

-----  
Nesta aula: operações com fracções  
-----

```
#-----  
# Operacoes com fraccoes  
#  
# - representar uma fraccao  
# - normalizar uma fraccao  
# - somar, subtrair multiplicar, dividir fraccoes  
#  
# --> funcoes puras  
#-----  
  
def mdc(m,n):  
    "mdc; assume-se m,n>=0, m>0 ou n>0"  
    if n==0:  
        return m  
    return mdc(n,m%n)  
  
def norma(f):  
    r=nsinal(f)  
    m=mdc(abs(r[0]),r[1])  
    r[0]=r[0]/m  
    r[1]=r[1]/m  
    return r
```

```

def nsinal(f):
    " denominador positivo"
    if f[1]<0:
        f[0]=-f[0]
        f[1]=-f[1]
    return f

def copia(f):
    return [f[0],f[1]] # nova co'pia

def soma(f,g):
    "soma de fraccoes"
    d=f[1]*g[1]
    n=(f[0]*g[1]+g[0]*f[1])
    return norma([n,d])

def mult(f,g):
    "produto de fraccoes"
    d=f[1]*g[1]
    n=(f[0]*g[1]+g[0]*f[1])
    return norma([f[0]*g[0],f[1]*g[1]])

def div(f,g):
    return mult(f,[g[1],g[0]])

```

```

# Exercício: subtrai soma(f,escalar(-1,g))
# Usar o produto escalar esc(a,fraccao)

def esc(a,f):
    return norma([a*f[0],f[1]])

def sub(f,g):
    return norma(soma(f,esc(-1,g)))

#---- testes -----
f=[1,6]
g=[2,6]
h=[2,-6]

print "norma",h, norma(h)
print "soma", f,g,soma(f,g)
print "sub", f,g,sub(f,g)
print "sub", f,f,sub(f,g)
print "prod", f,g,mult(f,g)
print "div", f,g,div(f,g)

```