

Aufgabe 1 Siedler

Teilnehmer-ID: ???

Bearbeiter dieser Aufgabe:

Daniel Hohmann

April 5, 2024

Inhaltsverzeichnis

1 **Loesungsidee** **2**

2 **Umsetzung** **2**

3 **Beispiele** **2**

4 **Quellcode** **2**

1 Loesungsidee

Ich habe mir die Aufgabe durchgelesen, und dabei gemerkt, dass die Hauptaufgabe im Endefekt nichts anderes ist, wie die den Punkt im polygon zu finden, der wen ich mein radius von 85km platziere, die meisten Doerfer im abstand von 10km halten kann. Danach muss ich im endefekt nur noch die restlichen

Doerfer im abstand von 20km platzieren.

Meine Loesungsidee ist folgende, ich suche mir ein startpunkt der garantiert immer im Polygon liegt. Meine wahl ist da auf den Centroid (masseschwerpunkt eines Polygons) gefallen, da er fasst immer schon relativ central im Polygon inneren liegt.

als naechstes fange ich einfach an kreise zu ziehen, wobei ich mit einem kreis mit dem radius von 10 anfangen und dann immer in 10er schritten bis 85 hochgehe. Ich berechene einfach fuer jeden kreis die punkte die auf seinem umfang im abstand von 10km liegen, und schaue ob der jeweilige punkt auch im polygon liegt.

danach gehe ich fuer den jeweiligen kreisumfang die Punkte durch, und schaue wen ich jezt den punkt als neuen centroid des kreises benutze, kann ich dann nach den ebengenannten bedingungen mehr punkte platzieren die innerhalb des Polygons liegen. Das bedeutet aber auch, das ich wen ich ein als Bsp. ein radius von 20 habe, vorher auch die neuen punkte vom radius von 10 berechnen muss.

Um am Ende die Restlichen punkte innerhalb des Polygons zu platzieren mache ich einfach weiter kreise, aber erst ab dem radius von 85, und nicht mehr der erhoehung von 10, sondern von 20 das mache ich solange bis der kreis nicht mehr innerhalb des Polygons liegt oder es nicht mehr schneidet.

2 Umsetzung

2.1 Geometriesche funktionen klasse Point, Polygon

2.2 Der Hauptalgorithmus und seine funktion

2.3 Erweiterung der Plot des Polygons samt punkten

3 Beispiele

4 Quellcode

4.1 Geometrie.h

4.2 main.cpp