

Know Values

```
unit = symunit;
mu = sym(0.21);
Wdriver = 6.57 * unit.N;
Wslider = 23.45 * unit.N;
Theta = sym(45) * unit.deg;
SpringForce = sym(38) * unit.N / unit.mm;
SpringDeflection = 6:26;
SpringDeflection = SpringDeflection * unit.mm;
```

Variables

```
syms Pforce Nwall Nangle Fangle Fwall Nbase Fbase Fspring
Fwall = mu * Nwall;
Fangle = mu * Nangle;
Fbase = mu * Nbase;
```

```
Fspring = SpringForce * SpringDeflection;
```

Slider

Sum of Forces in x-direction and y-direction

```
equ1 = -Fspring - Fbase + Nangle * cos(Theta) - Fangle * cos(Theta) == 0;
equ2 = -Wslider + Nbase - Nangle * sin(Theta) - Fangle * sin(Theta) == 0;

for index = 2:21
equ2(index) = -Wslider + Nbase - Nangle * sin(Theta) - Fangle * sin(Theta) == 0;
end

[NbaseSolutions, NangleSolutions] = solve(equ1(1), equ2(1), Nbase, Nangle);
for index = 2:21
[NbaseSolutions(index), NangleSolutions(index)] = solve(equ1(index), equ2(index), Nbase, Nangle);
end
NbaseSolutions

NbaseSolutions =
( 2944055 / 5359 N 3403855 / 5359 N 167985 / 233 N 4323455 / 5359 N 4783255 / 5359 N 5243055 / 5359 N 5702855 / 5359 N 6162655 / 5359 N 6622455 / 5359 N 7082255 / 5359 N 7542055 / 5359 N 8001855 / 5359 N 8461655 / 5359 N 8921455 / 5359 N 9381255 / 5359 N 9841055 / 5359 N 10300855 / 5359 N 10760655 / 5359 N )
```

NangleSolutions

```
NangleSolutions =
( 2329245 / 5359 sqrt(2) N 2709245 / 5359 sqrt(2) N 134315 / 233 sqrt(2) N 3469245 / 5359 sqrt(2) N 3849245 / 5359 sqrt(2) N 4229245 / 5359 sqrt(2) N 4609245 / 5359 sqrt(2) N 4989245 / 5359 sqrt(2) N 5369245 / 5359 sqrt(2) N 5749245 / 5359 sqrt(2) N 6129245 / 5359 sqrt(2) N 6509245 / 5359 sqrt(2) N 6889245 / 5359 sqrt(2) N 7269245 / 5359 sqrt(2) N )
```

Driver

Sum of Forces in x-direction

```
equ3 = solve(Nwall - Nangle * cos(Theta) + Fangle * cos(Theta) == 0,Nwall);
```

Nwall Values

```
NwallSolutions = subs(equ3, Nangle, NangleSolutions)

NwallSolutions =
( 36802071 / 107180 N 42806071 / 107180 N 2122177 / 4660 N 54814071 / 107180 N 60818071 / 107180 N 66822071 / 107180 N 72826071 / 107180 N 78830071 / 107180 N 84834071 / 107180 N 90838071 / 107180 N 96842071 / 107180 N 102846071 / 107180 N 108850071 / 107180 N 114854071 / 107180 N 120858071 / 107180 N 126862071 / 107180 N 132866071 / 107180 N )
```

Sum of Forces in x-direction

```
equ4 = solve(-Pforce - Wdriver + Fwall + Nangle * sin(Theta) + Fangle * sin(Theta), Pforce);
```

Pforce values

```
PforceSolutions = subs(equ4, {Nwall, Nangle}, {NwallSolutions, NangleSolutions})

PforceSolutions =
( 6339199131 / 10718000 N 7384883131 / 10718000 N 366546397 / 466000 N 9476251131 / 10718000 N 10521935131 / 10718000 N 11567619131 / 10718000 N 12613303131 / 10718000 N 13658987131 / 10718000 N 14704671131 / 10718000 N 15750355131 / 10718000 N 16796039131 / 10718000 N 17841723131 / 10718000 N 18887407131 / 10718000 N 19933091131 / 10718000 N )
```

```
PKgfSolutions = unitConvert(PforceSolutions, unit.kgf)
```

```
PKgfSolutions =
( 63391991310 / 1051076747 kgf 73848831310 / 1051076747 kgf 523637710 / 6528427 kgf 94762511310 / 1051076747 kgf 105219351310 / 1051076747 kgf 115676191310 / 1051076747 kgf 126133031310 / 1051076747 kgf 136589871310 / 1051076747 kgf 147046711310 / 1051076747 kgf 22500507330 / 150153821 kgf 167960391310 / 1051076747 kgf 178417231310 / 1051076747 kgf 188874071310 / 1051076747 kgf 199330911310 / 1051076747 kgf )
```

```
vpa(PforceSolutions, 6)

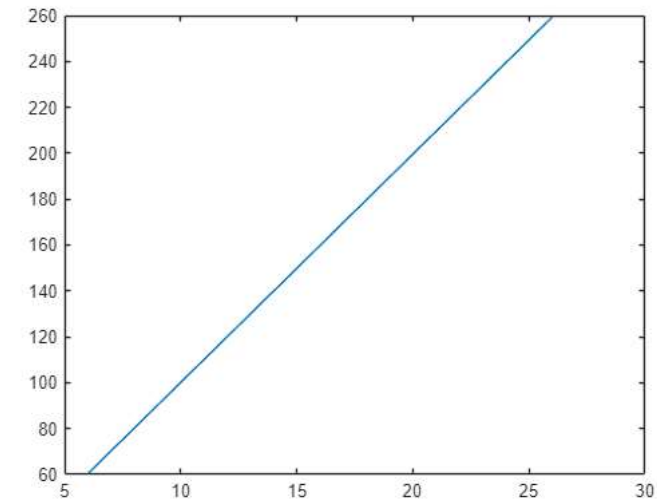
ans = (591.454 N 689.017 N 786.58 N 884.144 N 981.707 N 1079.27 N 1176.83 N 1274.4 N 1371.96 N 1469.52 N 1567.09 N 1664.65 N 1762.21 N 1859.78 N 1957.34 N 2054.9 N 2152.47 N 2250.03 N 2347.59 N 2447.11 N)
```

```
vpa(PKgfSolutions, 5)

ans = (60.311 kgf 70.26 kgf 80.209 kgf 90.158 kgf 100.11 kgf 110.05 kgf 120.0 kgf 129.95 kgf 139.9 kgf 149.85 kgf 159.8 kgf 169.75 kgf 179.7 kgf 189.64 kgf 199.59 kgf 209.54 kgf 219.49 kgf 229.44 kgf 239.39 kgf 249.34 kgf)
```

Plot of Spring Deflection vs Press Force Requirements

```
plot(separateUnits(SpringDeflection), separateUnits(PKgfsolutions))
```



```
test = fit(separateUnits(SpringDeflection)', separateUnits(PKgfsolutions)', 'poly1')
```

Warning: Converting X to matrix of double.
Warning: Converting Y to vector of double.

```
test =  
Linear model Poly1:  
test(x) = p1*x + p2  
Coefficients (with 95% confidence bounds):  
p1 = 9.949 (9.949, 9.949)  
p2 = 0.6193 (0.6193, 0.6193)
```

```
plot(test)
```

