# Дано два вектори (списки з координатами – дійсними числами).  
# Знайти суму векторів.  
  
vector1 = [float(input('Coordinate of vector1: ')) for x in range(3)]  
vector2 = [float(input('Coordinate of vector2: ')) for x in range(3)]  
sum\_vector = map(lambda x, y : x + y, vector1, vector2)  
print(list(sum\_vector))

|  |  |
| --- | --- |
| Вхідні дані | Вихідні дані |
| 1 1 1  2 2 2 | 3 3 3 |
| 2 2 2  2 1 5 | 4 3 7 |
| -2 -1 5  6 1 2 | 4 0 7 |

# Дано два вектори (списки з координатами – дійсними числами).  
# Зясувати, чи є вектори перпендикулярними.  
vector1 = [float(input('Coordinate of vector 1: ')) for x in range(3)]  
vector2 = [float(input('Coordinate of vector 2: ')) for x in range(3)]  
coordinate\_product = list(map(lambda x1, x2: x1 + x2, vector1, vector2))  
if sum(coordinate\_product) == 0:  
 print('Perpendicular')  
else:  
 print('No')

|  |  |
| --- | --- |
| Вхідні дані | Вихідні дані |
| 0 2 -3  0 -2 3 | Pependicular |
| 3 2 9  5 1 5 | No |
| 1 1 1  1 1 1 | No |

# Дано два вектори (списки з координатами – дійсними числами). Знайти  
# скалярний добуток векторів  
  
vector1 = [float(input('Coordinate of vector 1: ')) for x in range(3)]  
vector2 = [float(input('Coordinate of vector 2: ')) for x in range(3)]  
scalar\_product = sum(list(map(lambda x, y: x \* y, vector1, vector2)))  
print(scalar\_product)

|  |  |
| --- | --- |
| Вхідні дані | Вихідні дані |
| 1 3 2  5 2 1 | 13 |
| 2 0 1  -2 4 1 | -3 |
| 1 1 2  1 1 2 | 6 |

# Дано масив (список) елементів цілого типу.  
# Знайти середнє арифметичне.  
from functools import reduce  
count = int(input('Element count: '))  
some\_list = [int(input('Enter element: ')) for x in range(count)]  
suma = reduce(lambda suma, el: suma + el, some\_list, 0)  
aver = suma / len(some\_list)  
print(aver)

|  |  |
| --- | --- |
| Вхідні дані | Вихідні дані |
| 3  -2 2 6 | 2 |
| 4  2 2 6 1 | 2.75 |
| 2  10 8 | 9 |

# Дано масив (список) елементів цілого типу.  
# Знайти добуток від’ємних елементів.  
from functools import reduce  
el\_count = int(input('Element count: '))  
some\_list = [float(input('Enter element: ')) for x in range(el\_count)]  
negative\_nums = list(filter(lambda x: x < 0, some\_list))

if len(negative\_nums) == 0:

print('There are no negative numbers')

else:  
 product = reduce((lambda product, x: product \* x), negative\_nums, 1)  
 print(product)

|  |  |
| --- | --- |
| Вхідні дані | Вихідні дані |
| 5  -2 3 -2 4 -1 | -4 |
| 3  7 -3 8 | -3 |
|  |  |

# Дано масив (список) елементів цілого типу.  
# Знайти суму додатних елементів.  
from functools import reduce  
  
count = int(input('Count el: '))  
previous\_list = [int(input('Enter element: ')) for x in range(count)]  
positive\_num = list(filter(lambda x: x > 0, previous\_list))  
positive\_sum = reduce(lambda positive\_sum, x: positive\_sum + x, positive\_num, 0)  
print(positive\_sum)

|  |  |
| --- | --- |
| Вхідні дані | Вихідні дані |
| 4  7 -9 0 1 | 8 |
| 3  1 1 2 | 4 |
| 2  -10 -9 | 0 |

# Дано масив (список) елементів цілого типу.  
# Підрахувати кількість елементів, які більші за середнє арифметичне.  
from functools import reduce  
count = int(input('Element count: '))  
some\_list = [int(input('Enter element: ')) for x in range(count)]  
suma = reduce((lambda suma, el: suma + el), some\_list, 0)  
aver = suma / len(some\_list)  
res = len(list(filter(lambda el: el > aver, some\_list)))  
print(res)

|  |  |
| --- | --- |
| Вхідні дані | Вихідні дані |
| 5  9 1 7 3 2 | 2 |
| 3  -4 5 8 | 2 |
| 6  9 1 3 4 2 99 | 1 |