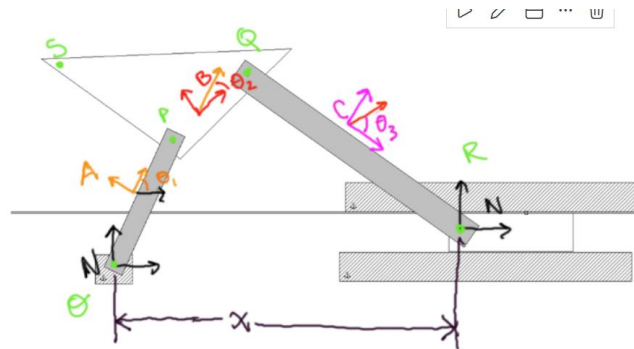
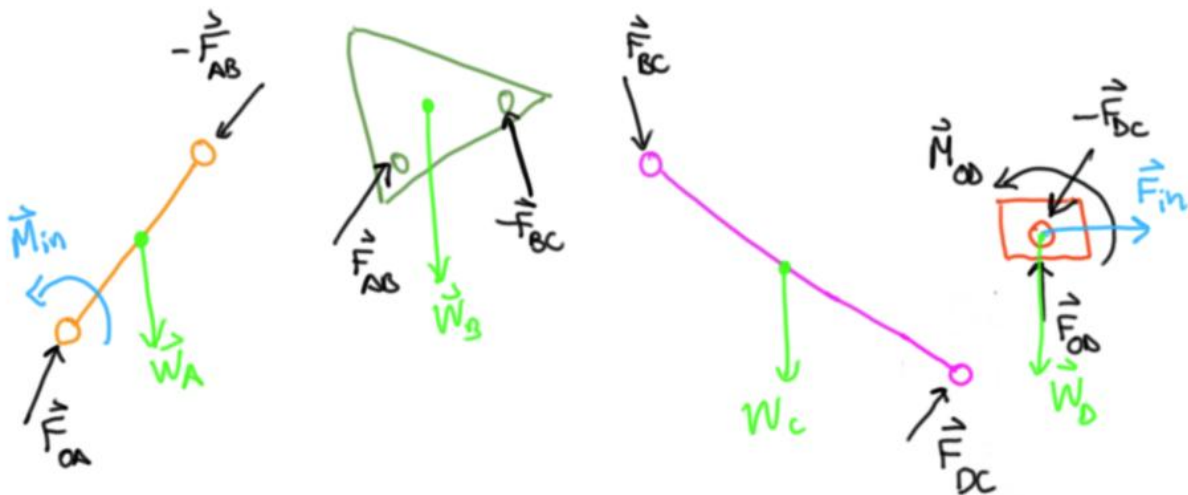


Estos son solo uno tips, porque si les comparto todo el código terminan aprendiendo menos.



Es importante hacer bien los diagramas de cuerpo libre. Observe que se aplican acción-reacción en las uniones. El momento de entrada y fuerza de entrada son conocidos en un problema de dinámica directa o incógnitas en dinámica inversa.



Un truco útil es escribir los centros de masa como expresiones vectoriales. A mí me gusta hacerlos como un diccionario, así puedo identificar más fácilmente los puntos a los que hago referencia en el código. Observe com['A'].

```
eqMA=Min*N.z+(points['O']-com['A']).cross(F0A)+
```

Una vez se tienen todas las ecuaciones de Newton-Euler (en este caso son 4 cuerpos x 3 ecuaciones = 12 ecuaciones en total), podemos pasar a generar el sistema en la forma lineal.

```
[FF,bf]=sympy.linear_eq_to_matrix(eqList,unkowns)
```

Recuerden que `linear_eq_to_matrix` nos factoriza un vector de incógnitas con la forma $Ax=b$. A `FF` y `bf` se les puede aplicar “`lambdify`” en función de variables cinemáticas conocidas en este instante del tiempo (dinámica inversa).

Recuerde que si tiene los ángulos en el tiempo, puede sacar las velocidades angulares con una derivada numérica usando `np.gradient`:

```
thetadotList=np.gradient(thetaList,tList,edge_order=2)
```