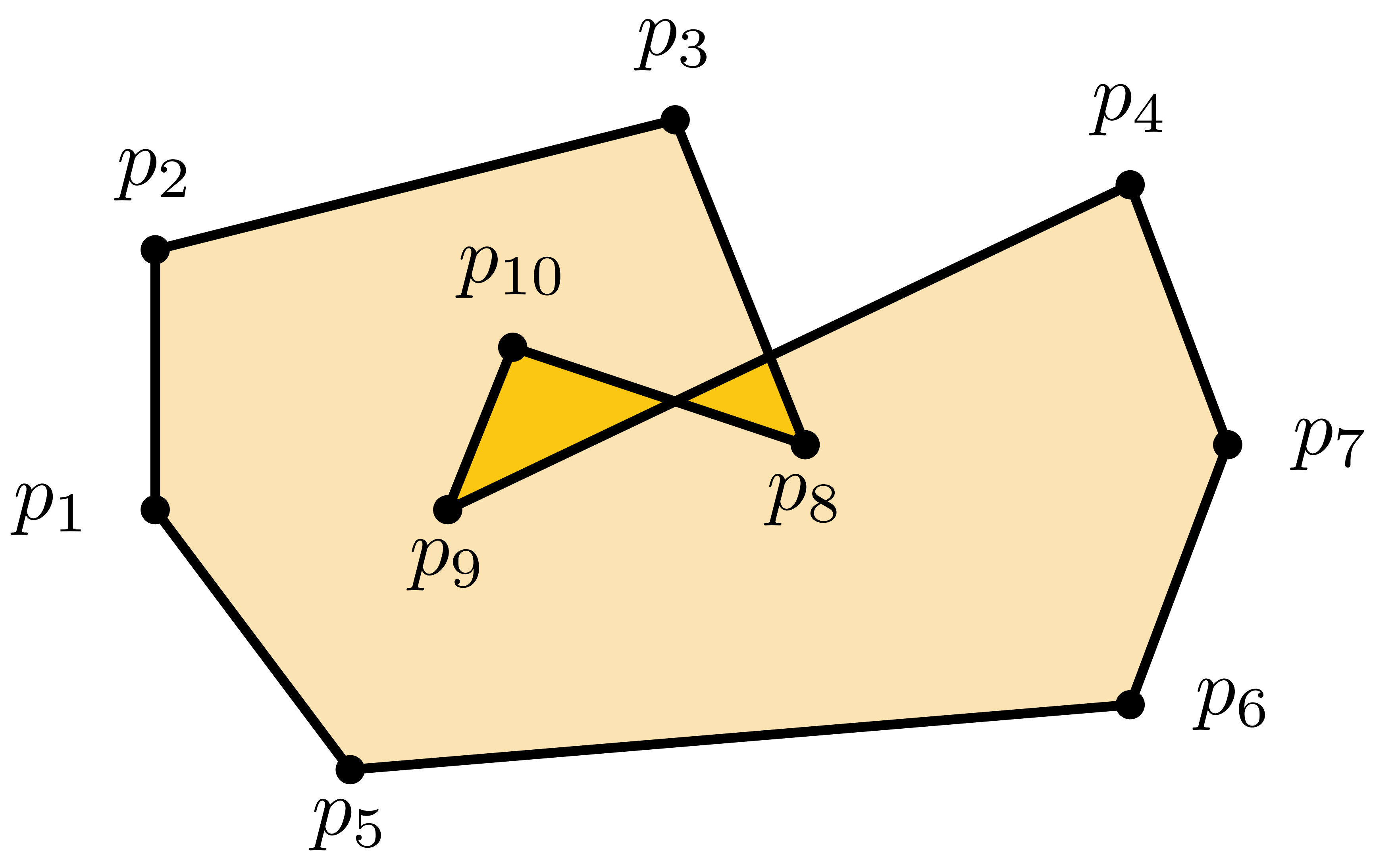
- Consider a finite point set $\mathscr{P} := \{p_1, p_2, \ldots, p_n\} \subset \mathbb{R}^2$.
- An ordered sequence $P := p_{\pi(1)}, p_{\pi(2)}, \ldots, p_{\pi(n)} \in (\mathbb{R} \times \mathbb{R})^n$ is a polygon.



Point sets & Polygons

What's the difference?

- Consider a finite point set $\mathscr{P} := \{p_1, p_2, \ldots, p_n\} \subset \mathbb{R}^2$.
- An ordered sequence $P := p_{\pi(1)}, p_{\pi(2)}, \ldots, p_{\pi(n)} \in (\mathbb{R} \times \mathbb{R})^n$ is a polygon.



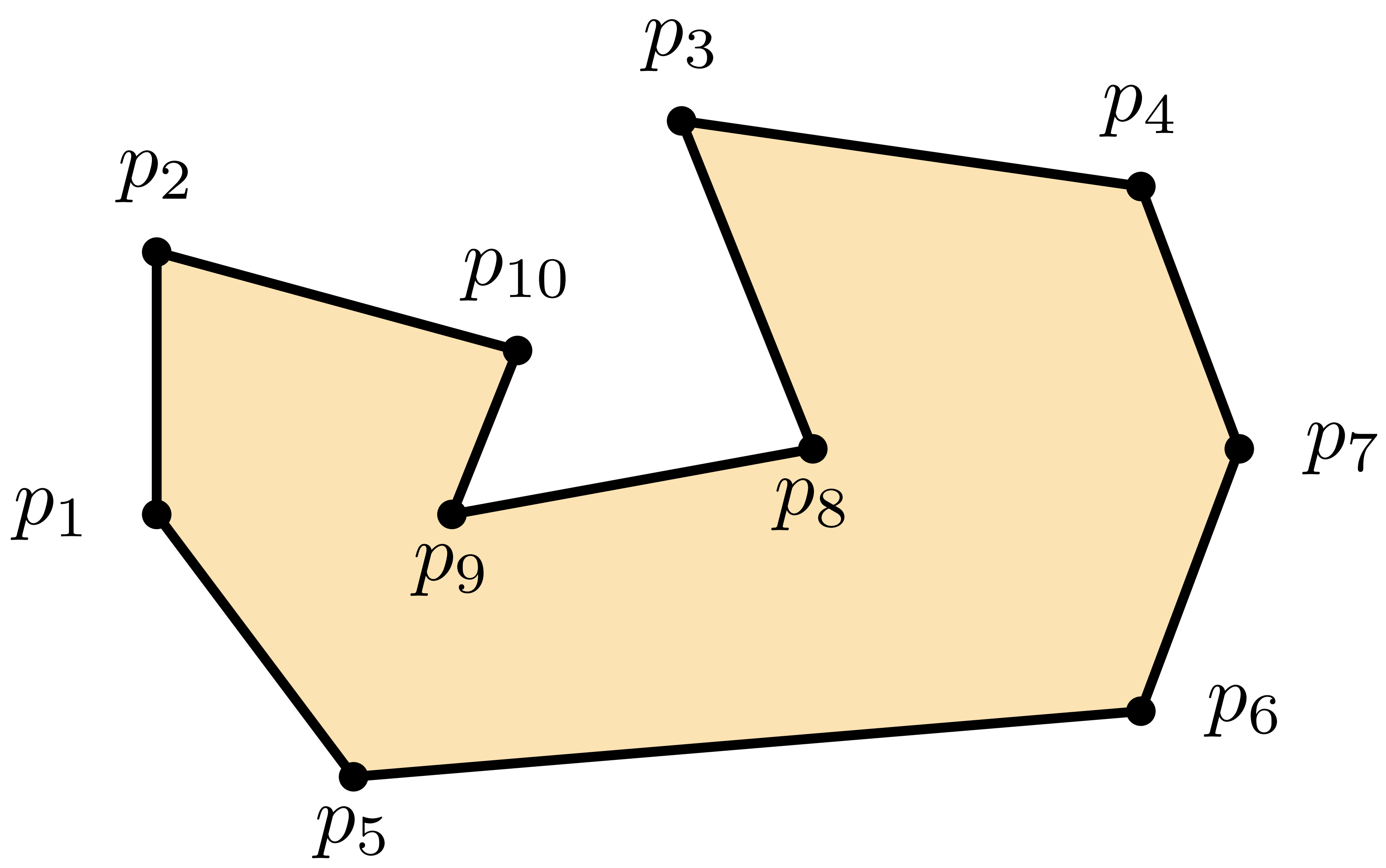
Point sets & Polygons

What's the difference?

- Consider a finite point set $\mathscr{P} := \{p_1, p_2, \ldots, p_n\} \subset \mathbb{R}^2$.
- A *simple* polygon P does not self-intersect.

Note: We will encounter further types of polygons later on :)

Typically, we describe polygons in CCW order:
$(p_1, p_5, p_6, p_7, p_4, p_3, p_8, p_9, p_10, p_2)$



Homework Sheet #1

Peter Kramer | October 24th, 2024

Computational Geometry – Sheet 1 **Winter 2024/2025**

Prof. Dr. Sándor P. Fekete **Due** 14.11.2024

Peter Kramer **Discussion** 21.11.2024

Please submit your handwritten answers in pairs, *using the box in front of IZ338* before *the exercise timeslot on the due date above. Make sure to include your* full names, matriculation numbers, *and the* programmes *that you are enrolled in.*

*In accordance with the **guidelines** of the TU Braunschweig, using AI tools to solve any part of the exercises is **not permitted**.*

Exercise 1. **(5 points)**

Refer to *Definitions 2.2 and 2.3* from the lecture. Prove or disprove: The intersection of two convex hulls according to *Definition 2.3* is itself convex for all finite sets $\mathcal{P}, \mathcal{Q} \subset \mathbb{R}^2$, i.e.,

$$\text{conv}(\text{conv}(\mathcal{P}) \cap \text{conv}(\mathcal{Q})) = \text{conv}(\mathcal{P}) \cap \text{conv}(\mathcal{Q}).$$

(Hint: Review the properties of convex hulls and combinations carefully.)

Definition E1 (General position). When investigating a problem, we often assume that the input data obeys *general position*, which excludes specific edges cases and simplifies analysis. For this sheet, a point set obeys general position if and only if no three points in it are collinear.

Exercise 2 (Point in Convex Polygon Problem). **(15 points)**

Using only the known predicates, design an $\mathcal{O}(\log n)$ algorithm for the following problem. Given a convex polygon $P = (p_1, p_2, \ldots, p_n) \in (\mathbb{R} \times \mathbb{R})^n$ and a point $q \in \mathbb{R}^2$, does P contain q? You may assume that the point set $\{p_1, p_2, \ldots, p_n, q\}$ obeys general position.

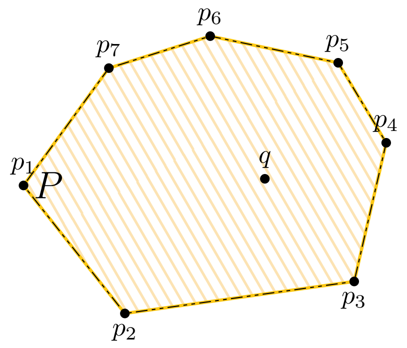


Figure 1: Example: the point q is located inside the convex polygon P.

Exercise 3 (Farthest Point Pairs). **(5+10 points)**

Given a set $\mathcal{P}$ of n points in the Euclidean plane, two points $p, q \in \mathcal{P}$ are a *farthest pair* in $\mathcal{P}$ if

$$\forall\, u, v \in \mathcal{P}: \quad |p - q| \geq |u - v|.$$

The Euclidean distance between p and q is then also called the *diameter* of $\mathcal{P}$.

a) Prove that all farthest pairs in $\mathcal{P}$ are vertices of the convex hull $\text{conv}(\mathcal{P})$.

b) Design an $\mathcal{O}(n)$ algorithm that approximates the diameter of $\mathcal{P}$ up to a constant factor. Briefly explain approximation factor and runtime!

1/1

Three weeks!

Computational Geometry – Sheet 1 **Winter 2024/2025**

Prof. Dr. Sándor P. Fekete **Due** 14.11.2024

Peter Kramer **Discussion** 21.11.2024

Please submit your handwritten answers in pairs, *using the box in front of IZ338* before *the exercise timeslot on the due date above. Make sure to include your* full names, matriculation numbers, *and the* programmes *that you are enrolled in.*

*In accordance with the **guidelines** of the TU Braunschweig, using AI tools to solve any part of the exercises is **not permitted**.*

These sheets are about logical deduction, so LLMs will most likely just generate plausible-sounding garbage.

Please save us the headache :)

Translation services such as DeepL are allowed, of course.
If you so prefer, you may write your solutions in german.

Prof. Dr. Sándor P. Fekete
Peter Kramer

Due 14.11.2024
Discussion 21.11.2024

Please submit your handwritten answers in pairs, using the box in front of `IZ338` *before the exercise timeslot on the due date above. Make sure to include your full names, matriculation numbers, and the programmes that you are enrolled in.*

In accordance with the `guidelines` *of the TU Braunschweig, using AI tools to solve any part of the exercises is* ***not permitted****.*

Exercise 1. **(5 points)**

Refer to *Definitions 2.2 and 2.3* from the lecture. Prove or disprove: The intersection of two convex hulls according to *Definition 2.3* is itself convex for all finite sets $\mathcal{P}, \mathcal{Q} \subset \mathbb{R}^2$, i.e.,

$$\operatorname{conv}(\operatorname{conv}(\mathcal{P}) \cap \operatorname{conv}(\mathcal{Q})) = \operatorname{conv}(\mathcal{P}) \cap \operatorname{conv}(\mathcal{Q}).$$

(Hint: Review the properties of convex hulls and combinations carefully.)

Definition E1 (General position). When investigating a problem, we often assume that the input data obeys *general position*, which excludes specific edges cases and simplifies analysis. For this sheet, a point set obeys general position if and only if no three points in it are collinear.

Exercise 2 (Point in Convex Polygon Problem). **(15 points)**

Using only the known predicates, design an $\mathcal{O}(\log n)$ algorithm for the following problem. Given a convex polygon $P = (p_1, p_2, \dots, p_n) \in (\mathbb{R} \times \mathbb{R})^n$ and a point $q \in \mathbb{R}^2$, does P contain q? You may assume that the point set $\{p_1, p_2, \dots, p_n, q\}$ obeys general position.

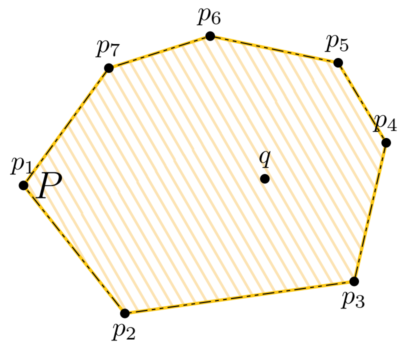


Figure 1: Example: the point q is located inside the convex polygon P.

Exercise 3 (Farthest Point Pairs). **(5+10 points)**

Given a set $\mathcal{P}$ of n points in the Euclidean plane, two points $p, q \in \mathcal{P}$ are a *farthest pair* in $\mathcal{P}$ if

$$\forall\, u, v \in \mathcal{P}: \quad |p - q| \geq |u - v|.$$

The Euclidean distance between p and q is then also called the *diameter* of $\mathcal{P}$.

a) Prove that all farthest pairs in $\mathcal{P}$ are vertices of the convex hull $\operatorname{conv}(\mathcal{P})$.

b) Design an $\mathcal{O}(n)$ algorithm that approximates the diameter of $\mathcal{P}$ up to a constant factor. Briefly explain approximation factor and runtime!

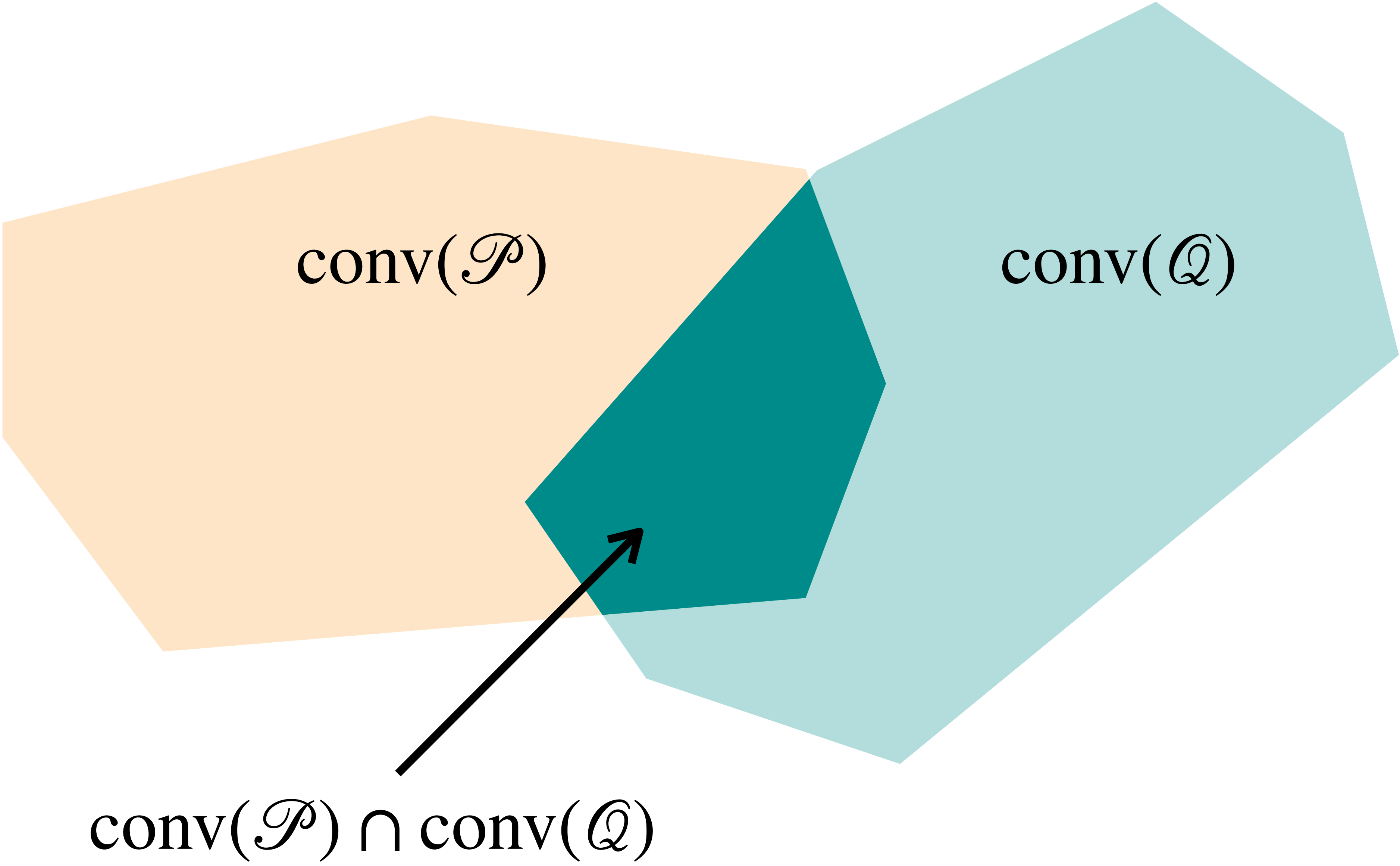
1/1

Exercise 1. **(5 points)**

Refer to *Definitions 2.2 and 2.3* from the lecture. Prove or disprove: The intersection of two convex hulls according to *Definition 2.3* is itself convex for all finite sets $\mathcal{P}, \mathcal{Q} \subset \mathbb{R}^2$, i.e.,

$$\operatorname{conv}(\operatorname{conv}(\mathcal{P}) \cap \operatorname{conv}(\mathcal{Q})) = \operatorname{conv}(\mathcal{P}) \cap \operatorname{conv}(\mathcal{Q}).$$

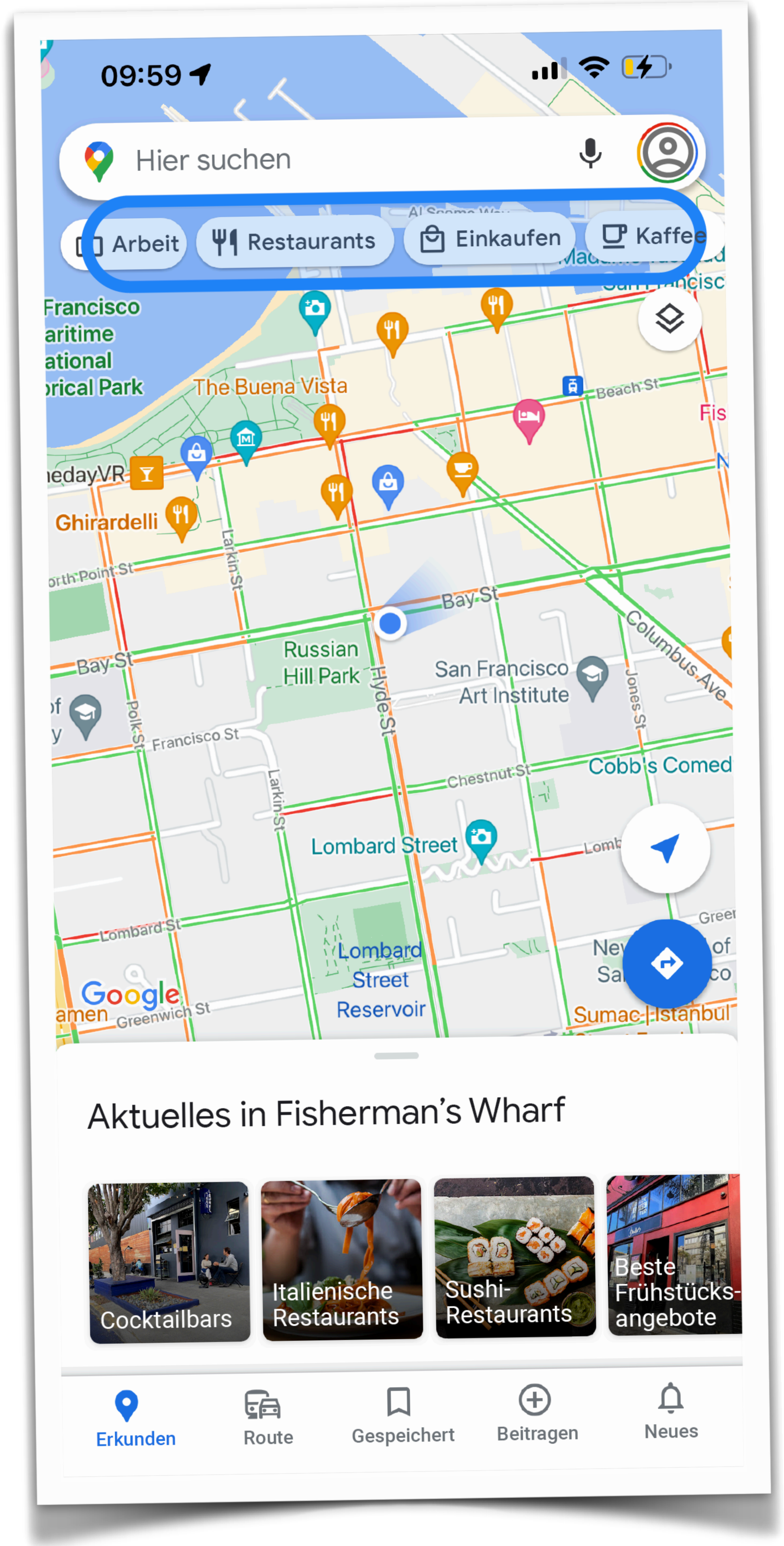
(Hint: Review the properties of convex hulls and combinations carefully.)



Point Location Problems

"Where am I?"

- Given geometric information such as a map in the plane, how can we decide **where** we are?



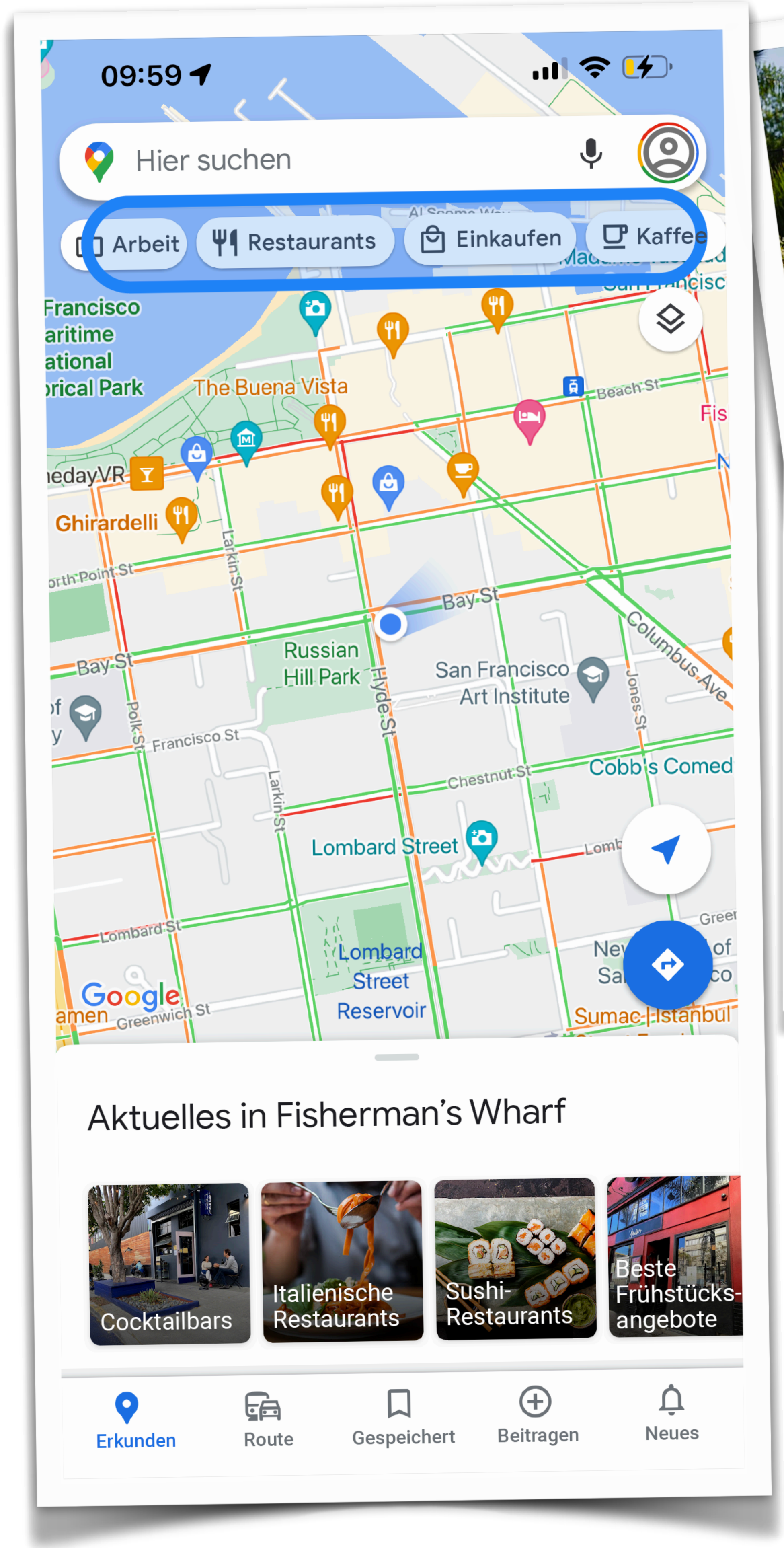
Point Location Problems

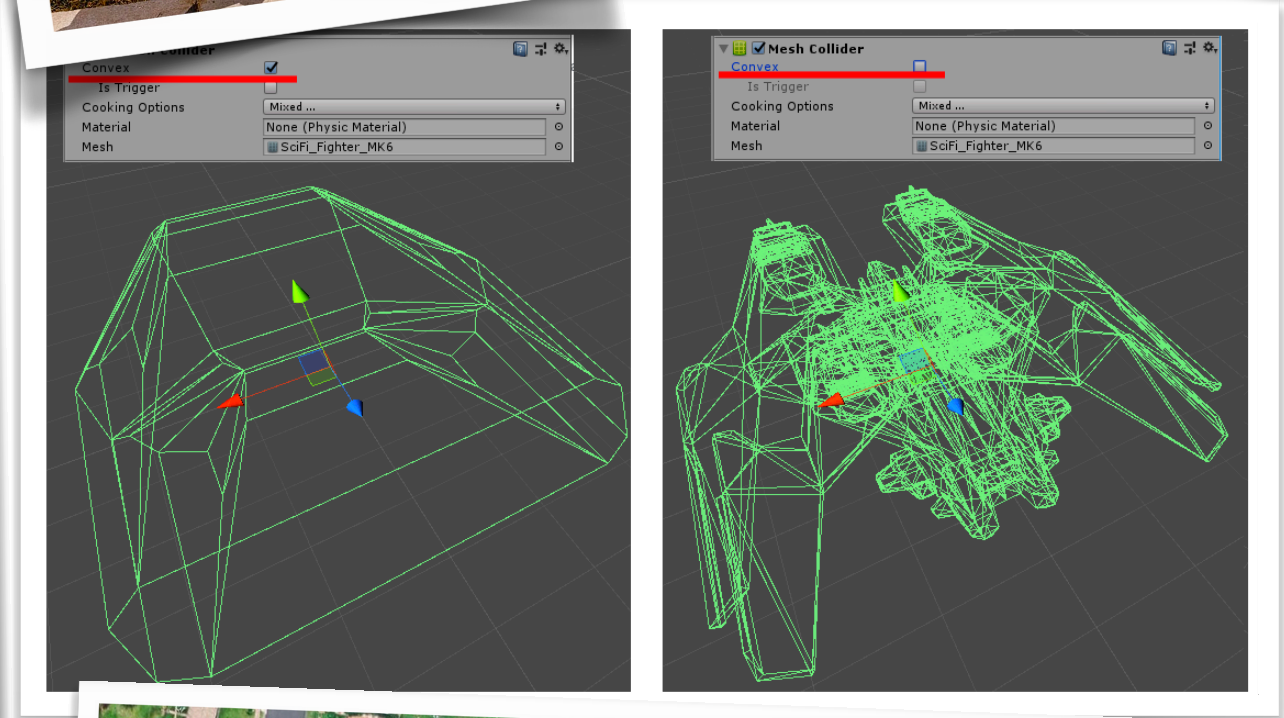
"Where am I?"

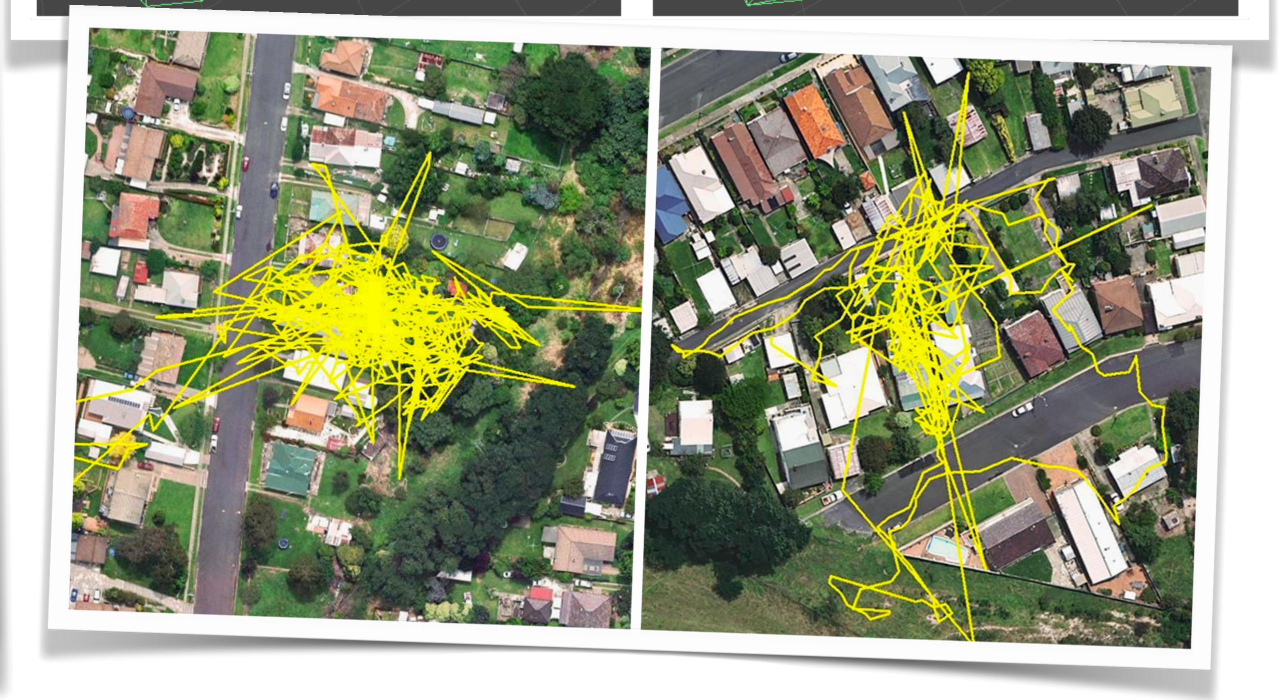
- Given geometric information such as a map in the plane, how can we decide **where** we are?

- *"In which country am I right now?"*
  "Can I leave this scooter here?"
  "Do these virtual objects collide?"

- ***"Is point p inside of region P?"***







Applications: Geofencing, Navigation, Simulation Software, Outlier Detection, …

Computational Geometry – Sheet 1 | **Winter 2024/2025**

Prof. Dr. Sándor P. Fekete
Peter Kramer

Due 14.11.2024
Discussion 21.11.2024

Please submit your handwritten answers in pairs, using the box in front of IZ338 *before the exercise timeslot on the due date above. Make sure to include your* full names, matriculation numbers, *and the programmes that you are enrolled in.*

In accordance with the **guidelines** *of the TU Braunschweig, using AI tools to solve any part of the exercises is* **not permitted**.

Exercise 1. **(5 points)**

Refer to *Definitions 2.2 and 2.3* from the lecture. Prove or disprove: The intersection of two convex hulls according to *Definition 2.3* is itself convex for all finite sets $\mathcal{P}, \mathcal{Q} \subset \mathbb{R}^2$, i.e.,

$$\operatorname{conv}(\operatorname{conv}(\mathcal{P}) \cap \operatorname{conv}(\mathcal{Q})) = \operatorname{conv}(\mathcal{P}) \cap \operatorname{conv}(\mathcal{Q}).$$

(Hint: Review the properties of convex hulls and combinations carefully.)

Definition E1 (General position). When investigating a problem, we often assume that the input data obeys *general position*, which excludes specific edges cases and simplifies analysis. For this sheet, a point set obeys general position if and only if no three points in it are collinear.

Exercise 2 (Point in Convex Polygon Problem). **(15 points)**

Using only the known predicates, design an $\mathcal{O}(\log n)$ algorithm for the following problem. Given a convex polygon $P = (p_1, p_2, \ldots, p_n) \in (\mathbb{R} \times \mathbb{R})^n$ and a point $q \in \mathbb{R}^2$, does P contain q? You may assume that the point set $\{p_1, p_2, \ldots, p_n, q\}$ obeys general position.

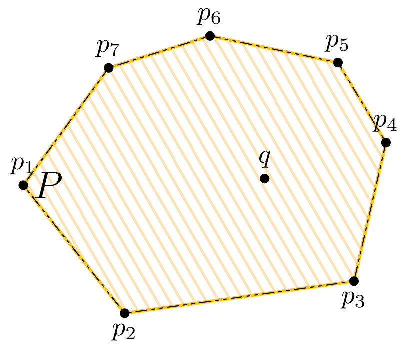


Figure 1: Example: the point q is located inside the convex polygon P.

Exercise 3 (Farthest Point Pairs). **(5+10 points)**

Given a set $\mathcal{P}$ of n points in the Euclidean plane, two points $p, q \in \mathcal{P}$ are a *farthest pair* in $\mathcal{P}$ if

$$\forall\, u, v \in \mathcal{P}: \quad |p - q| \geq |u - v|.$$

The Euclidean distance between p and q is then also called the *diameter* of $\mathcal{P}$.

a) Prove that all farthest pairs in $\mathcal{P}$ are vertices of the convex hull $\operatorname{conv}(\mathcal{P})$.

b) Design an $\mathcal{O}(n)$ algorithm that approximates the diameter of $\mathcal{P}$ up to a constant factor. Briefly explain approximation factor and runtime!

1/1

Definition E1 (General position). When investigating a problem, we often assume that the input data obeys *general position*, which excludes specific edges cases and simplifies analysis. For this sheet, a point set obeys general position if and only if no three points in it are collinear.

Exercise 2 (Point in Convex Polygon Problem). **(15 points)**

Using only the known predicates, design an $\mathcal{O}(\log n)$ algorithm for the following problem. Given a convex polygon $P = (p_1, p_2, \ldots, p_n) \in (\mathbb{R} \times \mathbb{R})^n$ and a point $q \in \mathbb{R}^2$, does P contain q? You may assume that the point set $\{p_1, p_2, \ldots, p_n, q\}$ obeys general position.

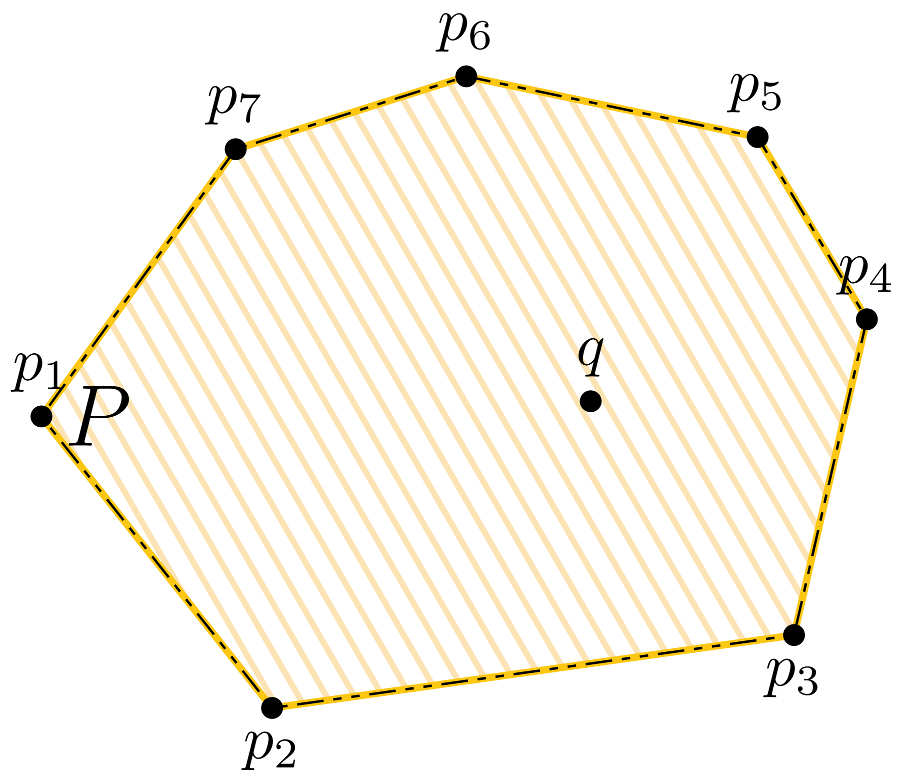


Figure 1: Example: the point q is located inside the convex polygon P.

Computational Geometry – Sheet 1 **Winter 2024/2025**
Prof. Dr. Sándor P. Fekete **Due** 14.11.2024
Peter Kramer **Discussion** 21.11.2024

Please submit your handwritten answers in pairs, using the box in front of `IZ338` *before the exercise timeslot on the due date above. Make sure to include your full names, matriculation numbers, and the programmes that you are enrolled in.*

In accordance with the `guidelines` *of the TU Braunschweig, using AI tools to solve any part of the exercises is* ***not permitted****.*

Exercise 1. **(5 points)**

Refer to *Definitions 2.2 and 2.3* from the lecture. Prove or disprove: The intersection of two convex hulls according to *Definition 2.3* is itself convex for all finite sets $\mathcal{P}, \mathcal{Q} \subset \mathbb{R}^2$, i.e.,

$$\operatorname{conv}(\operatorname{conv}(\mathcal{P}) \cap \operatorname{conv}(\mathcal{Q})) = \operatorname{conv}(\mathcal{P}) \cap \operatorname{conv}(\mathcal{Q}).$$

(Hint: Review the properties of convex hulls and combinations carefully.)

Definition E1 (General position). When investigating a problem, we often assume that the input data obeys *general position*, which excludes specific edges cases and simplifies analysis. For this sheet, a point set obeys general position if and only if no three points in it are collinear.

Exercise 2 (Point in Convex Polygon Problem). **(15 points)**

Using only the known predicates, design an $\mathcal{O}(\log n)$ algorithm for the following problem. Given a convex polygon $P = (p_1, p_2, \dots, p_n) \in (\mathbb{R} \times \mathbb{R})^n$ and a point $q \in \mathbb{R}^2$, does P contain q? You may assume that the point set $\{p_1, p_2, \dots, p_n, q\}$ obeys general position.

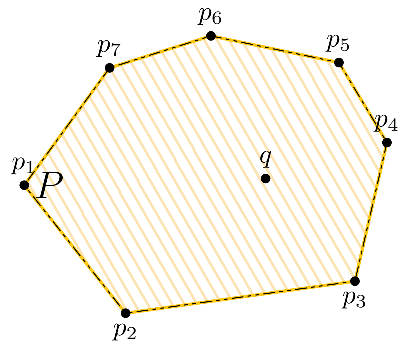


Figure 1: Example: the point q is located inside the convex polygon P.

Exercise 3 (Farthest Point Pairs). **(5+10 points)**

Given a set $\mathcal{P}$ of n points in the Euclidean plane, two points $p, q \in \mathcal{P}$ are a *farthest pair* in $\mathcal{P}$ if

$$\forall\, u, v \in \mathcal{P}: \quad |p - q| \ \geq\ |u - v|.$$

The Euclidean distance between p and q is then also called the *diameter* of $\mathcal{P}$.

a) Prove that all farthest pairs in $\mathcal{P}$ are vertices of the convex hull $\operatorname{conv}(\mathcal{P})$.

b) Design an $\mathcal{O}(n)$ algorithm that approximates the diameter of $\mathcal{P}$ up to a constant factor. Briefly explain approximation factor and runtime!

1/1

Exercise 3 (Farthest Point Pairs). **(5+10 points)**

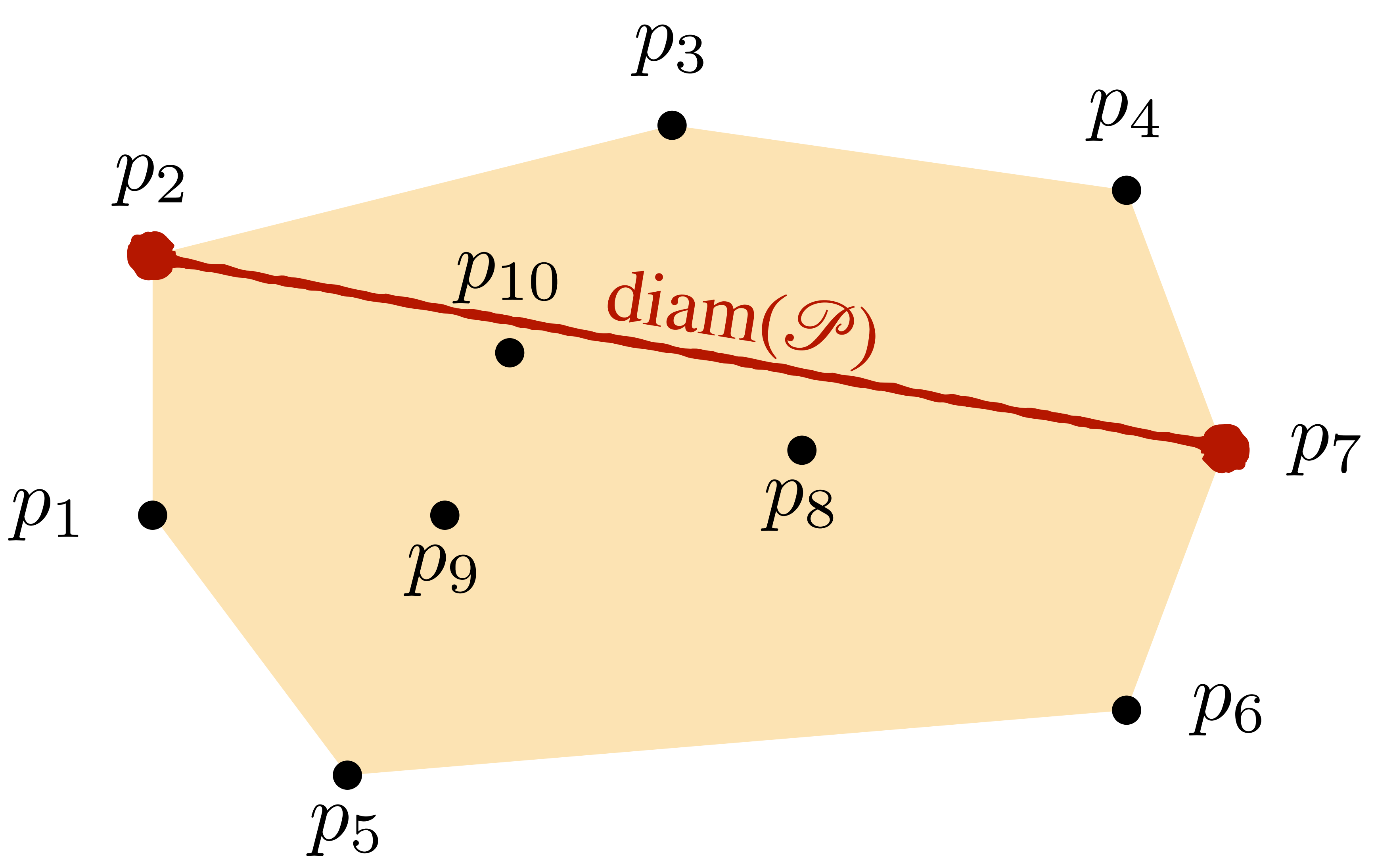
Given a set $\mathcal{P}$ of n points in the Euclidean plane, two points $p, q \in \mathcal{P}$ are a *farthest pair* in $\mathcal{P}$ if

$$\forall\, u, v \in \mathcal{P}: \quad |p - q| \ \geq\ |u - v|.$$

The Euclidean distance between p and q is then also called the *diameter* of $\mathcal{P}$.

a) Prove that all farthest pairs in $\mathcal{P}$ are vertices of the convex hull $\operatorname{conv}(\mathcal{P})$.

b) Design an $\mathcal{O}(n)$ algorithm that approximates the diameter of $\mathcal{P}$ up to a constant factor. Briefly explain approximation factor and runtime!



For a c-approximation, argue that your algorithm's output is

- at most $c \cdot \operatorname{diam}(\mathscr{P})$, and
- at least $\frac{1}{c} \cdot \operatorname{diam}(\mathscr{P})$.

TL;DL

The important parts

- **Sign up for the mailing list!**
- One tutorial a week (except holidays).
- Biweekly homework: **Find partners!**
- First **sheet is out now**, due in 3 weeks.
- Material can be found on the website.
  - Slides, videos, and homework sheets.

- Any questions? Ask Peter or Kai:
  - kramer@ibr.cs.tu-bs.de
  - kobbe@ibr.cs.tu-bs.de

October						
28	29	30	31			

November						
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

Extra slides

Big O Notation

Catching up, if necessary.

- If you speak German: There is plenty of material on the course website of “Algorithmen und Datenstrukturen”

Institut für Betriebssysteme
und Rechnerverbund
Algorithmik

- If you don’t: There is plenty of other courseware (e.g., from MIT) available!

MIT OpenCourseWare

- There also exist a variety of short-form explanation videos, such as: