Übungsblatt 4

(11. Mai 2007)

Aufgabe 1 Scheiben Sie die Klasse `Star`, eine weitere Unterklasse von `GeometricObject`. Es sollen Sterne in dieser Klasse dargestellt werden. Ein Stern hat dabei einen äußeren Radius für die Spitzen der Strahlen und einen inneren Radius für die Zacken. Implementieren Sie für die Klasse `Star` die Methoden `toString`, `paintMe` und `area`.

Testen Sie die Darstellung von Sternen. Zeichnen Sie dabei auch Sterne, die zu einem Viereck und zu einem Kreis werden. Testen Sie, ob der Flächeninhalt eines Sterns, der einen Kreis annähert, dem Flächeninhalt aus der Klasse `Circle`, gleicht.

Lösung

Star.hpp

```
1  #ifndef STAR_H
2  #define STAR_H
3  #include "GeometricObject.hpp"
4
5  class Star: public GeometricObject{
6   public:
7    double innerR;
8    int beams;
9    Star(Vertex* p,double inner,double outer,int beams);
10
11   virtual double area();
12   virtual std::string toString();
13
14   virtual void paintMe(QPainter* p);
15 };
16 #endif
```

Star.cpp

```
1  #include "Star.hpp"
2  #include <QPolygonF>
3  #include <cmath>
4  #include <iostream>
5
```

```
6  Star::Star(Vertex* p,double inner,double outer,int beams)
7                             :GeometricObject(p,2*outer,2*outer){
8    innerR = inner;
9    this->beams = beams;
10 }
11
12 double Star::area(){
13   return beams*sin(M_PI/beams)*innerR*width/2;
14 }
15
16 std::string Star::toString(){
17     return "Star with "+intToString(beams)+" beams "
18             +GeometricObject::toString();
19 }
20
21 void Star::paintMe(QPainter* p){
22    QPolygonF polygon;
23    const double alpha = M_PI/beams;
24    const double outerR = width/2;
25    for (int i=0;i<beams;i++){
26      const double beta = 2*i*alpha;
27      QPointF outerP((pos->x+width/2+outerR*cos(beta))
28                     ,(pos->y+height/2+outerR*sin(beta)));
29      QPointF innerP(pos->x+width/2+innerR*cos(beta+alpha)
30                     ,pos->y+height/2+innerR*sin(beta+alpha));
31      polygon << outerP << innerP;
32    }
33
34    p->drawPolygon(polygon);
35 }
```