Übungsblatt 3 - Polygontriangulierung

Ausgabe: 29. April 2014
Abgabe: 06. Mai 2014

1 Korrektheitsbeweis

In der Vorlesung wurde der Algortihmus **MakeMonotone** vorgestellt. Im Verlauf der Ausführung des Algorithmus wird die Funktion **handleMergeVertex** augefrufen. Beweisen Sie die Korrektheit der Funktion.
Hinweis: Für Details über **MakeMonotone** und die verwendeten Datenstrukturen siehe Vorlesungsfolien.

Algorithmus 1 : handleMergeVertex

$e \leftarrow$ rechte Kante
if *isMergeVertex(helper(e))* **then**
  $\mathcal{D} \leftarrow$ füge $\overline{\text{helper}(e)v}$ ein
lösche e aus $\mathcal{T}$
$e' \leftarrow$ Kante links von v in $\mathcal{T}$
if *isMergeVertex(helper(e'))* **then**
  $\mathcal{D} \leftarrow$ füge $\overline{\text{helper}(e')v}$ ein
helper$(e') \leftarrow v$

2 Lineare Laufzeit?

Der in der Vorlesung vorgestellte Algorithmus zur Partitionierung eines Polygons in y-monotone Teilpolygone hat eine worst-case Laufzeit von $\mathcal{O}(n \log n)$. Modifizieren Sie den Algorithmus so, dass die worst-case Laufzeit auf $\mathcal{O}(n)$ sinkt, unter der Annahme, dass die Anzahl der Wendeknoten in den Eingabe-Polygonen in $\mathcal{O}(1)$ ist.

Bitte umblättern

3 Das Kunstgalerie-Problem – Randabdeckung

Beweisen, oder widerlegen Sie folgende Aussage:
Sei $\mathcal{P}$ ein einfaches Polygon. Positioniert man Kameras so, dass alle Kameras zusammengenommen den vollständigen Rand von $\mathcal{P}$ abdecken, dann decken sie auch automatisch das gesamte Innere von $\mathcal{P}$ ab.

4 Polygon-Splitting

Geben Sie einen $\mathcal{O}(n \log n)$ Algorithmus an, der ein einfaches Polygon, bestehend aus n Knoten, in zwei einfache Polygone auftrennt wobei jedes aus höchstens $\lfloor 2n/3 \rfloor + 2$ Knoten besteht.
Hinweis: Triangulieren Sie das Eingabe-Polygon und verwenden Sie dann den Dualgraphen dieser Triangulierung.