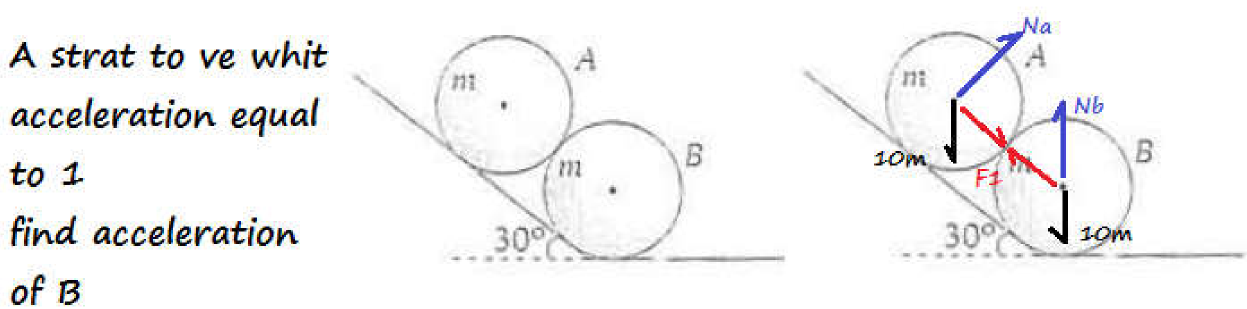
```python
In [1]: from sympy import *
        from polyclass import *
        from libaldo_math import *
        from libaldo_show import *
        from physic_lib import *
        from IPython.display import display, Math
        init_printing()
```

Please take note the following aclarations:
This library is created to help us with equations, algorithms and mainly with practical and general physics support, but does not solve problems automatically...

Medium Dynamic Mechanic Physics Problems




```python
In [2]: # creating varuables
        m,F1,Na,Nb=symbols('m F1 Na Nb',positive=True)
```

```python
In [3]: # creating A object and addinf Forces
        A=mparticle(a=1)
        A.add_forza(10*m,-pi/3)# = add weight in this case axis X is -30 grad
         with move direcction
        A.add_forza(F1,pi)
        A.add_forza(Na,pi/2)
```

```python
In [4]: # Acelleration due X res = Xres /mass
        F1=csolve(A.simple_ac()-1,F1,'F')
```

$$F = 4m$$

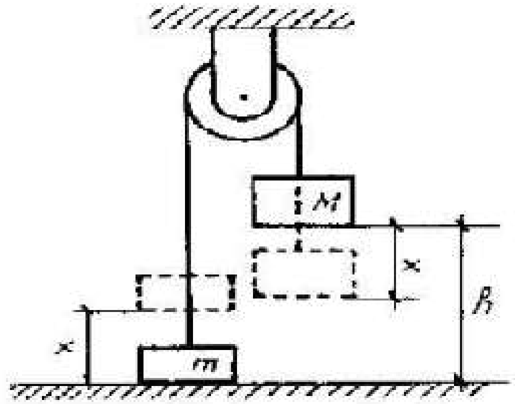
```python
In [5]: # creating B object and addinf Forces
        B=mparticle()
        B.add_forza(F1,-pi/6)# = add weight in this case axis X is -30 grad  wi
        th move direcction
        B.add_forza(Nb,pi/2)
        B.add_forza(10*m,-pi/2)
```

```python
In [6]: B.simple_ac()
```

Out[6]: $2\sqrt{3}$

The systen start from rest ,

find velocity of B when touch the floo



In [7]:
```
Mv,mV,T,g,M,m,h,Ac=symbols('Mv mV T g M  m  h Ac',positive=True)
```

In [8]:
```
A=mparticle(m=m,g=g,y1=0,y2=h,a=pi/2,ac=Ac,v1=0)
B=mparticle(m=M,g=g,y1=h,y2=0,a=-pi/2,ac=Ac,v1=0)
```

In [9]:
```
A.add_forza(m*g,-pi/2) #weight
A.add_forza(T,pi/2) # Tension
B.add_forza(M*g,pi/2) #weight
B.add_forza(T,-pi/2) # Tension
```

In [10]:
```
T1=csolve(A.simple_ac('y')-B.simple_ac('y'),T);T1
```

Out[10]: $\frac{2Mgm}{M+m}$

In [11]:
```
A.setValue(T,T1)
B.setValue(T,T1)
```

In [12]:
```
Accel=A.simple_ac('y',kope='s');Accel
```

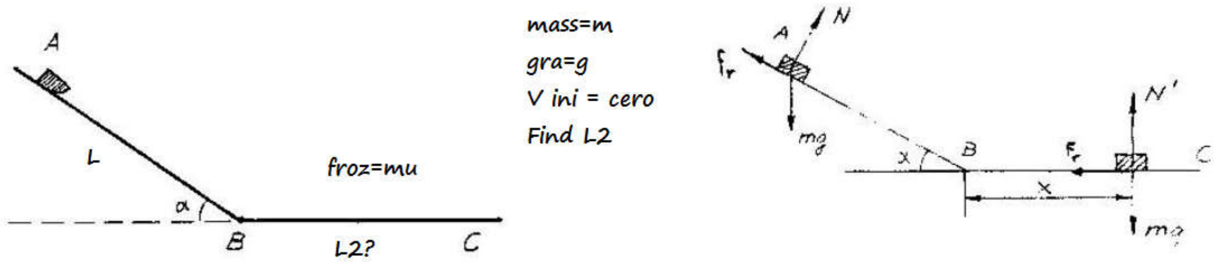
Out[12]: $\frac{g(M-m)}{M+m}$

In [13]:
```
v=rpow(2*Accel*h);v
```

Out[13]: $\sqrt{2}\sqrt{h}\sqrt{\frac{g(M-m)}{M+m}}$

In [14]:
```
#Method 2 , total energia equal cero
csolve(A.energia()+B.energia(),v2,korden=1)
```

Out[14]: $\sqrt{2}\sqrt{\frac{gh(M-m)}{M+m}}$



In [15]:
```
alpha,L,m,g,mu,N1,fr,L2=symbols('alpha L m g mu N1 fr L2',positive=True
)
```

In [16]:
```
#  take note that  we take  AB like axis X
A=mparticle(v1=0,m=m,g=g,x1=0,x2=L,y1=0,y2=0)
```

In [17]:
```
A.add_forza(m*g,-pi/2+alpha) # adding weight
```

In [18]:
```
A.setValue(fr,-A.y_res()*mu) # setting fr = Y(weight)*mu
```

In [19]:
```
A.add_forza(fr,pi)# adding fr
```

In [20]:
```
W1=A.work_due_forza('x',kope='f');W1 # Work1 from A 2 B = Fx*L
```

Out[20]: $-Lgm(\mu\cos(\alpha) - \sin(\alpha))$

In [21]:
```
L2=symbols('L2')
B=mparticle(m=m,g=g,x1=0,x2=L2,y1=0,y2=0)
```

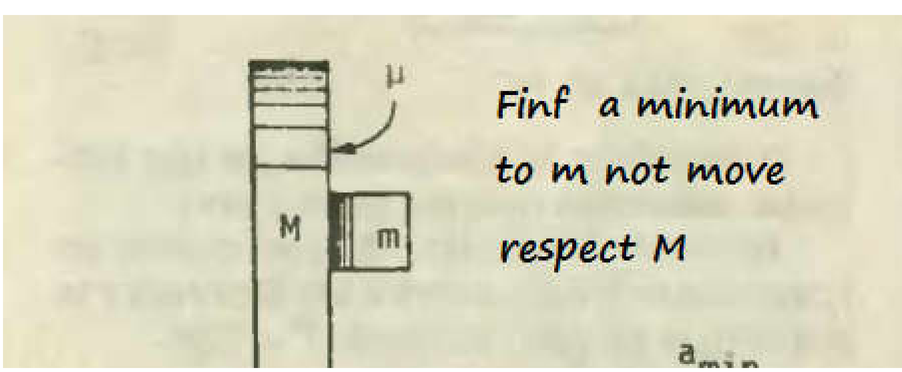
In [22]:
```
B.add_forza(m*g*mu,pi)
```

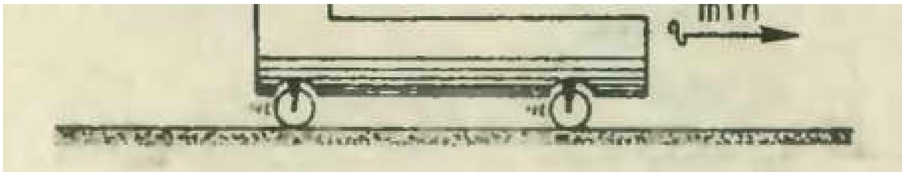
In [23]:
```
W2=B.work_due_forza('x',kope='f');W2 # Work= Fx*(x2-x1)
```

Out[23]: $-L_2 g m \mu$

In [24]:
```
# Total Work = cero because Vel ini and Vel final equal cero
L2=csolve(W1+W2,L2,'L2') # Find L2
```

$$L2 = \frac{L(-\mu\cos(\alpha) + \sin(\alpha))}{\mu}$$





In [25]:
```
M,m,g,N1,Fr=symbols('M m g N1 Fr',positive=True)
```

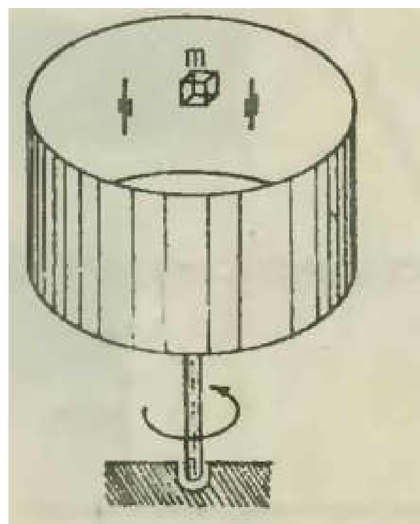
In [26]:
```
A=mparticle(m=m)
A.add_forza(m*g,-pi/2)
A.add_forza(N1,0)
A.add_forza(8*N1/10,pi/2)
```

In [27]:
```
A.setValue(N1,csolve(A.y_res(),N1))
```

In [28]:
```
A.simple_ac()
```

Out[28]: $\frac{5g}{4}$

Find ANgular velocity if cooef fricction = 0.25, g=10, radius = 10m



Tips ...... from now we not include algorithms like find X if 2+X= 50

In [29]:
```
m, w=symbols('m w',positive=True)
```

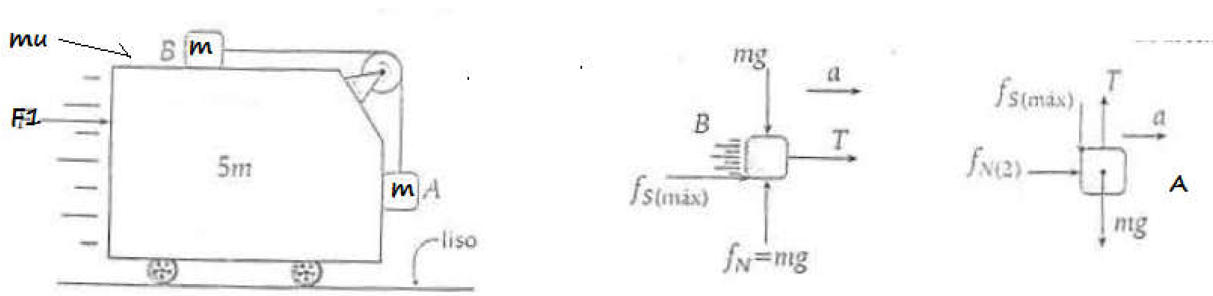
In [30]:
```
P=mparticle(m=m,g=10,r=10,w=w)
```

In [31]:
```
P.simple_Fcentripeta()
```

Out[31]: $10mw^2$

In [32]:
```
# F centripeta = Normal = Frozz/mu = m*s/mu
W=csolve(P.simple_Fcentripeta()-10*m*4,w,'W_a',kpositive=True)
```

$W = 2$



Find all possible if the system make that A start to up when applied F1

In [34]:
```
F1,T,m,g,a,fr,fr2,mu,N1=symbols('F1 T m g a fr fr2 mu N1',positive=True
)
```

In [35]:
```
# in A ...first Eq F = mass * accelll
e1=polyclass(F1,a*(5*m+m+m))
```

In [36]:
```
# in B ...  second and third Eqs
e2=polyclass(fr,m*g*mu)
e3=polyclass(T,m*a-fr)
```

In [37]:
```
# in A last Eqss
e4=polyclass(N1,m*a)
e5=polyclass(fr2,N1*mu)
e6=polyclass(fr2+m*g,T )
```

In [38]:
```
vec=[e1,e2,e3,e4,e5,e6]
P=Mpolyclass(vec);P.showQ()
```

$$F_1 = 7am$$
$$fr = gm\mu$$
$$T = am - fr$$
$$N_1 = am$$
$$fr_2 = N_1\mu$$
$$fr_2 + gm = T$$

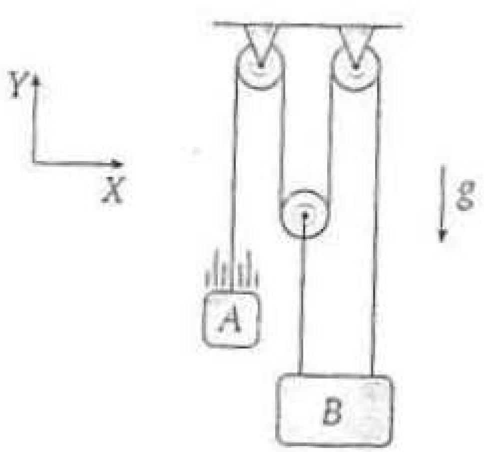
In [39]:
```
# in Msolve you need applied yours intuitions and neerd algebra  knowle
dge to solve
P.Msolve([N1,fr2,T,fr,a],[3,4,2,1,5])
```

$$F_1 = -\frac{7gm(\mu+1)}{\mu-1}$$
$$fr = gm\mu$$
$$T = -gm\mu - \frac{gm(\mu+1)}{\mu-1}$$
$$N_1 = -\frac{gm(\mu+1)}{}$$

$$\mu - 1$$

$$fr_2 = -\frac{gm\mu(\mu+1)}{\mu-1}$$

$$a = -\frac{g(\mu+1)}{\mu-1}$$



the system is
abandoned from rest

In the system :
5mA=3mB
find acceleration of A

We solve whit pure Algebra with polyclass and Mpolyclass

In [40]:
```
m,g,T,Aa,Ab=symbols('m g T Aa Ab',positive=True)
```

In [41]:
```
#relation between mass
ma=m
mb=5*m/3
```

In [42]:
```
# Analisis A
e1=polyclass(T-ma*g,ma*Aa);e1.s()
```

Out[42]: $T - gm = Aam$

In [43]:
```
# for geometri
Ab=Aa/3
```

In [44]:
```
# Analisis B
e2=polyclass(mb*g-3*T,mb*Ab);e2.s()
```

Out[44]: $-3T + \frac{5gm}{3} = \frac{5Aam}{9}$

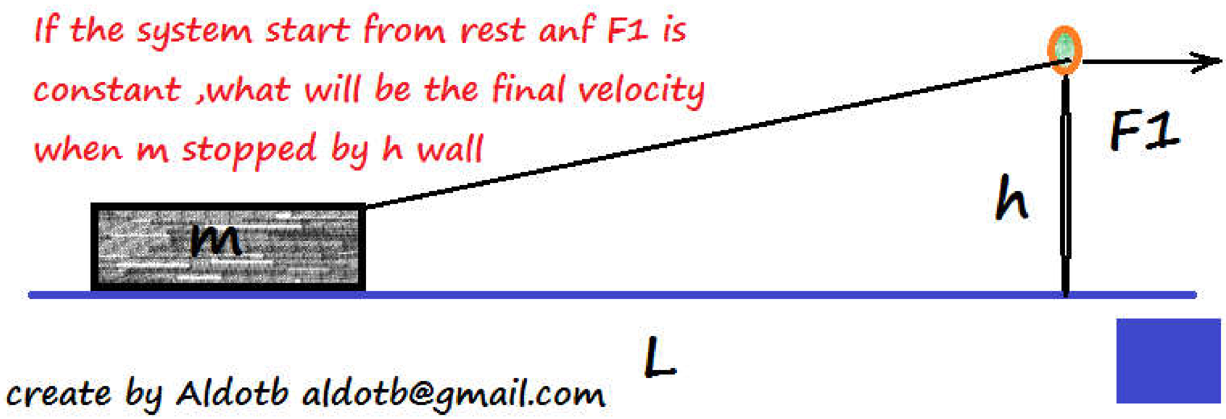
In [45]:
```
vec=[e1,e2]
P=Mpolyclass(vec);P.s()
```

$$T - gm = Aam$$
$$-3T + \frac{5gm}{3} = \frac{5Aam}{9}$$

In [46]:
```
P.Msolve([T,Aa],[0,1])
```

$$T = \frac{5gm}{8}$$

$$Aa = -\frac{3g}{8}$$



In [47]:
```
L,m,g,F1,alpha,x,h,V2=symbols('L m g F1 alpha x h V2',positive=True)
```

In [48]:
```
hipo=rpow(kpow(h,2)+kpow(L-x,2),2)
hipo
```

Out[48]:

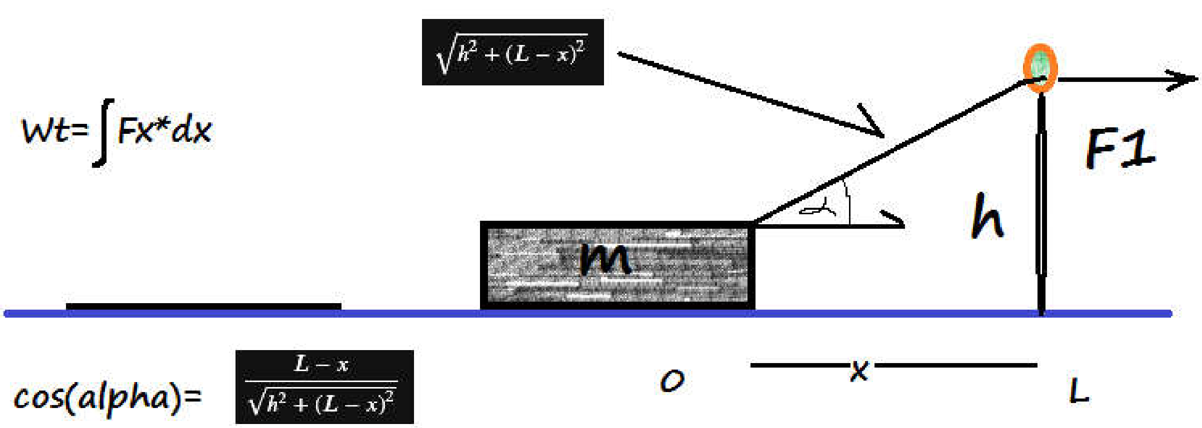
$$\sqrt{h^2 + (L - x)^2}$$

In [49]:
```
kcos=(L-x)/hipo
kcos
```

Out[49]:

$$\frac{L - x}{\sqrt{h^2 + (L - x)^2}}$$

In [50]:
```
W=integrate(F1*kcos,(x,0,L)) # Work = sum(F*dx)
```



In [51]:
```
P=mparticle(m=m,x1=0,y1=0,x2=L,y2=0,v1=0,v2=V2)
```

In [52]:
```
P.energia()
```

Out[52]:

$$\frac{V_2^2 m}{2}$$

In [53]:
```
# Work Total = Energy Total, whit this find V final
Vf=csolve(P.energia()-W,V2,'V_f',korden=0)
```

$$V_f = \sqrt{2}\sqrt{\frac{F_1\left(-h+\sqrt{L^2+h^2}\right)}{m}}$$