

## 2.7 Geometrische Algorithmen

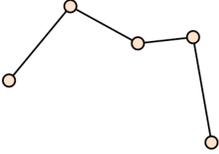
- 2.7.1 Inside-Test
- 2.7.2 Konvexe Hülle
- 2.7.3 Nachbarschaften
- 2.7.4 Schnittprobleme

1  **Datenstrukturen und Algorithmen**  
Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer 

Viele Probleme aus der Praxis lassen sich geometrisch formulieren. In diesem Kapitel werden wir ein paar Beispielprobleme betrachten. Auch Probleme in hochdimensionalen Räumen sind geometrische Probleme. Hier werden wir allerdings nur 2D und 3D Probleme betrachten.

## Polygone

- Sequenz von Knoten/Eckpunkten  
 $p_1, \dots, p_n$
- Sequenz von Kanten  
 $e_i = (p_i, p_{i+1})$



2  **Datenstrukturen und Algorithmen**  
Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer 

Ist ein Polygon offen oder geschlossen? Dieses hier ist offen...

### Polygone

- Eigenschaften
  - offen/geschlossen

$p_1 = p_n$

Datenstrukturen und Algorithmen  
Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer

... und dieses ist geschlossen.

### Polygone

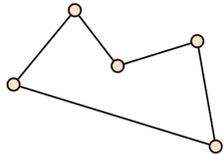
- Eigenschaften
  - offen/geschlossen
  - einfach/komplex

Datenstrukturen und Algorithmen  
Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer

Komplexes Polygon = Polygon mit Selbstüberschneidung. Wo ist außen, wo ist innen?  
Nicht eindeutig definiert.

### Polygone

- Eigenschaften
  - offen/geschlossen
  - einfach/komplex



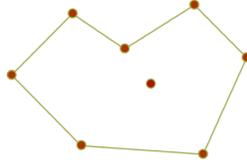
5

**Datenstrukturen und Algorithmen**  
 Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer

Problemstellungen immer der Art: Man hat eine Menge von Punkten/Polygonen und man will bestimmte Fragen dazu beantworten können.

### Inside-Test

- Liegt ein gegebener Punkt q innerhalb eines geschlossenen Polygons  $p_1, \dots, p_n$ ?



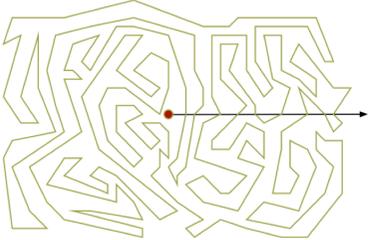
6

**Datenstrukturen und Algorithmen**  
 Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer

Dieses Problem ist algorithmisch genauso schwer zu beantworten, wie...

### Inside-Test

- Liegt ein gegebener Punkt  $q$  innerhalb eines geschlossenen Polygons  $p_1, \dots, p_n$ ?



The diagram shows a complex, irregular polygon drawn with thin green lines. A red dot representing a point  $q$  is located inside the polygon. A horizontal black arrow points to the right from the red dot, representing a ray used for the inside-test algorithm.

7  Datenstrukturen und Algorithmen  
Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer  FRIEDRICH-ALEXANDER  
UNIVERSITÄT

...dieses. Man verfolgt einen Strahl ausgehend von dem Punkt nach außerhalb. Jedes Mal wenn der Strahl das Polygon schneidet, wechselt der Status zwischen „innerhalb“ und „außerhalb“.

### Inside-Test

- Liegt ein gegebener Punkt  $q$  innerhalb eines geschlossenen Polygons  $p_1, \dots, p_n$ ?

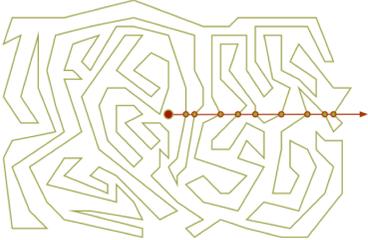


The diagram shows the same complex polygon as in slide 7, but drawn with thick orange lines. A red dot representing a point  $q$  is located inside the polygon. A horizontal black arrow points to the right from the red dot, representing a ray used for the inside-test algorithm.

8  Datenstrukturen und Algorithmen  
Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer  FRIEDRICH-ALEXANDER  
UNIVERSITÄT

### Inside-Test

- Liegt ein gegebener Punkt  $q$  innerhalb eines geschlossenen Polygons  $p_1, \dots, p_n$ ?

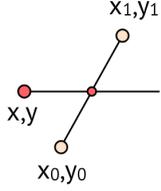



**Datenstrukturen und Algorithmen**  
 Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer  


Wir schießen eine Gerade vom Punkt aus ins Unendliche und zählen Schnittpunkte mit Polygon. Anzahl Schnitte gerade = Punkt außerhalb, Anzahl Schnitte ungerade = Punkt innerhalb.  
 Per Konvention „schießen“ wir den Strahl immer in positive x-Richtung. Die Richtung spielt aber (offenbar) keine Rolle.

### Inside-Test

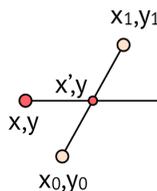
- $\text{Inside}(x, y, x[1..n], y[1..n])$   
 $k \leftarrow 0$   
**for**  $i \leftarrow 1$  **to**  $n$  **do**  
   **if**  $y[i] < y[i+1]$  **then**  
      $x_0 \leftarrow x[i]; y_0 \leftarrow y[i]$   
      $x_1 \leftarrow x[i+1]; y_1 \leftarrow y[i+1]$   
   **else**  
      $x_1 \leftarrow x[i]; y_1 \leftarrow y[i]$   
      $x_0 \leftarrow x[i+1]; y_0 \leftarrow y[i+1]$   
   **if**  $y_0 \leq y$  **and**  $y < y_1$  **then**  
     **if**  $(x-x_0) \times (y_1-y_0) < (y-y_0) \times (x_1-x_0)$  **then**  
        $k \leftarrow k + 1$   
   **return**  $\text{odd}(k)$




**Datenstrukturen und Algorithmen**  
 Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer  


### Schnittberechnung

- $(y-y_0) / (y_1-y_0) = (x'-x_0) / (x_1-x_0)$
- $(y-y_0) \times (x_1-x_0) = (x'-x_0) \times (y_1-y_0)$
- $x < x'$
- $x-x_0 < x'-x_0$
- $(x-x_0) \times (y_1-y_0) < (x'-x_0) \times (y_1-y_0)$
- $(x-x_0) \times (y_1-y_0) < (y-y_0) \times (x_1-x_0)$



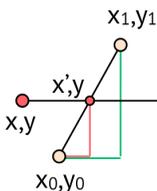
11

**Datenstrukturen und Algorithmen**  
 Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer

Hier wird das Kreuzprodukt zweier Vektoren ausgewertet. Dies kann durch die „Rechte-Hand-Regel“ veranschaulicht werden (siehe z.B. Wikipedia, etc.).

### Schnittberechnung

- $(y-y_0) / (y_1-y_0) = (x'-x_0) / (x_1-x_0)$
- $(y-y_0) \times (x_1-x_0) = (x'-x_0) \times (y_1-y_0)$
- $x < x'$
- $x-x_0 < x'-x_0$
- $(x-x_0) \times (y_1-y_0) < (x'-x_0) \times (y_1-y_0)$
- $(x-x_0) \times (y_1-y_0) < (y-y_0) \times (x_1-x_0)$

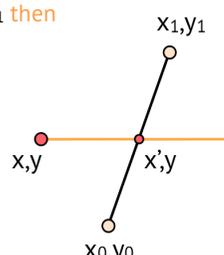


12

**Datenstrukturen und Algorithmen**  
 Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer

**Spezialfälle**

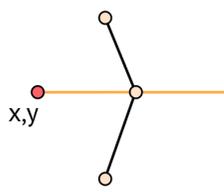
- Inside(...)
- ...
- if  $y_0 \leq y$  and  $y < y_1$  then
- ...



13  **Datenstrukturen und Algorithmen**  
Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer  FRIEDRICH-SCHILLER-UNIVERSITÄT

**Spezialfälle**

- Inside(...)
- ...
- if  $y_0 \leq y$  and  $y < y_1$  then
- ...



14  **Datenstrukturen und Algorithmen**  
Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer  FRIEDRICH-SCHILLER-UNIVERSITÄT

Hier ist das Bestimmen des Schnittpunkts etwas schwieriger, weil die Gerade genau durch den Punkt geht. Zählt der Schnittpunkt zur oberen oder zur unteren Kante? Dies muss per Konvention bestimmt werden. Dadurch, dass im ersten Fall auf „kleiner-gleich“ und im zweiten Falle auf „echt kleiner“ getestet wird, ist diese Entscheidung wieder eindeutig.

**Spezialfälle**

- Inside(...)
- ...
- if  $y_0 \leq y$  and  $y < y_1$  then
- ...

x,y

15 **Datenstrukturen und Algorithmen**  
 Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer  RWTH AACHEN UNIVERSITY

**Spezialfälle**

- Inside(...)
- ...
- if  $y_0 \leq y$  and  $y < y_1$  then
- ...

x,y

16 **Datenstrukturen und Algorithmen**  
 Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer  RWTH AACHEN UNIVERSITY

**Spezialfälle**

- Inside(...)
- ...
- if  $y_0 \leq y$  and  $y < y_1$  then
- ...

x,y

17 **Datenstrukturen und Algorithmen**  
 Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer

**Spezialfälle**

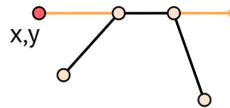
- Inside(...)
- ...
- if  $y_0 \leq y$  and  $y < y_1$  then
- ...

x,y

18 **Datenstrukturen und Algorithmen**  
 Prof. Dr. Leif Kobbelt, Thomas Stroder, Fabian Emmes, Sven Middelberg, Michael Kremer

## Spezialfälle

- Inside(...)  
...  
if  $y_0 \leq y$  and  $y < y_1$  then  
...  
...



## Analyse

- Der Aufwand des Inside-Tests ist  $O(n)$ .

