

Яндекс



Кортежи. Преобразование коллекций

Программа

"Подробный список покупок"



Программа "Подробный список покупок"

```
n = int(input())
purchase_list = []
purchase_amount_list = []
for i in range(n):
    purchase = input()
    purchase_list.append(purchase)
    purchase_amount = input()
    purchase_amount_list.append(purchase_amount)
for i in range(n - 1, -1, -1):
    print(purchase_list[i], purchase_amount_list[i])
```

Программа "Подробный список покупок"

```
n = int(input())
purchase_list = []
purchase_amount_list = []
for i in range(n):
    purchase = input()
    purchase_list.append(purchase)
    purchase_amount = input()
    purchase_amount_list.append(purchase_amount)
for i in range(n - 1, -1, -1):
    print(purchase_list[i], purchase_amount_list[i])
```

или:

```
for i in range(n):
    print(purchase_list[-1 - i], purchase_amount_list[-1 - i])
```

Решение задачи "Отбор"



Решение задачи "Отбор"

```
n = int(input())
pupils = []
for i in range(n):
    pupils.append(input())

for pupil in pupils:
    print(pupil)
print()

for pupil in pupils:
    if pupil[-1] in ['4', '5']:
        print(pupil)
```

Кортежи (тип tuple)



Кортежи (тип tuple)

```
empty = ()  
card = ( '7', 'пик' )  
t = (18, )  
print( len(card) )
```

Кортежи (тип tuple)

```
empty = ()  
card = ( '7', 'пик' )  
t = (18, )  
print(len(card))
```

Кортежи (тип tuple)

```
empty = ()  
card = ( '7', 'пик' )  
t = (18, )  
print(len(card))  
print(card[0])
```

Кортежи (тип tuple)

```
empty = ()  
card = ( '7', 'пик' )  
t = (18, )  
print(len(card))  
print(card[0])
```

2

7

Кортежи (тип tuple)

```
empty = ()  
card = ( '7', 'пик' )  
t = (18, )  
print(len(card))  
print(card[0])  
print(card + t)
```

2

7

Кортежи (тип tuple)

```
empty = ()  
card = ( '7', 'пик' )  
t = (18, )  
print(len(card))  
print(card[0])  
print(card + t)
```

2

7

('7', 'пик', 18)

Кортежи (тип tuple)

```
empty = ()  
card = ( '7', 'пик' )  
t = (18, )  
print(len(card))  
print(card[0])  
print(card + t)  
card[1] = 'треф'
```

2

7

('7', 'пик', 18)

Кортежи (тип tuple)

```
empty = ()  
card = ( '7', 'пик' )  
t = (18, )  
print(len(card))  
print(card[0])  
print(card + t)  
card[1] = 'треф'
```

2

7

('7', 'пик', 18)

'tuple' object does not support item assignment

Сравнение кортежей



Сравнение кортежей

(1, 2, 3) == (1, 3, 2)

Сравнение кортежей

(1, 2, 3) == (1, 3, 2)

False

Сравнение кортежей

(1, 2, 3) == (1, 3, 2)

False

(1, 2, 3) < (1, 2, 4)

Сравнение кортежей

(1, 2, 3) == (1, 3, 2)

False

(1, 2, 3) < (1, 2, 4)

True

Сравнение кортежей

`(1, 2, 3) == (1, 3, 2)`

False

`(1, 2, 3) < (1, 2, 4)`

True

`(1, 2) <= (5,)`

Сравнение кортежей

(1, 2, 3) == (1, 3, 2)

False

(1, 2, 3) < (1, 2, 4)

True

(1, 2) <= (5,)

True

Сравнение кортежей

(1, 2, 3) == (1, 3, 2)

False

(1, 2, 3) < (1, 2, 4)

True

(1, 2) <= (5,)

True

('7', 'треф') > ('7', 'червей')

Сравнение кортежей

(1, 2, 3) == (1, 3, 2)

False

(1, 2, 3) < (1, 2, 4)

True

(1, 2) <= (5,)

True

('7', 'треф') > ('7', 'червей')

False

Сравнение кортежей

(1, 2, 3) == (1, 3, 2)

False

(1, 2, 3) < (1, 2, 4)

True

(1, 2) <= (5,)

True

('7', 'треф') > ('7', 'червей')

False

(1, 2) != ('7', 'пик')

Сравнение кортежей

`(1, 2, 3) == (1, 3, 2)` False

`(1, 2, 3) < (1, 2, 4)` True

`(1, 2) <= (5,)` True

`('7', 'треф') > ('7', 'червей')` False

`(1, 2) != ('7', 'пик')` True

Сравнение кортежей

`(1, 2, 3) == (1, 3, 2)` False

`(1, 2, 3) < (1, 2, 4)` True

`(1, 2) <= (5,)` True

`('7', 'треф') > ('7', 'червей')` False

`(1, 2) != ('7', 'пик')` True

`(3, 4) < ('5', 'бубен')`

Сравнение кортежей

`(1, 2, 3) == (1, 3, 2)`

False

`(1, 2, 3) < (1, 2, 4)`

True

`(1, 2) <= (5,)`

True

`('7', 'треф') > ('7', 'червей')`

False

`(1, 2) != ('7', 'пик')`

True

`(3, 4) < ('5', 'бубен')`

'<' not supported
between instances
of 'int' and 'str'

Присваивание кортежей



Присваивание кортежей

```
(n, s) = (10, 'hello')
```

Присваивание кортежей

```
(n, s) = (10, 'hello')
```

то же самое, что

```
n, s = 10, 'hello'
```


Присваивание кортежей

```
(n, s) = (10, 'hello')
```

то же самое, что

```
n, s = 10, 'hello'
```

то же самое, что

```
n = 10
```

```
s = 'hello'
```

Присваивание кортежей

```
cards = [('7', 'пик'), ('Д', 'треф'), ('Т', 'пик')]
value, suit = cards[0]
print('Достоинство карты:', value)
```

Присваивание кортежей

```
cards = [('7', 'пик'), ('Д', 'треф'), ('Т', 'пик')]
value, suit = cards[0]
print('Достоинство карты:', value)
```

Достоинство карты: 7

Присваивание кортежей

```
cards = [('7', 'пик'), ('Д', 'треф'), ('Т', 'пик')]
value, suit = cards[0]
print('Достоинство карты:', value)
print('Масть карты:', suit)
```

Достоинство карты: 7

Присваивание кортежей

```
cards = [('7', 'пик'), ('Д', 'треф'), ('Т', 'пик')]
value, suit = cards[0]
print('Достоинство карты:', value)
print('Масть карты:', suit)
```

Достоинство карты: 7

Масть карты: пик

Обмен значениями переменных



Обмен значениями переменных

```
a, b = 1, 2
```

```
a = b
```

```
b = a
```

```
print(a, b, sep=', ')
```

Обмен значениями переменных

```
a, b = 1, 2
```

```
a = b
```

```
b = a
```

```
print(a, b, sep=', ')
```

2, 2

Обмен значениями переменных

```
a, b = 1, 2
```

```
a = b
```

```
b = a
```

```
print(a, b, sep=', ')
```

2, 2

Обмен значениями переменных

```
a, b = 1, 2
```

```
t = a
```

```
a = b
```

```
b = t
```

```
print(a, b, sep=', ')
```

Обмен значениями переменных

```
a, b = 1, 2
```

```
t = a
```

```
a = b
```

```
b = t
```

```
print(a, b, sep=', ')
```

```
2, 1
```

Обмен значениями переменных

```
a, b = 1, 2
```

```
a, b = b, a
```

```
print(a, b, sep=', ')
```

Обмен значениями переменных

```
a, b = 1, 2
```

```
a, b = b, a
```

```
print(a, b, sep=', ')
```

2, 1

Обмен значениями переменных

```
a, b = 1, 2
```

```
a, b = b, a
```

```
print(a, b, sep=', ')
```

```
a, b, c = 3, 2, 1
```

```
b, a, c = c, a, b
```

```
print(b, c, a, sep=', ')
```

```
2, 1
```

Обмен значениями переменных

```
a, b = 1, 2
```

```
a, b = b, a
```

```
print(a, b, sep=', ')
```

```
a, b, c = 3, 2, 1
```

```
b, a, c = c, a, b
```

```
print(b, c, a, sep=', ')
```

```
2, 1
```

```
1, 2, 3
```

Числа Фибоначчи



Числа Фибоначчи

```
n = int(input())  
f1, f2 = 0, 1  
for i in range(n):  
    print(f2)  
    f1, f2 = f2, f1 + f2
```

Числа Фибоначчи

```
n = int(input())  
f1, f2 = 0, 1  
for i in range(n):  
    print(f2)  
    f1, f2 = f2, f1 + f2
```

1

Числа Фибоначчи

```
n = int(input())  
f1, f2 = 0, 1  
for i in range(n):  
    print(f2)  
    f1, f2 = f2, f1 + f2
```

1
1

Числа Фибоначчи

```
n = int(input( ))
f1, f2 = 0, 1
for i in range(n):
    print(f2)
    f1, f2 = f2, f1 + f2
```

1
1
2

Числа Фибоначчи

```
n = int(input( ))  
f1, f2 = 0, 1  
for i in range(n):  
    print(f2)  
    f1, f2 = f2, f1 + f2
```

```
1  
1  
2  
3
```

Числа Фибоначчи

```
n = int(input())  
f1, f2 = 0, 1  
for i in range(n):  
    print(f2)  
    f1, f2 = f2, f1 + f2
```

```
1  
1  
2  
3  
5
```

Числа Фибоначчи

```
n = int(input( ))
f1, f2 = 0, 1
for i in range(n):
    print(f2)
    f1, f2 = f2, f1 + f2
```

1
1
2
3
5
8

Числа Фибоначчи

```
n = int(input())  
f1, f2 = 0, 1  
for i in range(n):  
    print(f2)  
    f1, f2 = f2, f1 + f2
```

```
1  
1  
2  
3  
5  
8  
13  
...
```


Сортировка



Сортировка

[4, 1, 9, 3, 1]

Сортировка

[4, 1, 9, 3, 1]

Отсортированный по возрастанию:

Сортировка

[4, 1, 9, 3, 1]

Отсортированный по возрастанию:

[1, 1, 3, 4, 9]

Сортировка

[4, 1, 9, 3, 1]

Отсортированный по возрастанию:

[1, 1, 3, 4, 9]

Отсортированный по убыванию:

Сортировка

[4, 1, 9, 3, 1]

Отсортированный по возрастанию:

[1, 1, 3, 4, 9]

Отсортированный по убыванию:

[9, 4, 3, 1, 1]

Сортировка

['банан' , 'арбуз' , 'виноград' , 'вино']

Сортировка

['банан' , 'арбуз' , 'виноград' , 'вино']

Отсортированный по возрастанию:

Сортировка

['банан' , 'арбуз' , 'виноград' , 'вино']

Отсортированный по возрастанию:

['арбуз' , 'банан' , 'вино' , 'виноград']

Сортировка

['банан' , 'арбуз' , 'виноград' , 'вино']

Отсортированный по возрастанию:

['арбуз' , 'банан' , 'вино' , 'виноград']

Отсортированный по убыванию:

Сортировка

['банан' , 'арбуз' , 'виноград' , 'вино']

Отсортированный по возрастанию:

['арбуз' , 'банан' , 'вино' , 'виноград']

Отсортированный по убыванию:

['виноград' , 'вино' , 'банан' , 'арбуз']

Сортировка пузырьком



Сортировка пузырьком

"Всплывает" в конец списка сначала самый большой элемент, потом второй по величине...

Для этого сравниваются по очереди первый элемент со вторым, второй с третьим и т.д., и больший из двух ставится дальше.

После первого прохода "всплыл" самый большой элемент, после второго — второй, и т.д.

Конец списка отсортирован, можно ещё раз по нему не проходить.

Сортировка пузырьком

30	20	100	10	1	2	3
20	30	100	10	1	2	3
20	30	100	10	1	2	3
20	30	10	100	1	2	3
20	30	10	1	100	2	3
20	30	10	1	2	100	3
20	30	10	1	2	3	100
20	30	10	1	2	3	100
20	10	30	1	2	3	100
20	10	1	30	2	3	100
20	10	1	2	30	3	100
20	10	1	2	3	30	100
10	20	1	2	3	30	100
10	1	20	2	3	30	100
10	1	2	20	3	30	100

Сортировка пузырьком

10	1	2	3	20	30	100
1	10	2	3	20	30	100
1	2	10	3	20	30	100
1	2	3	10	20	30	100
1	2	3	10	20	30	100
1	2	3	10	20	30	100
1	2	3	10	20	30	100

Всего 21 сравнение для 7 элементов

Сортировка пузырьком

```
n = int(input())
a = []
for i in range(n):
    a.append(int(input()))

for i in range(n-1):
    for j in range(n-1-i):
        if a[j] > a[j+1]:
            a[j], a[j+1] = a[j+1], a[j]

print(a)
```


Преобразования между коллекциями



Преобразования между коллекциями

```
writer = ( 'Лев Толстой', 1827 )  
a = list(writer)  
a[1] = 1828  
writer = tuple(a)  
print(writer)
```

Преобразования между коллекциями

```
writer = ( 'Лев Толстой', 1827 )  
a = list(writer)  
a[1] = 1828  
writer = tuple(a)  
print(writer)
```

```
( 'Лев Толстой', 1828 )
```

Преобразования между коллекциями

```
a = [1, 2, 1, 1, 2, 2, 3, 3]  
print(len(set(a)))
```

Преобразования между коллекциями

```
a = [1, 2, 1, 1, 2, 2, 3, 3]  
print(len(set(a)))
```

Преобразования между коллекциями

```
a = [1, 2, 1, 1, 2, 2, 3, 3]
print(len(set(a)))
names = {'Иван', 'Петр', 'Сергей', 'Алексей'}
print(list(names))
```

Преобразования между коллекциями

```
a = [1, 2, 1, 1, 2, 2, 3, 3]
print(len(set(a)))
names = {'Иван', 'Петр', 'Сергей', 'Алексей'}
print(list(names))
```

3

['Петр', 'Иван', 'Сергей', 'Алексей']

Яндекс