Teacher_harry henderson
Grade.Quarter.Unit_______________
Day(s)__1____
Dates:_ TBD _

RRPS SECONDARY MATH UNIT/LESSON PLAN TEMPLATE 2013-2014

UNIT TITLE:_____spinning compass part 2_____

Stage 1 – Preparation (PLC)

Grade Level Content Standard(s)/Standard(s) for Mathematical Practice	DOK levels and Learning Targets	Essential Question(s)
N-Q.2 Define appropriate quantities for the purpose of descriptive modeling. G-MG Modeling with Geometry, Apply geometric concepts in modeling situations	1, 2, 3 Applying geometry law of sines and law of cosigns equations to real world problem. Modifying agent based model into a serial code [C++] then paralellize. Another object is optimizing the code. Using technology to explore mathematical relations.	How do we convert between agent based and serial? How do we optimize the code? How do we paralellize the code.

Resources / Materials Needed
computer with access to a multi-core supercomputer with OpenMP

This material is based upon work supported by the National Science Foundation under the NSF EPSCoR Cooperative Agreement No. EPS-1003897 with additional support from the Louisiana Board of Regents

Objectives: cover phase 2

PHASE 1: Develop differential equations to represent the physical object
PHASE 2: Convert agent based code to serial code and use those equations to build the serial code in C++
PHASE 3: Optimize code using gradient
PHASE 4: Introduce and apply OpenMP to the serial code

Prior knowledge:
Familiarity with programming preferably with both agent based and serial coding.
Geometry/trigonometry math background.

Stage 2 – Implementation (PLC)

Suggested Time/ Dates	Setting (Whole Group, Small Group, Pairs, Independent)	Lesson/Unit Components	Assessment options/how used
5-10 minutes	Whole group	**Warm Up(s)** review agent based model breed [axles axle] breed [magnets magnet] breed [metals metal] axles-own	Provide premade serial codes if time or learning abilities are an issue. Resources: http://openmp.org/wp/resources/ https://computing.llnl.gov/tutorials/openMP/ http://openmp.org/mp-documents/omp-hands-on-SC08.pdf

Teacher_harry henderson
Grade.Quarter.Unit_______________
Day(s)__1____
Dates:_ TBD _

RRPS SECONDARY MATH UNIT/LESSON PLAN TEMPLATE 2013-2014

UNIT TITLE:_____spinning compass part 2_____

		[
		  angular-velocity
		]
		
		globals
		[
		  hub
		  mag-force-constant ; Multiplier for magnetic attraction; incorporates not only
		                    ; magnetic force, but also torque arm length.
		
		  friction ; Drag acting opposite to rotation.
		
		  finished?
		  total_rotation
		  max_rotation
		  direction
		  direction_check
		
		  number_mag
		  mag_list
		  metal_list
		  number_metal
		
		  max_velocity
		
		  counter_mag
		  counter_metal
		  happy_mag
		  happy_metal
		
		  number-magnets
		]
		
		to setup
		; clear-all
		  clear-turtles
		  clear-links
		  setup-globals
		  setup-compass
		  setup-field
		  initiate	

```
  reset-ticks
end

to setup-globals
  set total_rotation 0
;  set max_rotation 0
  set direction 0
  set direction_check 0

  set mag-force-constant 1
;  set friction 0.999999
  set friction 1 - 1 / 10 ^ friction_scale
  if friction = 1 - 1 / 10 ^ 10
  [
    set friction 1
  ]
  set finished? false

  set counter_mag 0
  set happy_mag true
  mag_loop
end

to mag_loop
  while [happy_mag]
  [
    ifelse (item (length mag_list - 1) mag_list) = (360 - (length mag_list) * 1 )  ; all done
    [
;       set counter_metal 0
;       set happy_metal true
;       set mag_list [32 31]
;       set mag_list [352 344 336 320]
       set happy_mag false
;       metal_loop
    ]
    [
      ifelse (358 - counter_mag * 1) = item counter_mag mag_list
      [
        set mag_list replace-item counter_mag mag_list ((item (counter_mag + 1) mag_list) + 1)
        set counter_mag counter_mag + 1
      ]
```

```
      [
        set mag_list replace-item counter_mag mag_list ((item counter_mag mag_list) + 1)
        set happy_mag false
        set counter_mag 0
      ]
    ]
  ]
end

to metal_loop ; set for 6 metals at 60 degree spacing
  while [happy_metal]
  [
    ifelse (item (length metal_list - 1) metal_list) = 353  ; all done
    [
       set max_rotation 8000
       set happy_metal false
       metal_loop
    ]
    [
      ifelse ((60 * (counter_metal + 1) - 4) + 1) = item counter_metal metal_list
      [
        set metal_list replace-item counter_metal metal_list ((counter_metal * 60) + 1)
        set counter_metal counter_metal + 1
      ]
      [
        set metal_list replace-item counter_metal metal_list ((item counter_metal metal_list) + 8)
        set happy_metal false
        set counter_metal 0
      ]
    ]
  ]
end

to setup-compass
  set metal_list mag_list

  create-axles 1
  [
    set angular-velocity 0
    set heading 90
    set hub self
```

Teacher_harry henderson

Grade.Quarter.Unit______________

Day(s)__1____

Dates:_ TBD _

RRPS SECONDARY MATH UNIT/LESSON PLAN TEMPLATE 2013-2014

UNIT TITLE:_____spinning compass part 2_____

		<pre> set color white
		  ]
		; let turtle-count (count turtles)
		;set mag_list sentence random 360 random 360
		;repeat random 10
		;[
		; set mag_list lput (random 360) mag_list
		;]
		; set mag_list [0]
		  ;set number-magnets 4
		  create-magnets 1
		  [
		    set size 4
		    set color white
		; set heading random-float 360
		; set label who
		; set label item (who - 1) mag_list
		; set heading item (who - 1) mag_list
		; set heading (item (number-magnets - who) mag_list)
		    set heading 0
		    jump 20
		    create-link-from hub ; No need to ask the hub to create this link
		    [
		      tie
		    ]
		  ]
		end
		
		to setup-field
		  let turtle-count (count turtles)
		; set metal_list [10]
		;set metal_list sentence random 360 random 360
		;repeat random 10
		;[
		; set metal_list lput (random 360) metal_list
		;]
		; set metal_list [61 179 299]
		
		; set metal_list lput 179 metal_list
		  create-metals length metal_list ; number-metals
		  [</pre>	

		set size 3 set shape "circle" set color yellow ; set heading random-float 360 ; set heading ((who - (count turtles with [breed = magnets])) * e) ^ 2 set heading (item (who - 2) metal_list) ; set label (who - number-magnets - 1) jump 24] end to initiate let angular-acceleration (mag-force-constant * sum [sum [20 * (sin subtract-headings towards hub towards myself) / (distance myself) ^ 3] of magnets] of metals) ask hub [set angular-velocity (angular-velocity * friction + angular-acceleration) if angular-velocity > max_velocity [set max_velocity angular-velocity] right angular-velocity ifelse angular-velocity > 0 [set direction -1 set direction_check -1 set angular-velocity angular-velocity - direction * 2 * (1 - friction) ; set angular-velocity direction / -500] [set direction 1 set direction_check 1 set angular-velocity angular-velocity - direction * 2 * (1 - friction) ; set angular-velocity direction / -500]]	

```
end

to test
  if max_rotation > 7200
  [
    ask turtle 0
    [
      let t 0
      repeat count turtles - 1
      [
        set t t + 1
        file-write [heading] of turtle t
      ]
      file-write ( precision max_rotation 2)
      file-print 0
    ]
    file-close
    stop
  ]
  go
end

to go
  if (finished?)
  [
  if max_rotation > 355
  [
    ask turtle 0
    [
      let t 0
      repeat count turtles - 1
      [
        set t t + 1
        file-write [heading] of turtle t
      ]
      file-write ( precision max_rotation 2)
      file-print 0
    ]
    set max_rotation 1
  ]
    setup
```

Teacher_harry henderson
Grade.Quarter.Unit_______________
Day(s)__1____
Dates:_ TBD _

RRPS SECONDARY MATH UNIT/LESSON PLAN TEMPLATE 2013-2014

UNIT TITLE:_____spinning compass part 2_____

```
;    stop
  ]
  rotate
  flip
  tick
end

to rotate
  let angular-acceleration (mag-force-constant * sum
    [
      sum
      [
        20 * (sin subtract-headings towards hub towards myself) / (distance myself) ^ 3
      ] of magnets
    ] of metals)
  ask hub
  [
    set angular-velocity (angular-velocity * friction + angular-acceleration  )
    right angular-velocity
    ifelse angular-velocity > 0
    [
      set direction -1
    ]
    [
      set direction 1
    ]
    set total_rotation total_rotation - angular-velocity
    if abs total_rotation > max_rotation
    [
     set max_rotation abs total_rotation
     set number_mag count magnets
     set number_metal count metals
    ]
  ]
end

to flip
  if direction != direction_check
  [
    if multi
    [
```

Teacher_harry henderson
Grade.Quarter.Unit_______________
Day(s)__1____
Dates:_ TBD _

RRPS SECONDARY MATH UNIT/LESSON PLAN TEMPLATE 2013-2014

UNIT TITLE:_____spinning compass part 2_____

<table>
<tr><td></td><td></td><td><pre>   set finished? true
   ]
  ]
  set direction_check direction
end</pre></td><td></td></tr>
<tr><td>Phases can be done on different days if class periods are close to an hour or back to back if longer time slot is allowed

10-20 minutes</td><td>Whole group</td><td>Explicit Instruction: Core Lesson(s)
Develop differential equations, Review agent based model of the code and Finite difference, Euler's method:

$value_{new} = value_{old} + time\ step * differential\ equation$

TO:

I will present the Euler forms as:

$\theta_{new} = \theta_{new} + \Delta t\, \omega_{old}$

theta_new = theta_new + time_step * omega_old;

WITH:
We will now compute torque by expanding the distance, d, using the law of cosines from the review above

$\omega_{new} = \omega_{old} + \Delta t \frac{r_1 \sin((\alpha - \theta_{new}))}{r_0 (r_0^2 + r_1^2 - 2 r_0 r_1 \cos((\alpha - \theta_{new})))^{(3/2)}}$

omega_new [n] = omega_old [n] + time_step * 2 * (radius_1 * sin (alpha - theta_new) /
(radius_0 * pow((pow(radius_0, 2) + pow(radius_1, 2) -
2 * radius_0 * radius_1 * cos(alpha - theta_new)), 1.5)));</td><td></td></tr>
<tr><td>20-30 minutes</td><td>Small Group, Pairs, or Independent</td><td>Guided and Independent Practice Opportunities
BY:
On your own now take the Euler forms with more than one metal piece, students should arrive at:</td><td></td></tr>
</table>

$$\theta_{new} = \theta_{old} + \sum_{n=1}^{u} \Delta t\, \omega_{old\, n}$$

```
for (int n=0; n<3; n++)
{
      theta_new = theta_new + time_step * omega_old [n];
}
```

$$\omega_{new} = \omega_{old} + \sum_{n=1}^{u} \Delta t \frac{r_1 \sin((\alpha_n - \theta_{new}))}{r_0 (r_0^2 + r_1^2 - 2 r_0 r_1 \cos((\alpha_n - \theta_{new})))^{(3/2)}}$$

```
for (int n=0; n<3; n++)
{
      omega_new [n] = omega_old [n] +
      time_step * 2 * ( radius_1 * sin ( alpha [n] - theta_new ) /
      ( radius_0 * pow( ( pow(radius_0, 2) + pow(radius_1, 2) -
      2 * radius_0 * radius_1 * cos( alpha [n] - theta_new )), 1.5)));
}
```

5-15 minutes	Whole group	**Closure for Core Lesson(s) and Independent Practice**

students can also add friction and the complete code will look like:

```
theta_new = 0;
for (int n=0; n<3; n++)
{
      theta_new = theta_new + time_step * omega_old [n];
}
theta_new = theta_old + theta_new;
for (int n=0; n<3; n++)
{
      omega_new [n] = omega_old [n] +
      time_step * 2 * ( radius_1 * sin ( alpha [n] - theta_new ) /
      ( radius_0 * pow( ( pow(radius_0, 2) + pow(radius_1, 2) -
      2 * radius_0 * radius_1 * cos( alpha [n] - theta_new )), 1.5)));

      if (omega_new [n] > 0)
```

Teacher_harry henderson
Grade.Quarter.Unit_______________
Day(s)__1____
Dates:_ TBD _

RRPS SECONDARY MATH UNIT/LESSON PLAN TEMPLATE 2013-2014

UNIT TITLE:_____spinning compass part 2_____

```
            {
                omega_new [n] = omega_new [n] -
                time_step * 2 * (friction * fabs (omega_old [n] / pow(radius_0, 2)));
            }
            if (omega_new [n] < 0)
            {
                omega_new [n] = omega_new [n] +
                time_step * 2 * (friction * fabs (omega_old [n] / pow(radius_0, 2)));
            }
            omega_old [n] = omega_new [n];
        }
        theta_old = theta_new;
```

Stage 3 – Individual Classroom Plan (Teacher)

Differentiation & Strategies to Individualize Unit/Lesson	Accommodations
This lesson is to be used with grades 10-12, depending on student population adjustments will need to be made to make the project level appropriate, these accommodations can also be used to make accommodations within grade level differentiation [e.g. IEP, gifted, etc…]	For lower levels provide prebuilt equation sheets/code and allow students to work in groups. If the lesson needs to be shorter this option can be used with upper level students also.

Stage 4 – Two Part Reflection (Teacher and PLC)

Which one of the "shifts" did this unit/lesson best reflect? Explain how.	Choose one of the following questions to answer or create your own: *1) How did this unit/lesson support 21st Century Skills? 2) How did this unit/lesson reflect academic rigor? 3) How did this unit/lesson cognitively engage students? 4) How did this unit/lesson engage students in collaborative learning and enhance their collaborative skills?*
<u>X Rigor:</u> **In major topics pursue conceptual understanding, procedural skill and fluency, and application with equal intensity.** This project covers several key aspects of mathematics, ties in the scientific method and wraps everything together in a comprehensive experiment from start to finish including analysis all with a hands on real world application.	This lesson seamlessly combines STEM together. Projects that incorporate math and science together not only teach both math and science they demonstrate the interconnections between the two and this yields a sum greater than the parts. Since this is a team project students bond over the importance of combined math and science project promoting the acceptance of STEM in a public fashion that is mostly absent in our society today.