

```

print "----- 1a -----"

#  0 1 2 3 4 5 6   len(a)=7
#  ^ ^ ^           12 = (7-2)/2 = 2
#  (0<->6), (1<->5),...
def invert(a):
    n=len(a)
    limite=(n-2)/2
    for i in range(limite+1):
        (a[i],a[n-1-i]) = (a[n-1-i],a[i])
    return a

for a in [[1,2,3],[5,6],[]]:
    print a, invert(a)
    print "a=",a

print

print "----- 1b -----"
def invert1(a):
    r=[]
    for elem in a:
        r = [elem]+r
    return r

# Outra solução, mais eficiente:
def invert2(a):
    r=[]
    n=len(a)
    i=n-1
    while i>=0:
        r.append(a[i])
        i = i-1
    return r

for a in [[1,2,3],[5,6],[]]:
    print a, invert1(a)
    print "a=",a

print

for a in [[1,2,3],[5,6],[]]:
    print a, invert2(a)
    print "a=",a

```

```

print "----- 2 -----"
def ordenacao(a,b):
    return teste1(b) and testeii(a,b) and testeiii(a,b)

def teste1(b):
    """ testa se "b" está ordenado """
    for i in range(len(b)-1):
        if b[i]>b[i+1]:
            return False
    return True

def testeii(a,b):
    """ testa se "a" e "b" têm o mesmo comprimento """
    return len(a)==len(b)

def testeiii(a,b):
    """ testa para todo o "x" em "a":
        "x" ocorre "k" vezes em "a" => "x" ocorre "k" vezes em "b"
    """
    vistos=[]
    for e in a:
        if not e in vistos:
            vistos.append(e)
            if conta(e,a) != conta(e,b):
                return False
    return True

def conta(e,a):
    c=0
    for x in a:
        if x==e:
            c=c+1
    return c

for (a,b) in ([([2,1],[1,2]),\
               ([2,1,2],[1,1,2]),\
               ([1,2,3],[1,2,4]), ([],[ ] ) ]):
    print a,b,ordenacao(a,b)

```

```

print "----- 3 -----"

def somap(p,q):
    r=[]
    i=0
    # primeiro, até ao mínimo dos comprimentos,
    while i<len(p) and i<len(q):
        r.append(p[i]+q[i])
        i = i+1
    # depois o resto do p[]
    while i<len(p):
        r.append(p[i])
        i = i+1
    # finalmente o resto do q[]
    while i<len(q):
        r.append(q[i])
        i = i+1
    # nota: um dos "restos" é vazio
    return corta0(r)

def corta0(a):
    """ corta os zeros no fim de "a" """
    m=len(a)-1
    while m>=0 and a[m]==0:
        m=m-1
    r=[]
    for i in range(m+1):
        r.append(a[i])
    return r

for (p,q) in [( [1,2,3],[4,5]),\
               ([1,2,3],[0,-2,-3]),\
               ([1,2,3],[-1,-2,-3])]:
    print p, q, somap(p,q)

```

#=====

Nota: incluímos testes para os exercícios.

Resultados obtidos----- 1a -----

[1, 2, 3] [3, 2, 1]

a= [3, 2, 1]

[5, 6] [6, 5]

a= [6, 5]

[] []

a= []

----- 1b -----

[1, 2, 3] [3, 2, 1]

a= [1, 2, 3]

[5, 6] [6, 5]

a= [5, 6]

[] []

a= []

[1, 2, 3] [3, 2, 1]

a= [1, 2, 3]

[5, 6] [6, 5]

a= [5, 6]

[] []

a= []

----- 2 -----

[2, 1] [1, 2] True

[2, 1, 2] [1, 1, 2] False

[1, 2, 3] [1, 2, 4] False

[] [] True

----- 3 -----

[1, 2, 3] [4, 5] [5, 7, 3]

[1, 2, 3] [0, -2, -3] [1]

[1, 2, 3] [-1, -2, -3] []

>>>