



Nome:

Número:

Data:

Curso:

Capítulo 6.2 - Funções de ordem superior

Utilizando funcionais sobre listas escreva uma função de ordem superior, `conta_p`, que recebe um número inteiro positivo n e um predicado unário p , e devolve o número de inteiros positivos menores ou iguais a n que satisfazem p . A sua função deve conter apenas uma instrução, a instrução `return`. Por exemplo:

```
>>> conta_p(87, lambda x: x % 100 == 0)
0
>>> conta_p(487, lambda x: x % 100 == 0)
4
```

Solução 1:

```
def conta_p(num, pred):
    return len(filtra(pred, [x for x in range(1, num+1)]))
```




Nome:

Número:

Data:

Curso:

Capítulo 6.2 - Funções de ordem superior

Utilizando funcionais sobre listas, escreva a função `concatena`, que recebe um predicado `pred` e uma lista `l` e concatena todas as listas em `l` que satisfazem o predicado. Pode assumir que a lista `l` não é vazia e só tem listas como elementos. A sua função deve conter apenas uma instrução, a instrução `return`. Por exemplo:

```
>>> concatena(lambda l : len(l) >= 2, [[1], [2, 3], [4, 5]])  
[2, 3, 4, 5]  
>>> concatena(lambda l : 2 in l, [[7, 2], [6, 4, 3], [5, 2]])  
[7, 2, 5, 2]
```

Solução:

```
def concatena(pred, l):  
    return acumula(lambda x,y : x + y, filtra(pred, l))
```