Math on a Sphere – Design Challenges

This document contains suggested design challenges and the associated code for creating these designs. These designs can be incorporated into your lesson plan or given to students who come to your workshop/class with advanced programming skills.

There are currently more than 25 design challenges. The initial summary pages indicate the names of the design challenges as well as screenshots of the designs. The later pages have the code for making the designs.

The design challenges are organized according to the background knowledge needed to complete them. For example, do learners need to understand the repeat command, or if statements, before attempting a particular design challenge? These classifications are intended to describe the basic knowledge that makes it possible to complete these designs. Many of the designs will still be tricky – after all, they are supposed to be challenges!

We highly encourage feedback about how we are classifying our design challenges. Do you think that certain challenges are much harder or easier than their classification implies? Please send your feedback to Lizzie Hager-Barnard at the Lawrence Hall of Science (lizziehb@berkeley.edu).
Fairly easy – can be done without knowledge of advanced commands

House

Beach ball version 1

Beach ball version 2

Running turtle
Tricky – but mostly just a guess and check process

Interference
** Kind of tricky, until you figure out what could be causing the interference pattern

Figure 8
** Hard to make specific curves, hard to come back to starting point

Wiggle
** Hard to make specific curves
Learners need to know how to make circles that aren't great circles

Smiley face

Rainbow

Bullseye target

Spiral (not using repeats)
Learners need to know how to make circles that aren’t great circles, continued

Dartboard

Track

Circles with a common point
Need to understand geometry

Square inscribed in a circle

Need to understand what variables are and how to use them; need to know the repeat command

Spiral (using repeats)

Triangles to squares spiral
**Tricky until you figure out that the turtle will naturally make triangles when it’s in the outer part of the spiral and squares when it’s down in the tight part of the spiral

Squares to hexagons spiral
**Tricky until you figure out that the turtle will naturally make triangles when it’s in the outer part of the spiral and squares when it’s down in the tight part of the spiral
Need to understand what variables are and how to use them; need to know the **repeat** command; need to know what **if statements** are and how to use them

Staircase

**Staircase with balls**

**Determining placement of the balls is tricky, but can be done through guess and check**

Spiral with **if statements**
Need to understand what variables are and how to use them; need to know the *repeat* and *random* commands

Random circles

Need to understand what variables are and how to use them; need to know how to use the *repeat* and *random* commands; need to know what *if statements* are and how to use them

Random shapes
This is a set of examples that can be used to teach learners more about how the pensize and turn angles affect the resulting design. These solid ring examples show that it can be kind of tricky to figure out if you’re going to form a solid ring or not. The specific images here don’t have to be duplicated – the idea is to have your learners try changing the pensize and turn angles, to see how this affects the design. If you have advanced learners, you could have them calculate if a solid ring will be formed, before they try running their designs.

Solid ring attempt #1

Solid ring attempt #2

Solid ring attempt #3

Solid ring attempt #4
Very advanced: lots of code, but no specific background knowledge required

Temari cube pattern

Temari octahedral “tape” pattern

Temari icosahedral pattern

Temari dodecahedral pattern
setheading 180
penup
forward 40

pendown
setheading 95
forward 30
left 82
forward 35

right 135
forward 10
left 130
forward 15

left 35
forward 45
left 65
forward 45

left 45
forward 15
left 135
forward 10

right 145
forward 40

left 80
forward 30

penup
right 92
forward 20
setspeed 50

set color orange

repeat 60 {
    forward 360
    right 1
}

set color blue

repeat 60 {
    forward 360
    right 1
}

set color yellow

repeat 60 {
    forward 360
    right 1
}
setspeed 50
set pensize 2

set color orange
repeat 30 {
    forward 180
    penup
    forward 180
    pendown
    right 2}

set color blue
repeat 30 {
    forward 180
    penup
    forward 180
    pendown
    right 2}

set color yellow
repeat 30 {
    forward 180
    penup
    forward 180
    pendown
    right 2}

set color red
repeat 30 {
    forward 180
    penup
    forward 180
    pendown
    right 2}

set color cyan

repeat 30 {
    forward 180
    penup
    forward 180
    pendown
    right 2}

set color white

repeat 30 {
    forward 180
    penup
    forward 180
    pendown
    right 2}

set color 0
set pensize 6
forward 2 right 90
repeat 60 {forward 2 right 33}

setheading 0
penup
forward 190
pendown

right 90
repeat 60 {forward 2 right 33}
set pensize 7
penup
back 90
repeat 20 {
  penup
  setspeed 50
  forward 30
  pendown
  setspeed 1
  forward 6
  penup
  setspeed 50
  forward 336
}
Title: Interference pattern

set speed 50
set pensize 1.33

set color orange

repeat 180 {
    forward 360
    right 2}
Title: Figure 8

Goal

right 45
forward 30

repeat 20 {
  forward 0.5
  right 4
}

repeat 40 {
  forward 0.5
  right 2
}

repeat 20 {
  forward 0.5
  right 4
}

right 22
forward 30

repeat 20 {
  forward 0.5
  left 4
}

repeat 40 {
  forward 0.5
  left 2
}
Last updated on 2013-04-23

repeat 20 {
   forward 0.5
   left 4
}

penup
setposition [30,20]
m = 0

repeat 40 {
    rt (20-m) fd 1
    m = m+1
}

penup
forward 100
set color orange
repeat 120 {
    forward 1
    right 3
}

penup
setposition [0,8]
setheading 180
pendown

repeat 34 {
    forward 1
    left 5.5
}

penup
setposition [6, 14]
pendown

repeat 12 {
    forward 0.5
    left 30
}

penup
setposition [5, 24]
pendown

repeat 12 {
    forward 0.5
}
Last updated on 2013-04-23

left 30
}

penup
setposition [40,40]
set color red
repeat 58 {
  forward 1
  right 3
}

penup
setposition [0,2]
setheading [0]
pendown

set color orange
repeat 51 {
  forward 1
  right 3.45
}

penup
setposition [0,4]
setheading [0]
pendown

set color yellow
repeat 46 {
  forward 1
  right 3.9
}

penup
setposition [0,6]
setheading [0]
pendown

set color green
repeat 41 {
  forward 1
  right 4.35
}

penup
setposition [0,8]
setheading [0]
pendown

set color cyan
repeat 37 {
  forward 1
  right 4.95
}

penup
setposition [0,10]
setheading [0]
pendown

set color blue
repeat 30 {
  forward 1
  right 6
}

penup
setposition [30,20]
Title: Bullseye target

penup
setposition [0,4]
setheading 0
pendown

set color red
set pensize 3

repeat 60 {
  forward 2
  right 6
}

penup
setposition [0,11]
setheading 0
pendown

set color white
set pensize 3

repeat 40 {
  forward 2
  right 10
}

penup
setposition [0,14]
setheading 0
pendown
set color red
set pensize 3

repeat 30 {
  forward 2
  right 14
}

penup
setposition [0,20]
setheading 0
pendown

set color white
set pensize 3.5

repeat 25 {
  forward 2
  right 50
}

penup
setposition [30,0]
Title: Spiral (not using repeats)

right 90
forward 20
right 20
forward 20
right 21
forward 20
right 22
forward 20
right 23
forward 20
right 24
forward 20
right 25
forward 20
right 26
forward 20
right 27
forward 20
right 28
forward 20
right 29
forward 20
right 30
forward 20
right 31
forward 20
right 32
forward 18
right 33
forward 18
right 34
forward 18
right 35
forward 18
right 36
forward 16
right 37
forward 16
right 38
forward 16
right 39
forward 16
right 40
forward 16
right 41
forward 16
right 42
forward 14
right 43
forward 14
right 44
forward 14
right 45
forward 14
right 46
forward 14
right 47
forward 12
right 48
forward 12
right 49
forward 12
right 50
forward 10
right 51
forward 10
right 52
forward 10
right 53
forward 8
right 54
forward 8
right 55
forward 6
penup
setposition [0,4]
setheading 0
pendown

set pensize 10

repeat 18 {
  set color yellow
  forward 5
  right 10
  set color 1
  forward 5
  right 10
}

penup
setposition [3,10]
setheading 0
pendown

set pensize 3

right 12.75

repeat 13 {
  set color red
  forward 4.5
  right 12.3
  set color green
  forward 4.5
right 12.3
}  

cleanup
setposition [40,0]
penup
setposition [0,4]
setheading 0
pendown
set pensize 10
repeat 15 {
  set color yellow
  forward 6
  right 10
  set color 1
  forward 6
  right 10
}
penup
setposition [0,4]
setheading 0
pendown
set pensize 2
repeat 14 {
  set color 1
  forward 6
  right 10
  set color yellow
  forward 6
  right 10
}
Title: Circles with a common point

repeat 100 {
  forward 2
  right 3
}

penup
setposition [0,0]
setheading 0
pendown

set color red
repeat 90 {
  forward 3
  right 3
}

penup
setposition [0,0]
setheading 0
pendown

set color green
repeat 120 {
  forward 1
  right 3
}


penup
setposition [0,0]
setheading 0
pendown

set pensize 2

repeat 33 {
    set color yellow
    forward 5
    right 10
}

penup
setposition [21,8]
setheading 85
pendown

forward 36

right 85
forward 36

right 85
forward 36

right 85
forward 36

penup
forward 20
Title: Spiral (using repeats)

\[
x = 20 \\
a = 20 \\
right 90 \\
repeat 60 \{ \\
  forward \ x \\
  right \ a \\
  x = x - 0.33 \\
  a = a + 0.5 \\
\} \\
penup \\
setheading 180 \\
forward 50
Title: Triangles to squares spiral

\[ x = 96 \]
\[ a = 90 \]

penup
forward 20
left a
forward 30

setheading a

pendown
repeat 12 {
  forward x
  right a
  \[ x = x \times 0.8 \]
}

penup
setheading 180
forward 50
Title: Squares to hexagons spiral

\[ x = 96 \]
\[ a = 60 \]

penup
forward 20
left a
forward 30
setheading a

pendown
repeat 15 {
    forward x
    right a
    x = x * 0.8
}

penup
setheading 180
forward 50
Title: Staircase

set speed 50

set pensize 0.75
x = 10
theta = 0
rowNumber = 0

right 90

repeat (24) {
  if ( (rowNumber/3) == (Math.floor(rowNumber/3)) ) {x = x + 5}
  forward x
  right 180
  forward x

  right 90
  back 1
  right 90

  rowNumber = rowNumber + 1
}

}
setspeed 50
set pensize 0.75

x = 10
theta = 0
rowNumber = 0

right 90
repeat (24) {
    if ( (rowNumber/3) == (Math.floor(rowNumber/3)) ) {x = x + 5}
    forward x
    right 180
    forward x
    right 90
    back 1
    right 90
}

rowNumber = rowNumber + 1
}

setspeed 5
set pensize 1

penup
setposition [4,8]
setheading 0
set color orange
Title: Spiral with if statements

x = 96
a = 60
counter = 0

penup
forward 20
left a
forward 30

setheading a

pendown
repeat 15 {

    if (counter/3 == Math.floor(counter/3)) {
        set color pink
        set pensize 1
        repeat 30 { rt (360/30) fd (2-counter/8) }
    }

    set color yellow
    forward x
    right a
    x = x * 0.8
    counter = counter + 1
}

penup
setheading 180
forward 50
Title: Random circles

```
repeat 40 {
  rand = (Math.random())
  set color (rand*139)
  set pensize (rand*3)

  stepLength = (Math.floor(rand*20))
  steps = 20
  repeat steps { rt (360/steps) fd stepLength}
}
```
Title: Random shapes

```javascript
repeat 15 {
    rand = (Math.random())
    set color (rand*139)
    set pensize (rand*3)
    setspeed 50

    xPos = (Math.floor(200*(Math.random())))
    yPos = (Math.floor(200*(Math.random())))

    penup
    setposition [xPos,yPos]
    pendown

    stepLength = (Math.floor(rand*20))
    console.log(stepLength)

    if (stepLength > 5) {
        steps = 20
        repeat steps { rt 90 fd stepLength}
    }
    else {
        steps = 20
        repeat steps { rt (360/steps) fd stepLength}
    }
}
```
Title: Solid ring attempts

Solid ring attempt #1
set pensize 20
repeat 100 {forward 2 right 7}

Solid ring attempt #2
set pensize 20
repeat 100 {forward 2 right 20}

Solid ring attempt #3
set pensize 8
repeat 40 {forward 2 right 24}

Solid ring attempt #4
set pensize 8
repeat 40 {forward 2 right 30}
Title: Temari cube pattern

```javascript
sq = function [n] {
    repeat 4 {fd 70.5 rt 60}}

cb = function [m] {
    repeat 3 {sq 0 rt 120}
    pu
    fd 180
    pd
    repeat 3 {sq 0 rt 120}
    pu
    bk 180
    pd
}

set pensize 12
set color white
repeat 6 {cb 0 rt 60}

set color red
lt 30
repeat 6 {fd 65 rt 35 fd 52 bk 52 lt 35 bk 65 rt 60}
rt 60
repeat 6 {fd 65 lt 35 fd 52 bk 52 rt 35 bk 65 rt 60}
lt 30

pu

fd 180
pd

lt 30
repeat 6 {fd 65 rt 35 fd 52 bk 52 lt 35 bk 65 rt 60}
```
rt 60
repeat 6 {fd 65 lt 35 fd 52 bk 52 rt 35 bk 65 rt 60}
lt 30

pu fd 180
pd

set color blue
set pensize 6
repeat 6 {cb 0 rt 60}
Title: Temari octahedral “tape” pattern

triline = function [n] {
    set color white
    pu fd 30 pd
    set color red
    rt 45
    bk 20
    fd 20
    set color yellow
    lt 90
    bk 20
    fd 20
    rt 45
    set color white
    pu fd 30 pd
    set color yellow
    rt 45
    bk 20
    fd 20
    set color red
    lt 90
    bk 20
    fd 20
    set color white
    rt 45
    pu fd 30 pd
}

tri = function [n] {
    repeat 3 {triline 90 rt 90}
}

oct = function [m] {
repeat 4 {tri 0 rt 90} 
pu 
fd 180 
pd 
repeat 4 {tri 0 rt 90} 
pu 
bk 180 
pd 
} 

set pensize 6 
set color white 
repeat 6 {oct 0 rt 60} 
// pu 
// rt 90 
// fd 90 
// lt 90 
// pd 
// repeat 6 {oct 0 rt 60} 
// pu 
// rt 90 
// bk 90 
// lt 90 
// pd
Title: Temari icosahedral pattern

crc = function[x] { repeat 20 { pu fd x rt 90 pd
    fd 6 bk 12 fd 6 lt 90 pu bk x rt 18 }
  }

set pensize 6
set color brown
repeat 5{repeat 3 {fd 63.5 rt 108} rt 72}
set pensize 4
set color red
repeat 5 {pu rt 36 fd 37.4128 crc 36 crc 18 pu bk 37.4128
    lt 36 rt 72}
pu fd 180 pd
set pensize 6
set color brown
repeat 5 {repeat 3 {fd 63.5 rt 108} rt 72}
set pensize 4
set color red
repeat 5 {pu rt 36 fd 37.4128 crc 36 crc 18 pu bk 37.4128
    lt 36 rt 72}
pu fd 180 pd
pu fd 63.5 rt 36 pd
set pensize 6
set color brown
repeat 5 {fd 63.5 rt 108 fd 63.5 lt 108}
set pensize 4
set color red
repeat 5 {pu rt 36 fd 37.4128 crc 36 crc 18 pu bk 37.4128
    lt 36 fd 63.5 rt 108
    pu lt 36 fd 37.4128 crc 36 crc 18 pu bk 37.4128
    rt 36 fd 63.5 lt 108}
lt 36 pu bk 63.5 pd
set pensize 3
set color blue

repeat 5 {repeat 3 {fd 63.5 rt 108} rt 72}
pu fd 180 pd
repeat 5 {repeat 3 {fd 63.5 rt 108} rt 72}
pu fd 180 pd
pu fd 63.5 rt 36 pd
repeat 5 {fd 63.5 rt 108 fd 63.5 lt 108}
l 36 pu bk 63.5 pd

set pensize 1
set color brown

repeat 5 {repeat 3 {fd 63.5 rt 108} rt 72}
pu fd 180 pd
repeat 5 {repeat 3 {fd 63.5 rt 108} rt 72}
pu fd 180 pd
pu fd 63.5 rt 36 pd
repeat 5 {fd 63.5 rt 108 fd 63.5 lt 108}
l 36 pu bk 63.5 pd
Title: Temari dodecahedral pattern

```plaintext
pent = function [x]
    { repeat 5 {set color gray set pensize 8
       rt 60 pd fd 68 bk 68
       set color sky set pensize 5
       fd 68 bk 68
       set color cyan set pensize 2
       fd 68 bk 68
       lt 60
       pu set color white set pensize 1
       fd 41.81 rt 60} }
regpent = function [x]
    { repeat 5 {fd 41.81 rt 60} }

repeat 3 {pent 1 rt 120}
pu fd 180 pd
repeat 3 {pent 1 rt 120}
pu fd 180 pd
pu fd 41.81 lt 60 pd
repeat 2 { pent 1
        pu fd 41.81 lt 60 pd }

pent 1
pu fd 41.81 rt 60 pd
pu fd 41.81 lt 60 pd
pent 1
pu fd 41.81 lt 60 pd
pent 1
pu fd 41.81 rt 60
pu fd 41.81 lt 60
pent 1
pu rt 60 bk 41.81
pu lt 120 fd 41.81 rt 60
```
set color blue
set pensize 2
pd

repeat 3 {regpent 1 rt 120}
pu fd 180 pd
repeat 3 {regpent 1 rt 120}
pu fd 180 pd
pu fd 41.81 lt 60 pd
repeat 2 {regpent 1
    pu fd 41.81 lt 60 pd }
regpent 1
pu fd 41.81 rt 60 pd
pu fd 41.81 lt 60 pd
regpent 1
pu fd 41.81 lt 60 pd
regpent 1
pu fd 41.81 rt 60
pu fd 41.81 lt 60
regpent 1
pu rt 60 bk 41.81
pu lt 120 fd 41.81 rt 60