

UNIVERSITÄT  
BAYREUTH

# Geometrie-Labor

## Dynamische Geometrie mit JSXGraph

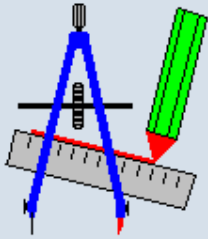
Michael Gerhäuser, Bianca Valentin

Universität Bayreuth

```
var brd = JSXGraph.initBoard('box', {axis: 1});  
var s = brd.createElement('slider', [[1, 3], [5, 10]]);  
var a = brd.createElement('slider', [[1, 2], [5, 10]]);  
var b = brd.createElement('slider', [[1, 1], [5, 10]]);  
f = function(x) { return Math.sin(x); }  
plot = brd.createElement('functiongraph', {  
  f: f,  
  s: s,  
  a: a,  
  b: b,  
  fillColor: '#ffff00',  
});  
brd.createText('f(x) = sin(x)', {x: 10, y: 10});
```

# Dynamische Geometriesysteme

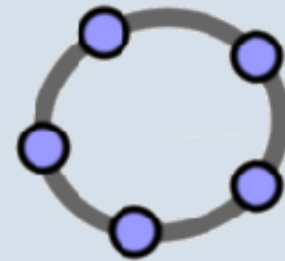
Dynageo



GEONExT



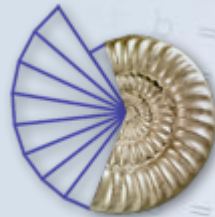
Geogebra



Cinderella



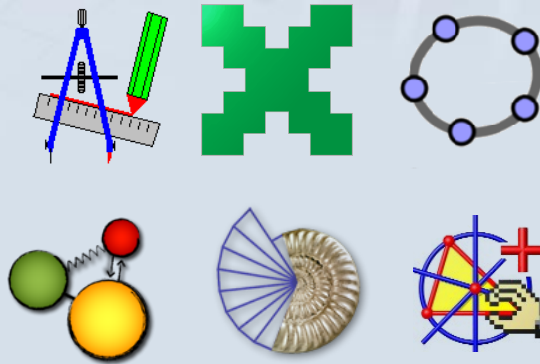
Zirkel und Lineal



Cabri



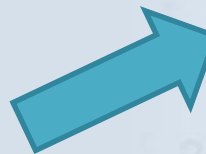
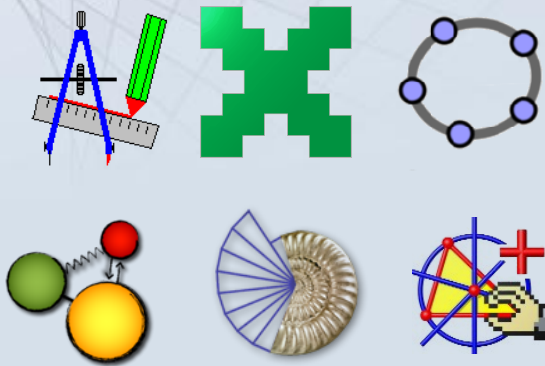
# Dynamische Geometriesysteme



Java™

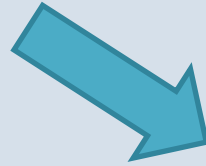
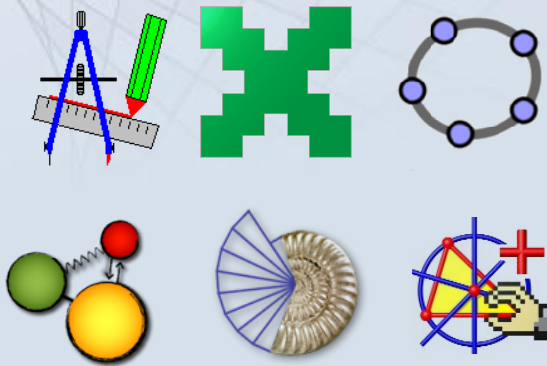
```
brd = JSXGraph.initBoard('box', {axs  
s = brd.createElement('slider', [[1, 3], [5  
a = brd.createElement('slider', [[1, 2], [5  
b = brd.createElement('slider', [[1, 1], [5  
f = function(x){ return Math.sin(x); }  
plot = brd.createElement('functiongraph',  
s = brd.createElement('riemannsum', [f,  
function(){ return s.Value();}, function()  
function(){return a.Value();}, function()  
function(){return b.Value();},  
fillColor: '#ffff00';
```

# Dynamische Geometriesysteme



```
brd =  
s = brd  
a = brd  
b = brd  
f = fun  
lot = k  
  
os = brd.  
function() { return s.Value(); }, function()  
function() { return a.Value(); },  
function() { return b.Value(); },  
fillColor: '#ffff00';
```

# Dynamische Geometriesysteme



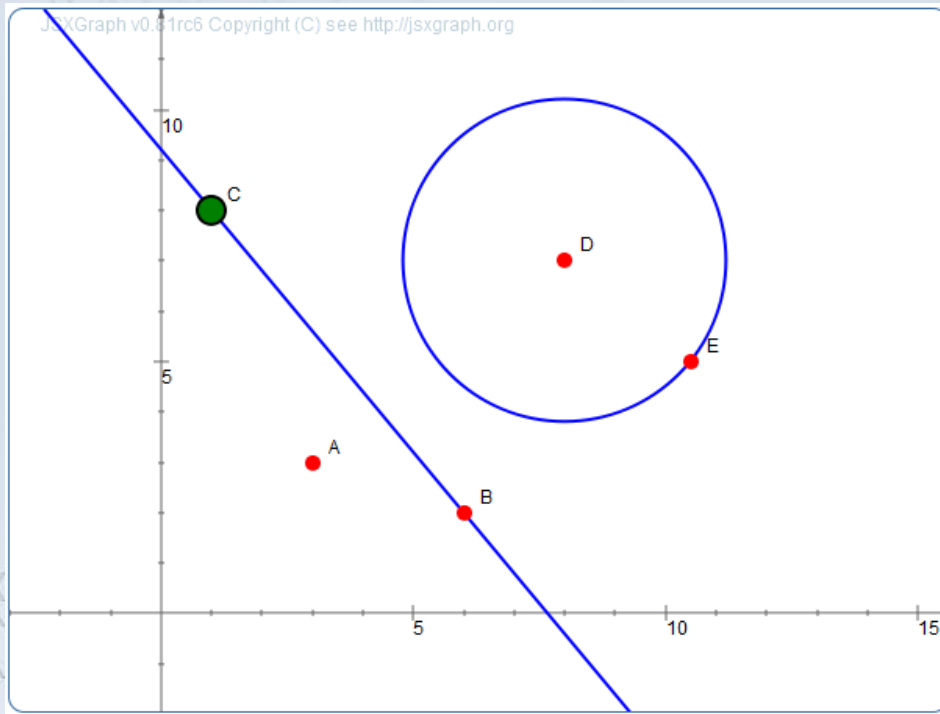


# JSXGraph

- entwickelt an der Universität Bayreuth
- komplett in JavaScript implementiert
- kein Plugin nötig
- unterstützt alle gängigen Browser und das iPhone bzw. iPad

```
var brd = JSXGraph.initBoard('box', {axis: true});
var s = brd.createElement('slider', [[1, 3], [5, 3]]);
var a = brd.createElement('slider', [[1, 2], [5, 2]]);
var f = function(x) { return Math.sin(x); };
var plot = brd.createElement('functiongraph', {
  f: f,
  fillColor: '#ffff00'
});
var riemannsum = brd.createElement('riemannsum', [f, s, a]);
function() { return s.Value(); }, function() { return a.Value(); }, function() { return riemannsum.Value(); }];
```

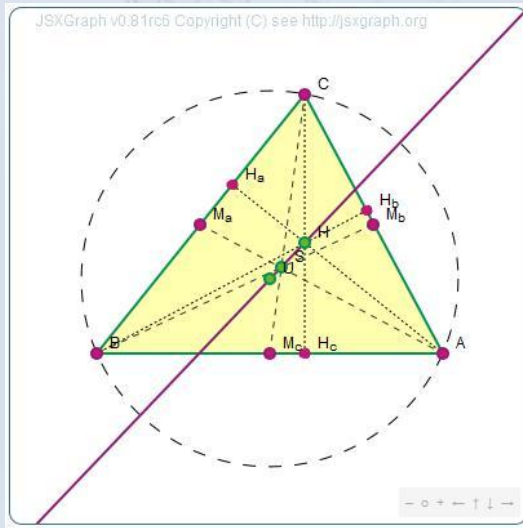
# Erstes Beispiel



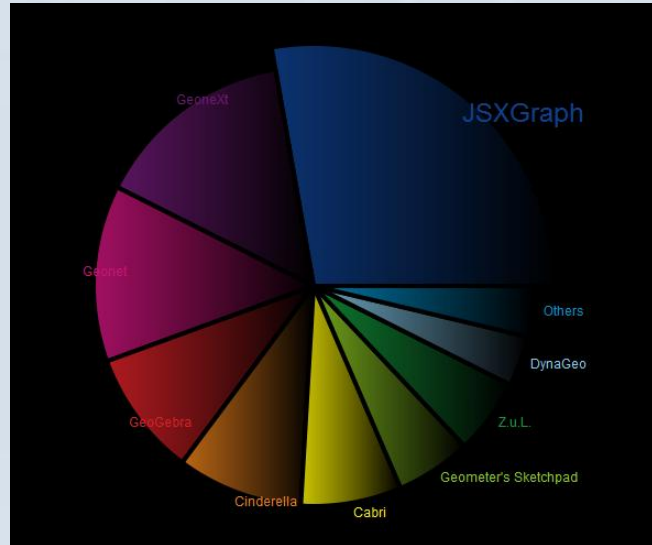
[Live Version](#)

```
var a = brd.create('point', [3,3],  
  {name:'A'});  
var b = brd.create('point', [6,2],  
  {name:'B'});  
var c = brd.create('point', [1,8],  
  {name:'C',strokeColor:'black',  
    fillColor:'green',size:8});  
var l = brd.create('line',[b, c]);  
var d = brd.create('point', [8,7],  
  {name:'D'});  
var e = brd.create('point',  
  [10.5,5], {name:'E'});  
var cr = brd.create('circle',  
  [d, e], {name:'k_1'});  
var i1 =  
  brd.create('intersection',  
  [cr, l, 0], {face:'square'});  
var i2 =  
  brd.create('intersection',  
  [cr, l, 1], {face:'[]', size:7});  
os = brd.createElement('functiongraph',  
  function(){ return s.Value();}, function()  
  function(){return a.Value();},  
  function(){return b.Value();},  
  {fillColor:'#ffff00',
```

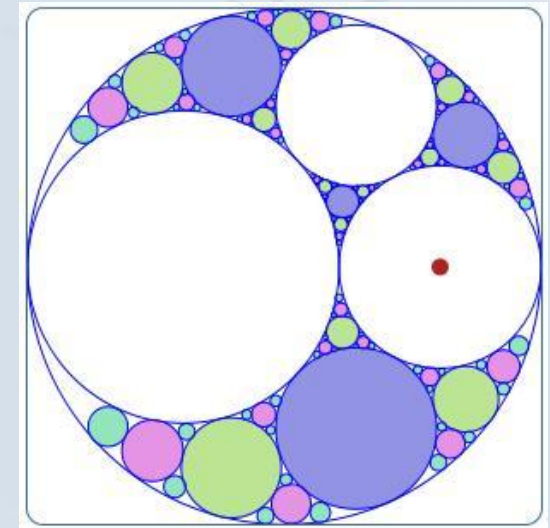
# Weitere Beispiele



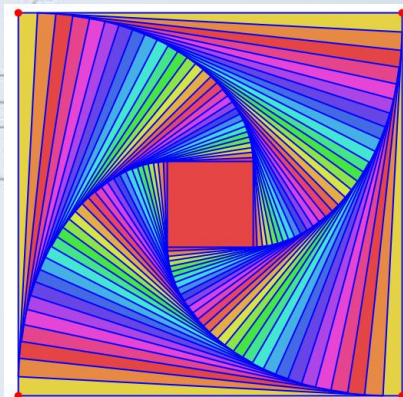
Geometrie: Eulergerade



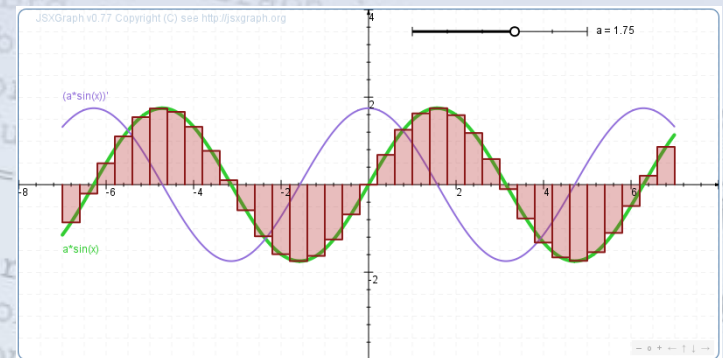
Diagramme



Apollonische Kreispackung



„Infinity“



Funktionen zeichnen



# Konstruieren mit JessieScript

$|(P,g)$   $g=]AB[$

$P(1,1)$

$1/2(A,B)$

$\angle(A,B,C)$

$Y[A,B,C,D]$

Konstruieren mit JessieScript

$Q(g,3,2)$

$f:x^2+2*x+5$

$|(P,g)$

$X=g\&k1$

$k(A,[PQ])$

```
var brd = JSXGraph.initBoard('box', {axis:
var s = brd.createElement('slider', [[1,3],[5
var a = brd.createElement('slider', [[1,2],[5
var b = brd.createElement('slider', [[1,1],[5
var f = function(x) { return Math.sin(x); }
var plot = brd.createElement('functiongraph',
var os = brd.createElement('riemannsum', [f,
function() { return s.Value(); }, function()
function() { return a.Value(); },
function() { return b.Value(); },
fillColor: '#ffff00';
```

# Ausprobieren!



<http://jsxgraph.uni-bayreuth.de/jessie>