

```
36     self.fingerprints = set()
37     if path:
38         self.file = open(os.path.join(path, 'fingerprint.txt'))
39         self.file.seek(0)
40         self.fingerprints.update(fp for fp in self.file)
41
42     @classmethod
43     def from_settings(cls, settings):
44         debug = settings.getbool('DEBUG', False)
45         return cls(job_dir(settings), debug)
46
47     def request_seen(self, request):
48         fp = self.request_fingerprint(request)
49         if fp in self.fingerprints:
50             return True
51         self.fingerprints.add(fp)
52         if self.file:
53             self.file.write(fp + os.linesep)
```

Robocontest masalalariga tayyorgarlik birinchi kun

Bugun **1-kunni** boshlaymiz:



1-KUN — Python asoslari (RoboContest uchun optimallashtirilgan)

🎯 Maqsad:

RoboContest'da **tez, aniq, xatosiz kirish-chiqish (I/O) va asosiy matematik fikrlashni o'zlashtirish.**

🧩 1. Python'da input olish

RoboContest'da har bir masala uchun kirish (input) **standart kiritish orqali** beriladi.

Demak, har safar `input()` yoki tezroq usul — `sys.stdin.readline()` ishlatalamiz.

```
# Oddiy usul
n = int(input()) # 1 ta son
```

```
a, b = map(int, input().split()) # 2 ta son  
arr = list(map(int, input().split())) # bir qatorda bir nechta sonlar
```

⚡ Tez usul (musobaqalar uchun):

```
import sys  
input = sys.stdin.readline
```

Bu yirik kirish ma'lumotlarda **2–3 barobar tezroq** ishlaydi.



2. Asosiy matematik amallar

Python son turlari:

- `int` — cheklanmagan butun son (C++ dagi overflow yo'q)
- `float` — o'nli son (lekin RoboContest'da odatda kerak emas)

Foydali amallar:

```
a = 17  
b = 5  
  
print(a + b) # 22  
print(a - b) # 12  
print(a * b) # 85  
print(a // b) # 3 (butun bo'lish)  
print(a % b) # 2 (qoldiq)  
print(a ** b) # 1419857 (daraja)
```

12
34

3. Shart operatorlari (if / elif / else)

RoboContest masalalarining 60% sharti bo'ladi.

```
x = int(input())
```

```
if x > 0:  
    print("Musbat")  
elif x < 0:  
    print("Manfiy")  
else:  
    print("Nol")
```

4. Tsikllar (for, while)

for:

```
for i in range(1, 6):  
    print(i, end=" ")  
# 1 2 3 4 5
```

while:

```
i = 1  
while i <= 5:  
    print(i)  
    i += 1
```

5. Ro'yxatlar (list)

```
arr = [5, 2, 7, 1, 9]  
  
print(sum(arr))      # 24  
print(max(arr))     # 9  
print(min(arr))     # 1  
print(len(arr))     # 5  
  
arr.sort()  
print(arr)          # [1, 2, 5, 7, 9]
```

```
arr.reverse()  
print(arr)      # [9, 7, 5, 2, 1]
```

abc 6. String (math)

```
s = input().strip()  
  
print(s.upper()) # katta harflar  
print(s.lower()) # kichik harflar  
print(s[::-1])  # teskarisi  
print(s.count('a')) # nechta 'a' borligini hisoblaydi
```

Palindrome tekshirish:

```
s = input().strip()  
if s == s[::-1]:  
    print("YES")  
else:  
    print("NO")
```

7. Foydali kutubxonalar

RoboContest uchun tez-tez ishlataladigan modullar:

Kutubxona	Vazifasi
math	sqrt, gcd, factorial, ceil, floor
collections	Counter, deque
itertools	combinations, accumulate (prefix sum)
bisect	binary search uchun

Misol:

```
import math  
a, b = map(int, input().split())  
print(math.gcd(a, b))
```

8. Masalalar bilan mustahkamlash

Masala 1: Son raqamlari yig'indisi

Input: 472

Output: 13

```
n = int(input())  
s = 0  
while n > 0:  
    s += n % 10  
    n //= 10  
print(s)
```

Masala 2: Foydalanuvchi kiritgan sonlar orasidan eng kattasi

Input:

5

3 8 1 9 4

Output:

9

```
n = int(input())  
arr = list(map(int, input().split()))  
print(max(arr))
```

Masala 3: Palindrom so'z

Input: level

Output: YES

```
s = input().strip()  
print("YES" if s == s[::-1] else "NO")
```

Masala 4: n gacha bo'lgan toq sonlar yig'indisi

Input: 10

Output: 25 (1+3+5+7+9)

```
n = int(input())  
print(sum(i for i in range(1, n+1, 2)))
```

Masala 5: EKUB va EKUK

Input:

12 18

Output:

6 36

```
import math  
a, b = map(int, input().split())  
print(math.gcd(a, b), math.lcm(a, b))
```

Uyga vazifa:

Shular asosida o'zing quyidagi mini-masalalarni yoz:

1. 1 dan n gacha bo'lgan sonlar yig'indisi
2. n sonining faktoriali
3. 2 ta sonning kvadratlari yig'indisi

4. So'zdagi har bir harfni teskari chiqarish (`input: hello` → `output: olleh`)
 5. Ro'yxatdagi eng kichik element indeksi
-