

Практическая работа 8

Задача 1

```
In [3]: n = int(input())

sum = 0
for i in range(1, n + 1):
    sum += i ** 2
print(sum)

3
14
```

Задача 2

```
In [6]: n = int(input())

for i in range(n):
    print("+___", end=" ")
print()
for i in range(n):
    print("|%s /" % (i + 1), end=" ")
print()
for i in range(n):
    print("|__\\", end=" ")
print()
for i in range(n):
    print("|   ", end=" ")
print()

2
+___ +___
|1 / |2 /
|__\ |__\
|   |
```

Задача 3

```
In [7]: s = list(input().split(' '))
for i in reversed(s):
    print(i, end=' ')

1 2 3 4 5
5 4 3 2 1
```

Задача 4

```
In [10]: a = [int(s) for s in input().split()]
k = int(input())
for i in range(k + 1, len(a)):
    a[i-1] = a[i]
a.pop()
print(' '.join([str(i) for i in a]))
```

```
4 6 2 4 3 5 12 24 3 5
0
6 2 4 3 5 12 24 3 5
```

Задача 5

```
In [14]: a = [int(i) for i in input().split()]
for i in range(1, len(a), 2):
    a[i-1], a[i] = a[i], a[i-1]
print(' '.join([str(i) for i in a]))
```

```
9 4 5 2 3
4 9 2 5 3
```

Задача 6

```
In [16]: lst = input().split()
for i in reversed(range(len(lst))):
    if lst[i] == '0':
        lst.append(lst.pop(i))
print(*lst)
```

```
0 0 0 0 0 0 0 0 0 1
1 0 0 0 0 0 0 0 0 0
```

Задача 7

```
In [20]: k = int(input())
N = [int(input()) for i in range(k)]
print(*sorted(N))
```

```
2
3
1
1 3
```

Задача 8

```
In [1]: S,N = map(int, input().split())
volume = sorted([int(input()) for _ in range(N)])
amount = sum(volume)
while amount > S and N:
    amount -= volume.pop()
    N -= 1
print(N)
```

```
100 3
50
30
50
2
```

Задача 9

```
In [25]: button = int(input())
norm_press_button = list(map(int, input().split()))
number_press = int(input())
spb = list(map(int, input().split()))
dict_clic = {}
for i in range(number_press):
    dict_clic[spb[i]] = dict_clic.get(spb[i],0)+1
for i in range(button):
    print(('NO', 'YES')[dict_clic[i+1] > norm_press_button[i]])
```

```
5
1 50 3 4 3
16
1 2 3 4 5 1 3 3 4 5 5 5 5 4 5
YES
NO
NO
NO
YES
```

Задача 10

```
In [4]: def CountSort(list):
sortedList = [0] * 101
for i in list:
    sortedList[i] += 1
for i in range(101):
    print((str(i) + ' ') * sortedList[i], end='')

list = [int(i) for i in input().split()]
CountSort(list)
```

```
9 8 7 6 5 4 3 2 1 0
0 1 2 3 4 5 6 7 8 9
```

Задача 11

```
In [13]: num = int(input())
members = [input().split() for _ in range(num)]
for i in sorted(members, key=lambda x: int(x[1]), reverse=True):
    print(i[0])

3
Ivanov 10
Petrov 15
Sidorov 20
Sidorov
Petrov
Ivanov
```

Задача 12

```
In [10]: tariffs = sorted([int(s) for s in input().split()])
distances = sorted([int(s) for s in input().split()], reverse=True)
print(sum(t*d for t, d in zip(tariffs, distances)))

1 2 3
1 2 3
10
```

Задача 13

```
In [14]: def reformat(string):
    string = string.replace('-', '').replace('(', '').replace(')', '')
    return string[-10:] if len(string)>7 else '495' + string[-7:]

n = 4
notes = [input() for _ in range(n)]
for note in notes[1:]:
    print('YES' if reformat(notes[0]) == reformat(note) else 'NO')

+78047952807
+78047952807
+76147514928
88047952807
YES
NO
YES
```

Задача 14

```
In [15]: motherland = {}  
for i in range(int(input())):  
    country, *cities = input().split()  
    for city in cities:  
        motherland[city] = country  
  
for i in range(int(input())):  
    print(motherland[input()])
```

```
2  
Russia Moscow Petersburg Novgorod Kaluga  
Ukraine Kiev Donetsk Odessa  
3  
Odessa  
Ukraine  
Moscow  
Russia  
Novgorod  
Russia
```

Задача 15

```
In [18]: def most_common(t):  
    return sorted(((x, t.count(x)) for x in set(t.split()))), key=lambda x: (-  
x[1], x[0]))[0][0]  
most_common(input())
```

```
oh you touch my tralala mmm my ding ding dong
```

```
Out[18]: 'ding'
```

Задача 16

```
In [19]: s = input()  
lst = [word for line in s.split('\n') for word in line.split()]  
print(sorted(set(lst), key=lambda x: (-lst.count(x), x)))
```

```
oh you touch my tralala mmm my ding ding dong  
['ding', 'my', 'dong', 'mmm', 'oh', 'touch', 'tralala', 'you']
```

Задача 17

```
In [ ]: def deposit(arg):
        name, money = arg
        bank[name] = bank.setdefault(name, 0) + int(money)
    def withdraw(arg):
        name, money = arg
        bank[name] = bank.setdefault(name, 0) - int(money)
    def balance(arg):
        name = arg[0]
        if name in bank:
            print(bank[name])
        else:
            print('ERROR')
    def transfer(arg):
        name_1, name_2, money = arg
        for name in (name_1, name_2):
            if name not in bank:
                deposit((name,0))
            withdraw((name_1, money))
            deposit((name_2, money))
    def income(arg):
        percent = int(arg[0])
        for name, balance in bank.items():
            if balance > 0:
                bank[name] = bank.get(name) + balance * percent//100
    bank = {}
    bank_fun = {
        'DEPOSIT' : deposit, 'WITHDRAW' : withdraw,
        'BALANCE' : balance, 'TRANSFER' : transfer,
        'INCOME' : income
    }
    while 1:
        data = input().split()
        fun_name = data[0]
        arg = data[1:]
        bank_fun[fun_name](arg)
```

```
DEPOSIT Ivanov 100
INCOME 5
BALANCE Ivanov
105
TRANSFER Ivanov Petrov 50
WITHDRAW Petrov 100
BALANCE Petrov
-50
BALANCE Sidorov
ERROR
```

Задача 18

```
In [1]: def height(man):
        if man not in p_tree:
            return 0
        else:
            return 1 + height(p_tree[man])
p_tree = {}
n = int(input())
for i in range(n - 1):
    child, parent = input().split()
    p_tree[child] = parent
heights = {}
for man in set(p_tree.keys()).union(set(p_tree.values())):
    heights[man] = height(man)
for key, value in sorted(heights.items()):
    print(key, value)
```

```
9
Alexei Peter_I
Anna Peter_I
Elizabeth Peter_I
Peter_II Alexei
Peter_III Anna
Paul_I Peter_III
Alexander_I Paul_I
Nicholaus_I Paul_I
Alexander_I 4
Alexei 1
Anna 1
Elizabeth 1
Nicholaus_I 4
Paul_I 3
Peter_I 0
Peter_II 2
Peter_III 2
```

Задача 19

```
In [4]: print(len(set(input().split())))
```

```
1 2 3 4 5 1 2 1 2 7 3
6
```

Задача 20

```
In [6]: iin_file = input()
a = []
for i in iin_file.split('\n'):
    a += i.split()
print(len(set(a)))
```

She sells sea shells on the sea shore; The shells that she sells are sea shell
s I'm sure. So if she sells sea shells on the sea shore, I'm sure that the she
lls are sea shore shells.

```
19
```

Задача 21

```
In [7]: a = input()
b = input()
c = ' '.join(map(lambda a, b : str(int(a != b)), a.split(), b.split()))
print(c)
```

```
0 0 1 1
0 1 0 1
0 1 1 0
```

Задача 22

```
In [10]: from itertools import accumulate
n = int(input())
a = []
for i in range(n+1):
    a.append(i)
a = map(lambda x: max(x,1), a)
print(*accumulate(a, lambda x, y: x * y))
```

```
1
1 1
```

Задача 23

```
In [13]: N = int(input())
res = []
for k in range(2, N+1):

    prime = True

    for i in range(2, k):
        if k%i == 0:
            prime = False
            break

    if prime:
        res.append(k)
print(res)
```

```
5
[2, 3, 5]
```

Задача 24


```
In [11]: import itertools
import functools
print(*(tuple([[[0]]]) + tuple(*map(lambda data: filter(lambda x: tuple(x[0])
== x[1],
    map(lambda var: (filter(lambda sta: ((var.index(sta[0]) >= var.index(
sta[1])) ^ (var.index(sta[2]) >= var.index(sta[3]))), data[1]), data[1], var
+ data[2]), data[0])), map(lambda x: (itertools.permutations(set(functools.re
duce(lambda x,y: x + y ,x[1]))), x[1], tuple(set(range(1, x[0] + 1)) - set(fu
nctools.reduce(lambda x,y: x + y ,x[1])))), map(lambda vvod: (next(vvod), tup
le(set(map(lambda y: tuple(y), map(lambda x: map(int, input().split()), range
(next(vvod))))))), [map(int, input().split())])))))[-1][-1])

3 2
2 1 2 3
1 2 3 2
3 2 1
```

In []: