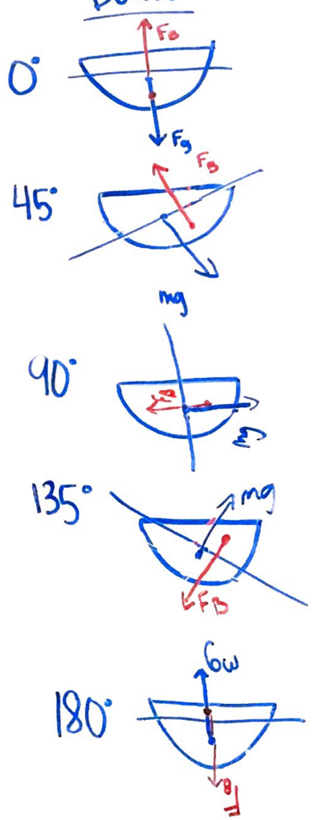
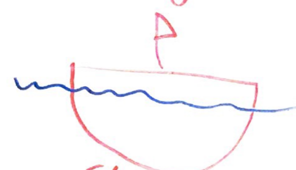
Boats



Confused:

- Mathematica not working !! ☹

Stable	Unstable
 swing set	 water bottle on edge of table
 water in a below-ground pool	 balancing a pencil on your finger
 laying	 ← colin ☺ standing
 stable	 unstable

Day 6

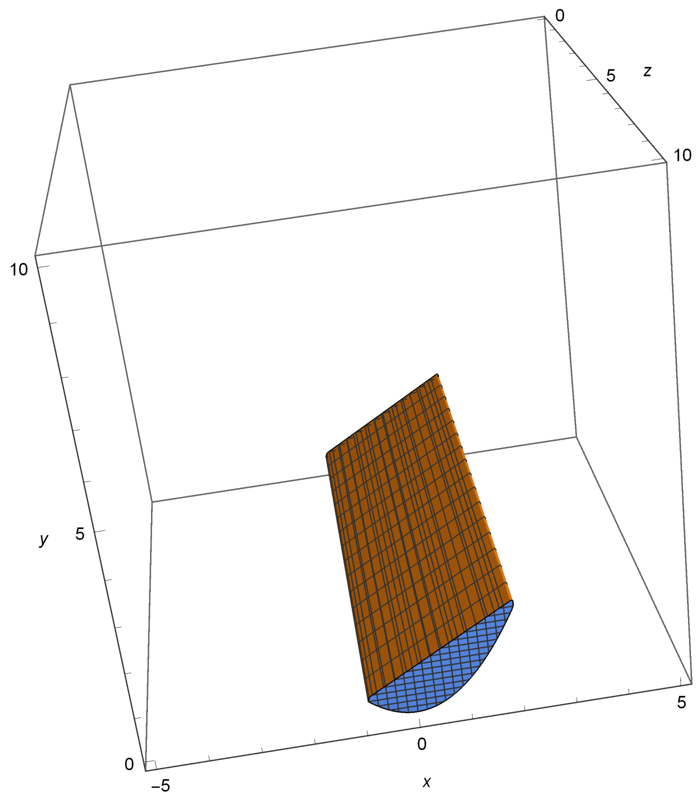
Defining Variables

In[768]:=
```
Clear[f, density, xmin1, xmax1, ymin1, ymax1, zmin1, zmax1, xballast, yballast,
 ballastmass, θ, boat, water, submerged, boatmass, totalmass, xcomboat, ycomboat,
 xcom, ycom, displacement, waterline, b, d, draft, cob, xcob, ycob, x, y, z]
Clear["Global`*"]
f = 2 * ((x / 2) ^2);
density = 300;
ballastmass = 5000;
xmin1 = -2;
xmax1 = 2;
ymin1 = 0;
ymax1 = 2;
zmin1 = 0;
zmax1 = 10;
xballast = 0;
yballast = 0;
θ = 30 Degree;
```

3D Plot of Boat’s submerged region (ESTIMATE. NOT ACTUALLY)

In[782]:=
```
RegionPlot3D[zmin1 ≤ z ≤ zmax1 && f ≤ y ≤ Tan[θ] * x + 1 && xmin1 ≤ x ≤ xmax1, {x, -5, 5},
 {y, 0, 10}, {z, zmin1, zmax1}, PlotPoints → 100, Axes → True, AxesLabel → {x, y, z}]
```

Out[782]=



Defining Boat region

In[783]:=
```
boat = ImplicitRegion[f < y < ymax1 && xmin1 < x < xmax1 && ymin1 < y < ymax1, {x, y}]
```

Out[783]= ImplicitRegion$\left[\frac{x^2}{2} < y < 2 \&\& -2 < x < 2 \&\& 0 < y < 2, \{x, y\}\right]$

Defining water region

In[784]:=
```
water =
 ImplicitRegion[ymin1 < y < Tan[θ] * x + b && xmin1 < x < xmax1 && ymin1 < y < ymax1, {x, y}]
```

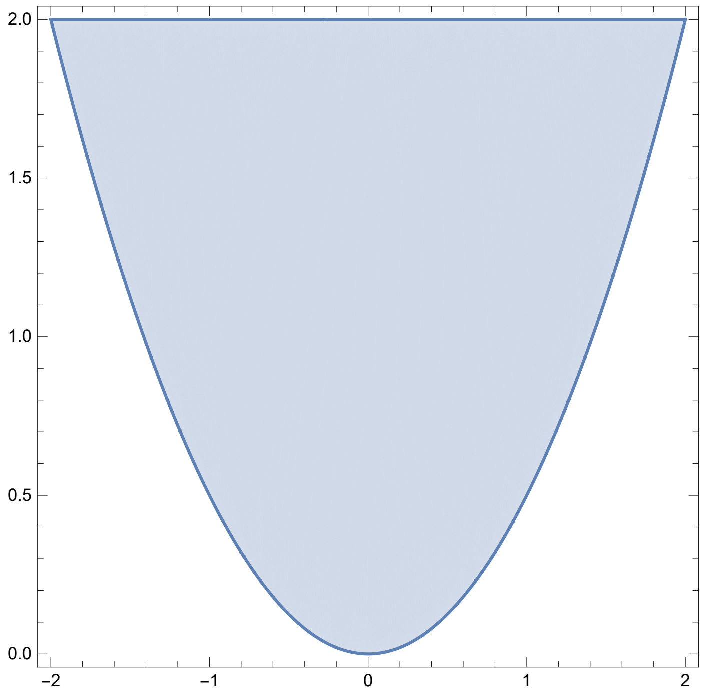
Out[784]= ImplicitRegion$\left[0 < y < b + \frac{x}{\sqrt{3}} \&\& -2 < x < 2 \&\& 0 < y < 2, \{x, y\}\right]$

Defining Submerged region of boat as intersection of boat and water

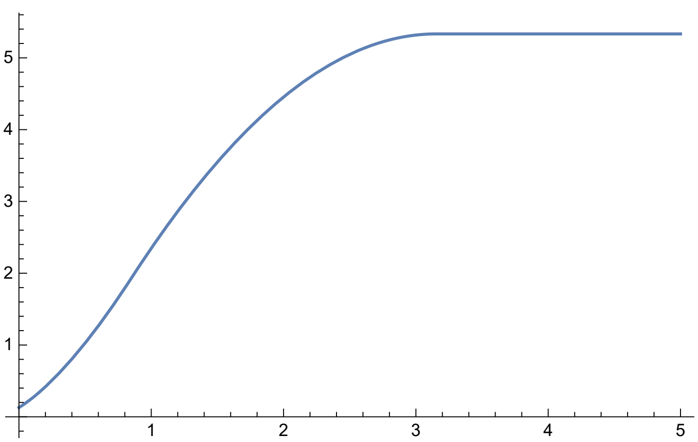
In[785]:=
```
submerged = RegionIntersection[boat, water]
RegionPlot[submerged /. b → 5]
Plot[RegionMeasure[submerged /. b → bb], {bb, 0, 5}]
```

Out[785]= ImplicitRegion$\left[-2 < x < 2 \,\&\&\, 0 < y < 2 \,\&\&\, 0 < y < b + \frac{x}{\sqrt{3}} \,\&\&\, \frac{x^2}{2} < y < 2, \{x, y\}\right]$

Out[786]=



Out[787]=



Figuring out total mass of boat by integration and adding ballastmass

∈ sign is escape, e, l, escape

In[788]:=
```
boatmass = 10 * Integrate[density, {x, y} ∈ boat]
```

Out[788]= 16 000

In[789]:=
```
totalmass = boatmass + ballastmass
```
Out[789]= 21 000

Finding COMs

In[790]:=
```
xcomboat = (1 / boatmass) * 10 * Integrate[x * density, {x, y} ∈ boat]
```
Out[790]= 0

Just boat

In[791]:=
```
ycomboat = N[(1 / boatmass) * 10 * Integrate[y * density, {x, y} ∈ boat]]
```
Out[791]= 1.2

COM of combined boat and ballast: 2 ways, second one takes into account ballast x or y coordinate

In[792]:=
```
xcom = N[(1 / totalmass) * 10 * Integrate[x * density, {x, y} ∈ boat]]
xcom = 1 / (total) * (xcomboat * boatmass + xballast * ballastmass)
```
Out[792]= 0.

Out[793]= 0

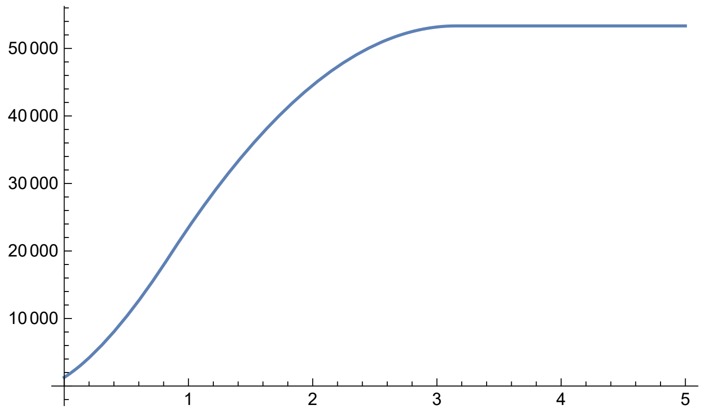
In[794]:=
```
ycom = N[(1 / totalmass) * 10 * Integrate[y * density, {x, y} ∈ boat]]
ycom = 1 / (totalmass) * (ycomboat * boatmass + yballast * ballastmass)
```
Out[794]= 0.914286

Out[795]= 0.914286

Finding Buoyant force with b as a variable

In[796]:=
```
displacement = N[10 * Integrate[1000, {x, y} ∈ submerged]]
Plot[displacement /. b → bb, {bb, 0, 5}]
```

Out[796]= $10.\ \begin{cases} 74.0741\left(1.73205\sqrt{1.+6.\,b}+10.3923\,b\sqrt{1.+6.\,b}\right) & -0.166667 < b \leq 0.845299 \\ -18.5185\left(46.5256-155.885\,b+46.7654\,b^2-3.4641\sqrt{1.+6.\,b}-20.7846\,b\sqrt{1.+6.\,b}\right) & 0.845299 < b < 1.1547 \,\|\, 1.1547 \leq b < 3.1 \\ 1000.\left(2.\left(2.+1.73205\left(-2.+b\right)\right)+0.166667\left(-8.-5.19615\left(-2.+b\right)^3\right)\right) & b == 3.1547 \\ 5333.33 & b > 3.1547 \\ 0. & \text{True} \end{cases}$

Out[797]=


Figuring out waterline based on displacement

In[829]:=
```
waterline = Solve[displacement == mass, b, Reals];
waterline /. mass → totalmass
draft = b /. %[[4]]
```

Solve: Solve was unable to solve the system with inexact coefficients. The answer was obtained by solving a corresponding exact system and numericizing the result.

Out[830]= {{b → Undefined}, {b → Undefined}, {b → Undefined}, {b → 0.909465}}

Out[831]= 0.909465

Finding Center of buoyancy

In[832]:=
```
cob = N[10 * Integrate[1000 * {x, y}, {x, y} ∈ submerged] / displacement];
xycob = cob /. b → draft
xcob = xycob[[1]]
ycob = xycob[[2]]
```

Out[833]= {0.574017, 0.809448}

Out[834]= 0.574017

Out[835]= 0.809448

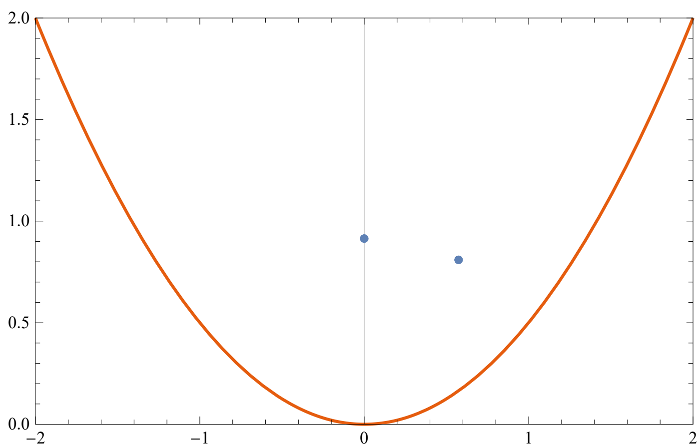
Plotting boat with waterline

In[836]:=
```
plotcob = {{xcob, ycob}}
plotcom = {{xcom, ycom}}
Show[Plot[f, {x, xmin1, xmax1}, PlotRange → {{xmin1, xmax1}, {ymin1, ymax1}},
  PlotTheme → "Scientific"], ListPlot[plotcob], ListPlot[plotcom],
 Plot[Tan[θ] + b, {x, xmin1, xmax1}, PlotRange → {{xmin1, xmax1}, {ymin1, ymax1}},
  PlotTheme → "Scientific"] ]
```

Out[836]= {{0.574017, 0.809448}}

Out[837]= {{0, 0.914286}}

Out[838]=



Calculating Righting Moment

In[839]:=
```
rmx = (xcom - xcob) * totalmass
rmy = (ycom - ycob) * totalmass
```

Out[839]= -12 054.4

Out[840]= 2201.6

Results for angles I have run it at:

30: COB = {{0.574017,0.809448}}, RM = (-12054, 2201)
45: COB = {0.771827,0.95692}, RM = (-16208, -895)