Asynchronous Discovery Activity Learning to Fly with the Wind

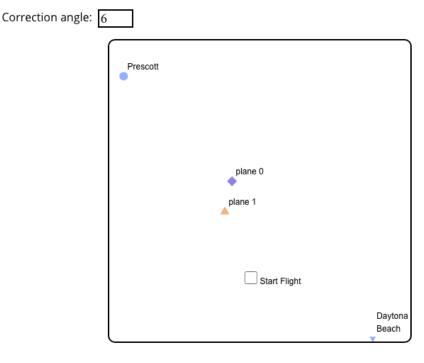
I would like to create and then study student interaction with an autonomous, discovery based Doenet activity. Generally, the activity would walk students through multiple stages of solving a traditional precalculus problem:

A plane is going from city A to city B. There is a wind of X mph blowing from XX direction. What direction does the pilot need to make to adjust for the wind?

The Doenet activity would provide feedback as the student investigates what additional information they may need to solve the question, potential concepts they could rely on, application of one concept, and then validation with a visual of a plane flying between the two cities (with loop to return if they weren't correct). After creating the activity, it would be included as a mandatory assignment within a course I would teach in Spring 2024. While data would be collected at this point, money from the mini-grant would be used to solicit students to participate in a survey on their experiences with the activity and record their thoughts as they interact with the activity.

See template Doenet activity to build on here:

A pilot is flying from Daytona Beach to Prescott. The two cities are 1874 miles away from one another and the angle from Daytona Beach to Prescott is W28N. At takeoff, there is a 12 mph wind blowing N37E that blows constantly throughout the entire flight 4-hour flight. At what angle should the pilot leave Daytona Beach to fly in a straight line and land in Prescott?



Doenet Code for Activity

```
<!-- Interactive for Flight Question -->
```

A pilot is flying from Daytona Beach to Prescott. The two cities are 1874 miles away from one another and the angle from Daytona Beach to Prescott is W28N. At takeoff, there is a 12 mph wind blowing N37E that blows constantly throughout the entire flight 4-hour flight. At what angle should the pilot leave Daytona Beach to fly in a straight line and land in Prescott?

```
<math name='wind mag deg' hide>
<!-- (MAG, DEGREE) -->
(12, 90-37)
<vector name='wind vec' hide>
<!-- (x comp, y comp) -->
($wind_mag_deg.x * cos(2 * pi * $wind_mag_deg.y / 360), $wind mag_deg.x * sin(2 * pi *
$wind_mag_deg.y / 360))
</vector>
Correction angle: <mathInput name='correction ang'></mathInput>
<math name='plane mag deg' hide>
<!-- (MAG, DEGREE) -->
(468.5, 180-28)
<math name='plane mag corr deg' hide>
<!-- (MAG, DEGREE) -->
($plane mag deg.x, $plane mag deg.y + $correction ang)
<vector name='plane org vec' hide>
<!-- (x comp, y comp) -->
($plane_mag_deg.x * cos(2 * pi * $plane_mag_deg.y / 360), $plane_mag_deg.x * sin(2 * pi *
$plane mag deg.y / 360))
</vector>
<vector name='plane corr vec' hide>
<!-- (x comp, y comp) -->
($plane_mag_corr_deg.x * cos(2 * pi * $plane_mag_corr_deg.y / 360),
$plane mag corr deg.x * sin(2 * pi * $plane mag corr deg.y / 360))
</vector>
```

```
<math name='plane loc' hide>
 <!-- (X_loc, Y_loc) -->
(500, 0)
<math name='DB loc' hide>
(750, 0)
<math name='PS loc' hide>
($DB loc.x + 1874 * cos(2 * pi * $plane mag deg.y / 360), 1874 * sin(2 * pi *
$plane mag deg.y / 360))
<vector name='resultant org vec' hide>
$wind vec + $plane org vec
</vector>
<vector name='resultant corr vec' hide>
 $wind vec + $plane corr vec
</vector>
<graph size='medium' name='FlightPath' showNavigation='False' showBorder='True'</pre>
displayXAxis='False' displayYAxis='False' xMin='-1000' xMax='1000' yMin='-10' yMax='1000'>
 line hide>
 y=x
 </line>
 <point name='Daytona Beach' labelIsName styleNumber='1'> $DB loc </point>
 <point name='Prescott' labelIsName styleNumber='1'> $PS loc </point>
 <point name='plane 0' labelIsName styleNumber='4'>
  ($time * $resultant org vec.x + $DB loc.x, $time * $resultant org vec.y)
 </point>
 <point name='plane 1' labelIsName styleNumber='3'>
 ($time * $resultant corr vec.x + $DB loc.x, $time * $resultant corr vec.y)
 </point>
 <booleanInput name='on' anchor='(75, 200)'>
  <label>Start Flight</label>
 </booleanInput>
</graph>
<number name='time' hide>
 0
```

</number>

 $< animate From Sequence\ target='time'\ from='0'\ to='4'\ step='0.1'\ animation Interval='10'\ animation On='\$on'\ animation Mode='increase\ once'/>$