

Олимпиада СПбГУ по информатике 2022/23 учебного года

A	B	C	D	E	F	Sum
100	100	100	60	28	10	398

Task A ()

```
nums = list(map(int, input().split()))
tasks = []
res = [0 for _ in range(len(nums))]

for i in range(len(nums)):
    tasks.insert(nums[i] - 1, i)

for i in range(len(nums)):
    res[tasks[i]] = i + 1

print(' '.join(map(str, res)))
```

Task B ()

```
string = input()
if string == 'first':
    n = int(input())
    arr = list(map(int, input().split()))
    sum_ = sum(arr)
    mid = sum(arr) // n
    rem = sum_ - mid * n
    print(rem * 1000 + mid)
else:
    _ = input()
    arr = list(map(int, input().split()))
    rem = arr[0] // 1000
    arr = [i % 1000 for i in arr]
    print(sum(arr) + rem)
```

Task C ()

```
res = set()

def add(a, b):
    if a > b:
        a, b = b, a
    res.add((a, b))

def q(wrapper, first, second):
    if first[0] == wrapper[0] and first[1] < wrapper[1]:
        add(first[0], wrapper[1] - first[1])

    if first[1] + second[1] <= wrapper[1] and first[0] <= wrapper[0] and second[0] <= wrapper[0]
    and (first[0] < wrapper[0] or second[0] < wrapper[0]):
        if first[1] + second[1] == wrapper[1] and first[0] == second[0]:
            add(wrapper[1], wrapper[0] - first[0])
        if first[0] + second[0] > wrapper[0] or (first[0] + second[0] == wrapper[0] and first[1] +
        second[1] == wrapper[1]):
            if first[0] < wrapper[0]:
                add(first[1], wrapper[0] - first[0])
        if first[1] + second[1] < wrapper[1]:
            if first[0] == wrapper[0]:
                add(first[0], wrapper[1] - first[1] - second[1])

def main():
    a1, b1 = map(int, input().split())
    a2, b2 = map(int, input().split())
    a3, b3 = map(int, input().split())

    add(a1, b1)
    add(a2, b2)
    add(a3, b3)

    for wrapper in ((1, a1, b1), (2, a2, b2), (3, a3, b3)):
        for first in ((1, a1, b1), (2, a2, b2), (3, a3, b3)):
            for second in ((1, a1, b1), (2, a2, b2), (3, a3, b3)):
                if wrapper[0] == first[0] or wrapper[0] == second[0] or first[0] == second[0]:
                    continue

                for wo in 0, 1:
                    for fo in 0, 1:
                        for so in 0, 1:
                            w = (wrapper[1], wrapper[2]) if wo else (wrapper[2], wrapper[1])
                            f = (first[1], first[2]) if fo else (first[2], first[1])
                            s = (second[1], second[2]) if so else (second[2], second[1])
                            q(w, f, s)

    l_res = list(res)
    l_res.sort()

    for i in l_res:
        print(f'{{i[0]}}_{{i[1]}}')

main()
```

Task D ()

```
import sys

n = int(input())
arr = list(map(int, input().split()))
f = [0 for i in range(n)]

if n == 1:
    print(1, arr[0])
    sys.stdout.flush()
    input()
    print(1, arr[0])
    sys.stdout.flush()
    input()
elif n == 2:
    while True:
        if arr[0] > 1 and arr[1] > 1 or arr[0] == 1 and arr[1] == 1:
            print(-1, -1)
            sys.stdout.flush()
            break
        elif arr[0] > 1:
            print(1, arr[0] - 1)
            arr[0] = 1
            sys.stdout.flush()
        elif arr[1] > 1:
            print(2, arr[1] - 1)
            arr[1] = 1
            sys.stdout.flush()
        elif arr[0] == 0:
            print(2, 1)
            sys.stdout.flush()
        elif arr[1] == 0:
            print(1, 1)
            sys.stdout.flush()

    a, b = map(int, input().split())
    if a == -1:
        break
    if b == 0:
        print(1 if a == 2 else 2, 0)
        sys.stdout.flush()
    else:
        arr[a - 1] -= b
```

Task E ()

```
t = int(input())
x = input() == 'transmit'

for _ in range(t):
    if x:
        a = int(input())
        for i, b in enumerate(bin(a)[2:].rjust(10, '0')):
            if b == '0':
                print('0' * 10)
            else:
                print('1' * (i + 1) + '0' * (9 - i))
    else:
        input()
        res = [0 for i in range(10)]
        for _ in range(10):
            q = input().count('1')
            if q:
                res[q - 1] = 1
        ress = ''
        for i in res:
            ress += '1' if i else '0'
        print(int(ress, 2))
```

Task F ()

```
def parse_num(string):
    state = 0
    res = []
    buf = ''
    buf1 = ''

    for i in string:
        if state == 0:
            if i == '(':
                state = 1
            else:
                res.append([i, 1])
        elif state == 1:
            if i == '|':
                state = 2
            else:
                buf += i
        else:
            if i == ')':
                state = 0
                res.append([buf, int(buf1)])
                buf = ''
                buf1 = ''
            else:
                buf1 += i
    return res

a = parse_num(input())
b = parse_num(input())

res = []
rem = 0

while True:
    if not a and not b:
        if rem:
            res.append([1, 1])
        break
    if not a:
        if rem:
            if b[-1][0] == '9':
                res.append([0, b.pop()[1]])
                continue
            a.append([rem, 1])
            rem = 0
        else:
            while b:
                res.append(b.pop())
            break
    if not b:
        if rem:
            if a[-1][0] == '9':
                res.append([0, a.pop()[1]])
                continue
            b.append([rem, 1])
            rem = 0
        else:
            while a:
                res.append(a.pop())
            break

    sum_ = int(a[-1][0]) + int(b[-1][0])
    if a[-1][1] <= b[-1][1]:
        count = a[-1][1]
        a.pop()
        if b[-1][1] == count:
            b.pop()
        else:
            b[-1][1] = b[-1][1] - count
    else:
        count = b[-1][1]
```

```

        b.pop()
        a[-1][1] = a[-1][1] - count
    if sum_ > 9:
        if rem:
            res.append([sum_ % 10 + 1, count])
        else:
            res.append([sum_ % 10, 1])
            rem = 1
            if count > 1:
                res.append([sum_ % 10 + 1, count - 1])
    else:
        if rem and sum_ == 9:
            res.append([0, count])
        elif rem:
            res.append([sum_ + 1, 1])
            rem = 0
            if count > 1:
                res.append([sum_, count - 1])
        else:
            res.append([sum_, count])

ress = ''
for i in res[::-1]:
    if i[1] < 6:
        ress += str(i[0]) * i[1]
    else:
        ress += f'({i[0]}|{i[1]})'

print(ress)

```