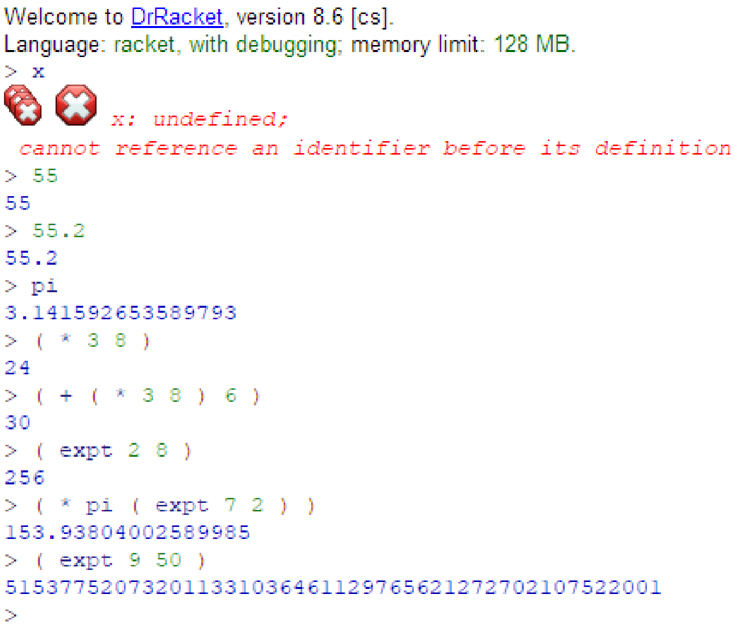
Racket Programming Assignment #1: First Interactions

Brandon LaPointe

Learning Abstract

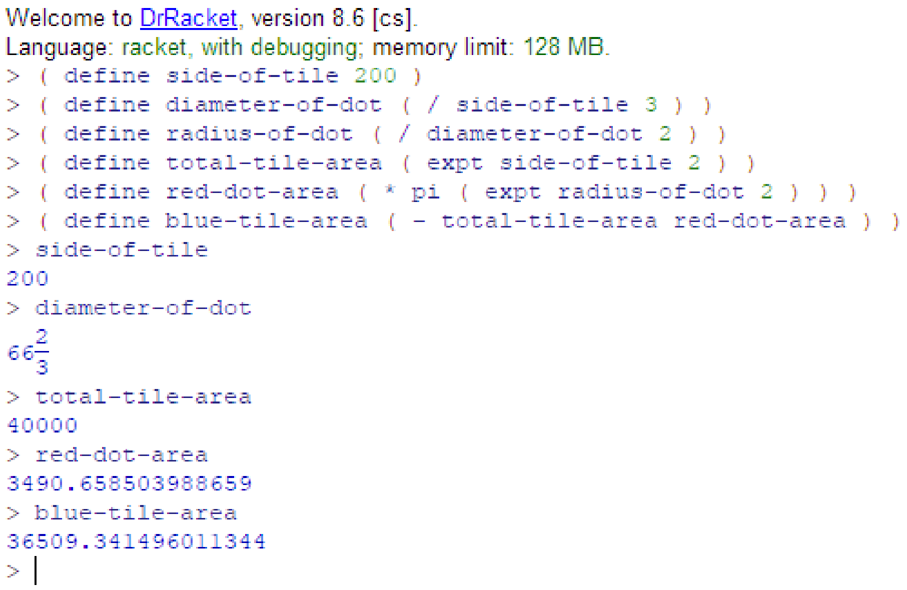
This assignment features relatively simple interactions in the Racket programming language. In fact, all of the computations take place within the interactions pane of the DrRacket PDE. In the first part of this assignment, I learned a little bit about numeric computations in Lisp. The next two parts of the assignment featured a square tile which was blue except for a centered red dot. In the second part of the assignment, I mimicked the solution of the problem of finding the area of the tile which was blue. In the third part, I mimicked the computational rendering of the tile. The last two parts of the assignment featured an image consisting of 5 concentric circles. In the fourth part of this assignment, I rendered the image of the concentric circles, consisting of 3 blue circles and 2 red circles overlayed over one another. In the fifth part, I computed the area of the red and blue portions based on the concentric circles image rendered in the fourth part of the assignment. Throughout the problem-solving segments of this assignment, the concept of binding values to variables was a predominant theme.

Interaction: Simple Numeric Processing



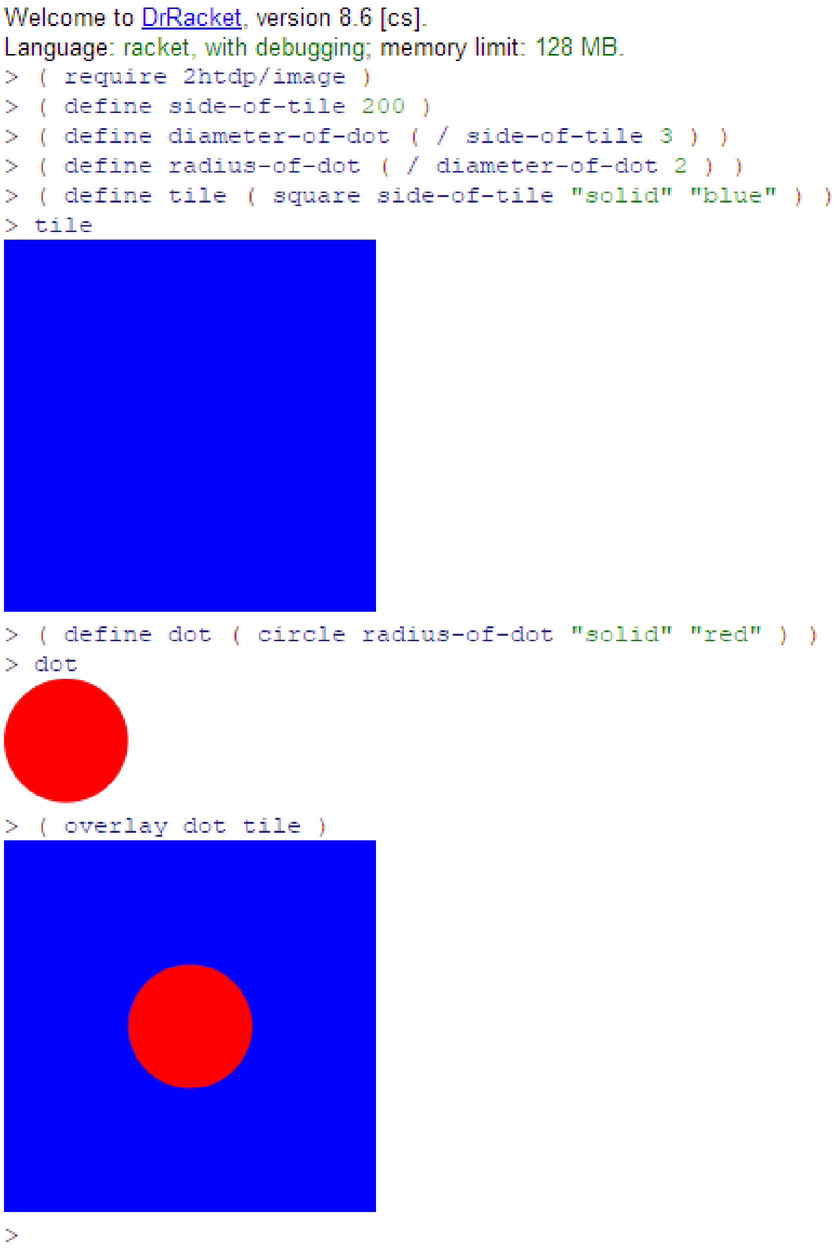
```
Welcome to DrRacket, version 8.6 [cs].
Language: racket, with debugging; memory limit: 128 MB.
> x
x: undefined;
 cannot reference an identifier before its definition
> 55
55
> 55.2
55.2
> pi
3.141592653589793
> ( * 3 8 )
24
> ( + ( * 3 8 ) 6 )
30
> ( expt 2 8 )
256
> ( * pi ( expt 7 2 ) )
153.93804002589985
> ( expt 9 50 )
515377520732011331036461129765621272702107522001
>
```

Interaction: Solution to the Red Circle and Blue Tile Area Problem



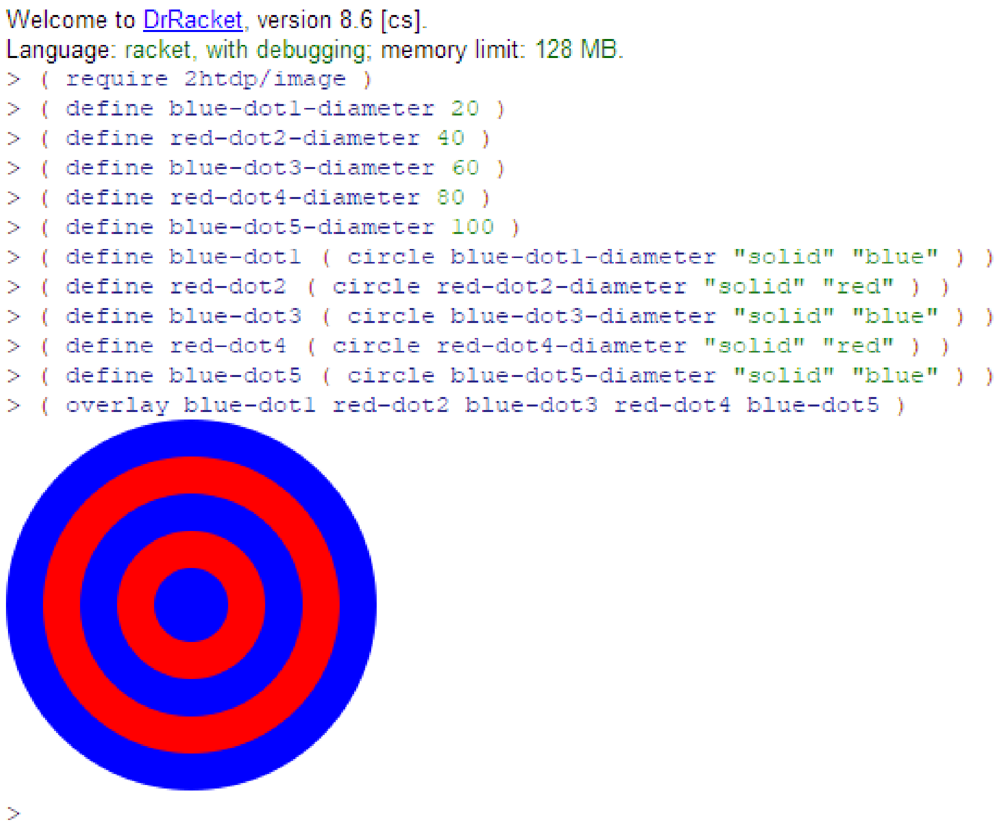
```
Welcome to DrRacket, version 8.6 [cs].
Language: racket, with debugging; memory limit: 128 MB.
> ( define side-of-tile 200 )
> ( define diameter-of-dot ( / side-of-tile 3 ) )
> ( define radius-of-dot ( / diameter-of-dot 2 ) )
> ( define total-tile-area ( expt side-of-tile 2 ) )
> ( define red-dot-area ( * pi ( expt radius-of-dot 2 ) ) )
> ( define blue-tile-area ( - total-tile-area red-dot-area ) )
> side-of-tile
200
> diameter-of-dot
66 2/3
> total-tile-area
40000
> red-dot-area
3490.658503988659
> blue-tile-area
36509.341496011344
> 
```

Interaction: Painting the Red Circle and Blue Tile



```
Welcome to DrRacket, version 8.6 [cs].
Language: racket, with debugging; memory limit: 128 MB.
> ( require 2htdp/image )
> ( define side-of-tile 200 )
> ( define diameter-of-dot ( / side-of-tile 3 ) )
> ( define radius-of-dot ( / diameter-of-dot 2 ) )
> ( define tile ( square side-of-tile "solid" "blue" ) )
> tile
> ( define dot ( circle radius-of-dot "solid" "red" ) )
> dot
> ( overlay dot tile )
>
```

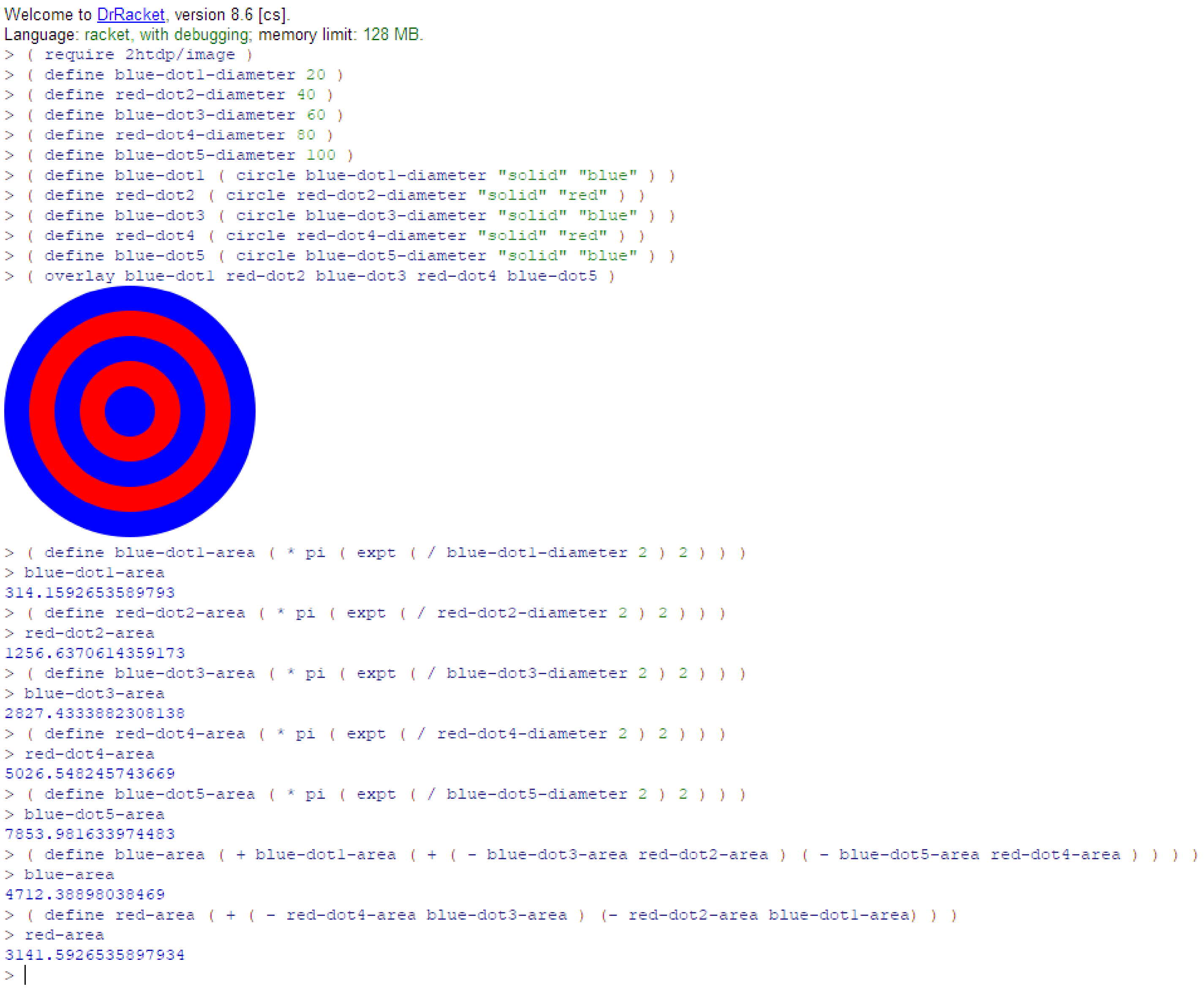
Interaction: Painting the Red and Blue Concentric Disks' Image



```
Welcome to DrRacket, version 8.6 [cs].
Language: racket, with debugging; memory limit: 128 MB.
> ( require 2htdp/image )
> ( define blue-dot1-diameter 20 )
> ( define red-dot2-diameter 40 )
> ( define blue-dot3-diameter 60 )
> ( define red-dot4-diameter 80 )
> ( define blue-dot5-diameter 100 )
> ( define blue-dot1 ( circle blue-dot1-diameter "solid" "blue" ) )
> ( define red-dot2 ( circle red-dot2-diameter "solid" "red" ) )
> ( define blue-dot3 ( circle blue-dot3-diameter "solid" "blue" ) )
> ( define red-dot4 ( circle red-dot4-diameter "solid" "red" ) )
> ( define blue-dot5 ( circle blue-dot5-diameter "solid" "blue" ) )
> ( overlay blue-dot1 red-dot2 blue-dot3 red-dot4 blue-dot5 )
```

```
>
```

Interaction: Computing the Area of the Concentric Disks’ Image



```
Welcome to DrRacket, version 8.6 [cs].
Language: racket, with debugging; memory limit: 128 MB.
> ( require 2htdp/image )
> ( define blue-dot1-diameter 20 )
> ( define red-dot2-diameter 40 )
> ( define blue-dot3-diameter 60 )
> ( define red-dot4-diameter 80 )
> ( define blue-dot5-diameter 100 )
> ( define blue-dot1 ( circle blue-dot1-diameter "solid" "blue" ) )
> ( define red-dot2 ( circle red-dot2-diameter "solid" "red" ) )
> ( define blue-dot3 ( circle blue-dot3-diameter "solid" "blue" ) )
> ( define red-dot4 ( circle red-dot4-diameter "solid" "red" ) )
> ( define blue-dot5 ( circle blue-dot5-diameter "solid" "blue" ) )
> ( overlay blue-dot1 red-dot2 blue-dot3 red-dot4 blue-dot5 )
```

```
> ( define blue-dot1-area ( * pi ( expt ( / blue-dot1-diameter 2 ) 2 ) ) )
> blue-dot1-area
314.1592653589793
> ( define red-dot2-area ( * pi ( expt ( / red-dot2-diameter 2 ) 2 ) ) )
> red-dot2-area
1256.6370614359173
> ( define blue-dot3-area ( * pi ( expt ( / blue-dot3-diameter 2 ) 2 ) ) )
> blue-dot3-area
2827.4333882308138
> ( define red-dot4-area ( * pi ( expt ( / red-dot4-diameter 2 ) 2 ) ) )
> red-dot4-area
5026.548245743669
> ( define blue-dot5-area ( * pi ( expt ( / blue-dot5-diameter 2 ) 2 ) ) )
> blue-dot5-area
7853.981633974483
> ( define blue-area ( + blue-dot1-area ( + ( - blue-dot3-area red-dot2-area ) ( - blue-dot5-area red-dot4-area ) ) ) )
> blue-area
4712.38898038469
> ( define red-area ( + ( - red-dot4-area blue-dot3-area ) (- red-dot2-area blue-dot1-area) ) )
> red-area
3141.5926535897934
> 
```