

Hướng dẫn giải
Đề sơ loại YICT 2024 bảng A lqdoj.edu.vn



Nội dung

- Tổng quan kiến thức cần nắm để làm bài
- Bài 1: Làm bài tập
- Bài 2: Cưa gỗ
- Bài 3: $N \text{ Div } K$
- Bài 4: Chữ số thứ K



Tổng quan kiến thức

1. Nhập xuất cơ bản
2. Các câu lệnh cơ bản (if, for, while, ...)
3. Thao tác với chuỗi ký tự (string, char)
4. Thao tác với các phép toán + - * / (chia lấy nguyên và chia lấy dư)
5. Các khái niệm toán học cơ bản: số chẵn, số lẻ,...



Bài 1

Làm bài tập

Bài 1 – Làm bài tập

✓ **Điểm:** 100 (p) ⌚ **Thời gian:** 1.0s 📄 **Bộ nhớ:** 256M 📄 **Input:** bàn phím 🖨 **Output:** màn hình

Khang và Lộc đang làm bài tập về nhà của mình. Khang bắt đầu làm bài tập ở giây thứ A và hoàn thành nó ở giây thứ B . Tương tự Lộc làm bài ở giây thứ C và hoàn thành nó ở giây thứ D .

Câu hỏi đặt ra là cả hai người cùng làm bài tập trong bao nhiêu giây?

Input

- Gồm bốn dòng, mỗi dòng chứa một số tự nhiên tương ứng A, B, C, D .

Output

- In ra lượng thời gian mà cả hai đang cùng làm bài tập.

Constraints

- $A < B < 10^{16}, C < D < 10^{16}$.

Example

? Test 1

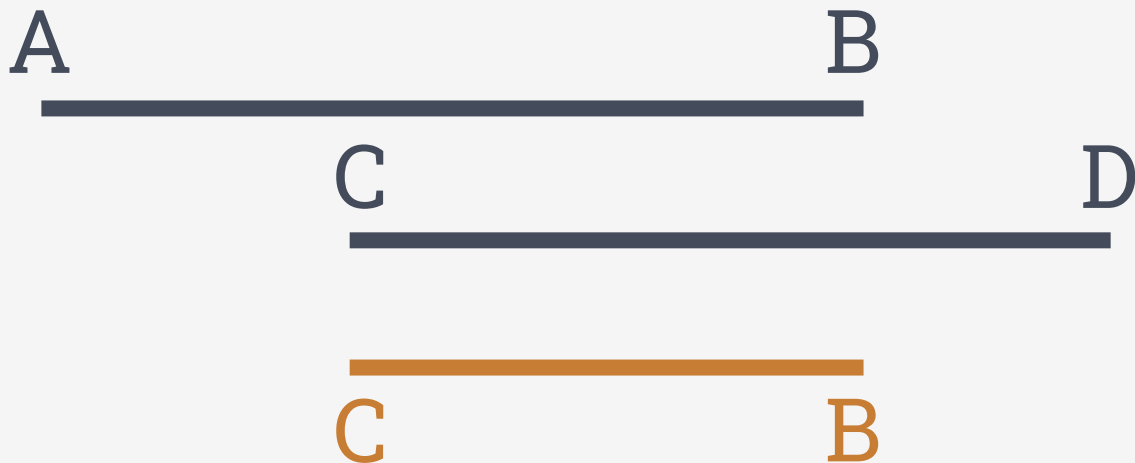
🔵 Input

0
75
25
100

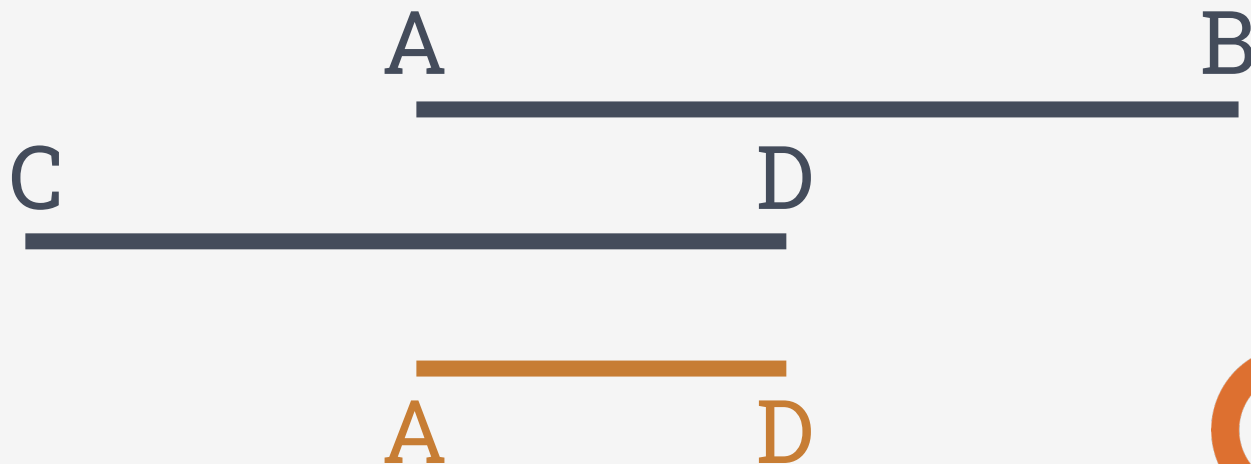
✓ Output

50

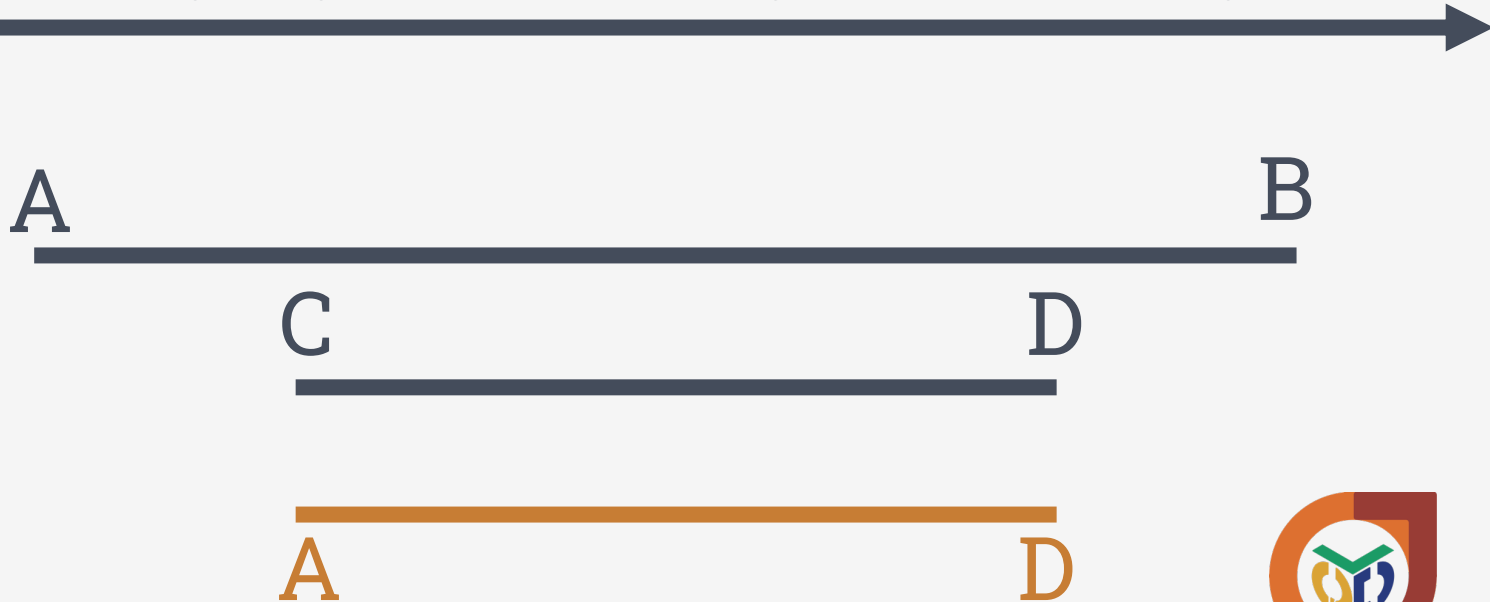
Mô tả khoảng thời gian làm bài của Khang và Lộc trên trục thời gian



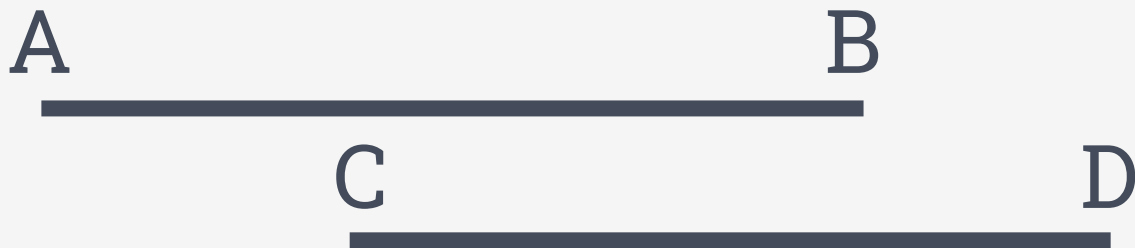
Mô tả khoảng thời gian làm bài của Khang và Lộc trên trục thời gian



Mô tả khoảng thời gian làm bài của Khang và Lộc trên trục thời gian



Mô tả khoảng thời gian làm bài của Khang và Lộc trên trục thời gian



$\text{Max}(A, C)$

$\text{Min}(B, D)$



```
a = int(input())
b = int(input())
c = int(input())
d = int(input())

x = max(a, c)
y = min(b, d)

if x < y:
    print(y - x)
else:
    print(0)
```



Bài 2

Cửa gỗ

✓ **Điểm:** 100 (p) ⌚ **Thời gian:** 1.0s 📄 **Bộ nhớ:** 256M 📄 **Input:** bàn phím 🖨 **Output:** màn hình

Ông Nam muốn cưa tấm gỗ độ dài L của mình thành nhiều mảnh, mỗi mảnh có độ dài không quá M . Mỗi lần cưa ra một mảnh có độ dài X ông mất một công sức là $X + 1$ (X công đo đạc và 1 công cưa).

Ông Nam cần sử dụng ít nhất bao nhiêu công sức để hoàn thành nhiệm vụ của mình?

Input

- Dòng thứ nhất chứa số tự nhiên L .
- Dòng thứ hai chứa số tự nhiên M

Output

- Một số nguyên duy nhất là kết quả của bài toán.

Constraints

- $1 \leq M \leq L \leq 10^9$.

? Test 1

🔗 Input

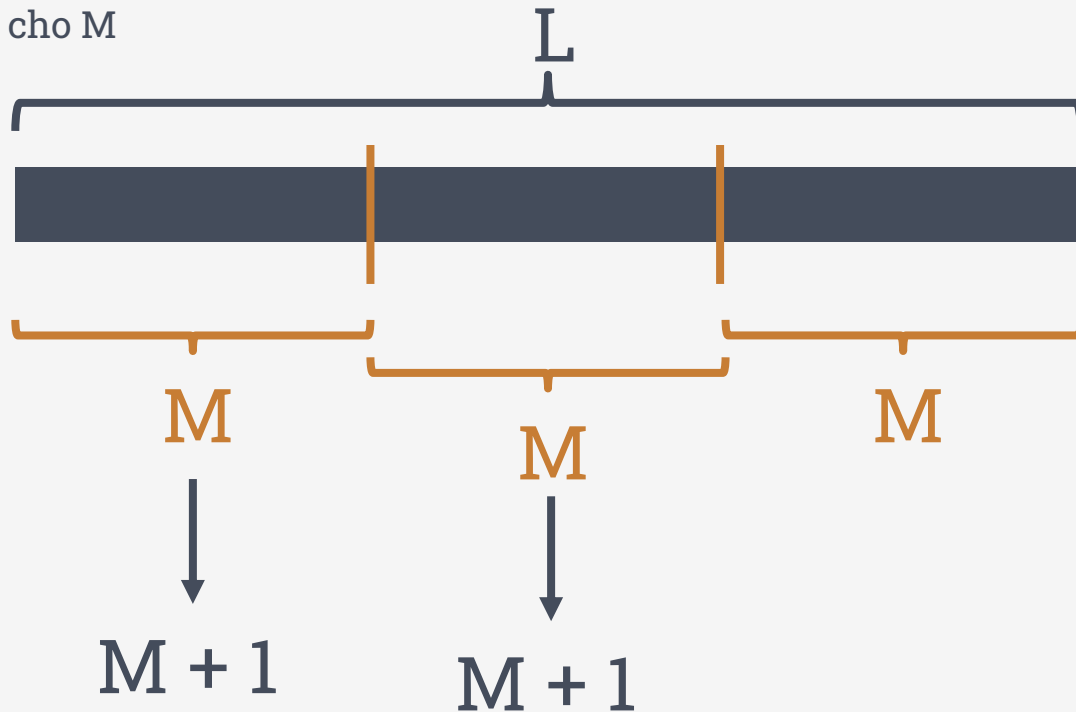
6

2

✓ Output

6

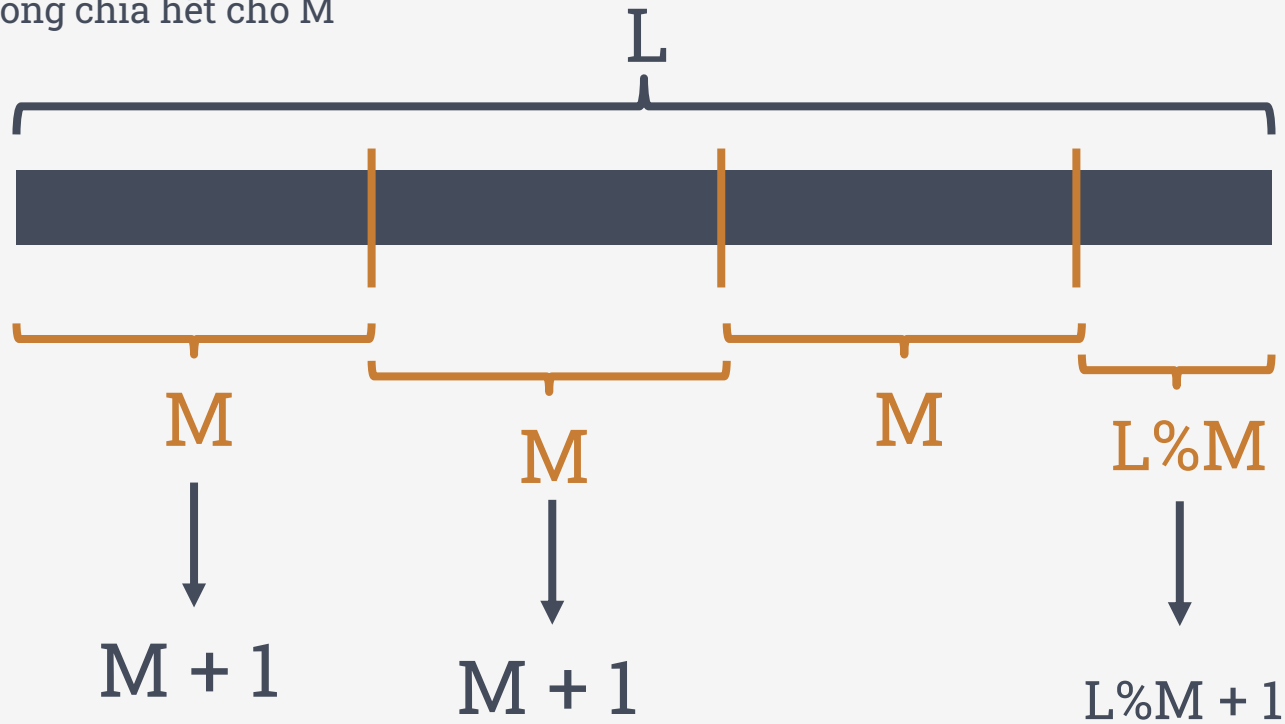
Trường hợp L chia hết cho M



$$\text{Ans} = (M + 1) * (L // M - 1)$$



Trường hợp L không chia hết cho M



$$\text{Ans} = (M + 1) * (L // M - 1) + L \% M + 1$$



```
l = int(input())  
m = int(input())  
  
if l % m == 0:  
    print((M + 1) * (L // M - 1))  
else:  
    print((M + 1) * (L // M - 1) + L % M + 1)
```



Bài 3

N Div K

✓ **Điểm:** 100 (p) ⌚ **Thời gian:** 1.0s 📄 **Bộ nhớ:** 256M 📄 **Input:** bàn phím

Định nghĩa: Phép $n \text{ div } a$ là phép chia lấy phần nguyên của n cho a .

Cho hai số nguyên dương n, k . Tính biểu thức:

$$S = n \text{ div } 1 + n \text{ div } 2 + \dots + n \text{ div } k$$

Input

- Dòng thứ nhất chứa số tự nhiên n .
- Dòng thứ hai chứa số tự nhiên k

Output

- In ra tổng S .

⚠ Note

$$16 \text{ div } 1 + 16 \text{ div } 2 + 16 \text{ div } 3 + 16 \text{ div } 4 = 16 + 8 + 5 + 4 = 33$$

Constraints

- $1 \leq n, k \leq 10^{10}$.

Example

? Test 1

🔵 Input

16

4

✓ Output

33

Code cây trầu

```
n = int(input())  
k = int(input())  
  
ans = 0  
for i in range(1, k + 1):  
    ans += n // i  
  
print(ans)
```



Ví dụ: N = 30, K = N

i	N // i
1	30
2	15
3	10
4	7
5	6
6	5
7	4
8	3
9	3
10	3
11	2
12	2
13	2
14	2
15	2
16	1
....
30	1



Ví dụ: N = 30, K = N

i	N // i	
1	30	 i không quá \sqrt{n}
2	15	
3	10	
4	7	
5	6	
6	5	 Nhiều giá trị liên tiếp trùng lặp N//i không quá \sqrt{n}
7	4	
8	3	
9	3	
10	3	
11	2	
12	2	
13	2	
14	2	
15	2	
16	1	
....	
30	1	



Bài 3 – N Div K

Ví dụ: N = 30, K = N

i	N // i	
1	30	
2	15	
3	10	
4	7	
5	6	
6	5	U = 5, V = 5 → tổng = 5 * (V - U + 1)
7	4	U = 4, V = 4 → tổng = 4 * (V - U + 1)
→ 8	3	
9	3	U = 8, V = 10 → tổng = 3 * (V - U + 1)
→ 10	3	
11	2	
12	2	
13	2	U = 11, V = 15 → tổng = 2 * (V - U + 1)
14	2	
15	2	
16	1	
....	U = 16, V = 30 → tổng = 1 * (V - U + 1)
30	1	



Bài 3 – N Div K

Ví dụ: N = 30, K = N

Gọi giá trị $N // i$
là x thì:
 $V = N / x$
 $U = N / (x + 1) + 1$

i	N // i	
1	30	
2	15	
3	10	
4	7	
5	6	
6	5	$U = 5, V = 5 \rightarrow \text{tổng} = 5 * (V - U + 1)$
7	4	$U = 4, V = 4 \rightarrow \text{tổng} = 4 * (V - U + 1)$
8	3	
9	3	$U = 8, V = 10 \rightarrow \text{tổng} = 3 * (V - U + 1)$
10	3	
11	2	
12	2	
13	2	$U = 11, V = 15 \rightarrow \text{tổng} = 2 * (V - U + 1)$
14	2	
15	2	
16	1	
....	$U = 16, V = 30 \rightarrow \text{tổng} = 1 * (V - U + 1)$
30	1	



Bài 3 – N Div K

Ví dụ: N = 30, K = 14

i	N // i	
1	30	
2	15	
3	10	
4	7	
5	6	
6	5	U = 5, V = 5 → tổng = 5 * (V - U + 1)
7	4	U = 4, V = 4 → tổng = 4 * (V - U + 1)
→ 8	3	
9	3	U = 8, V = 10 → tổng = 3 * (V - U + 1)
→ 10	3	
11	2	
12	2	
13	2	U = 11, V = K = 14 → tổng = 2 * (V - U + 1)
→ 14	2	
15	2	
16	1	
....	
30	1	

Khi giá trị k khác n
thì cho chạy v từ k
ngược về



```
n = int(input())
k = int(input())
ans = 0
v = k
while v > 0:
    x = n // v
    u = n // (x + 1) + 1
    ans += x * (v - u + 1)
    v = u - 1
print(ans)
```



Bài 4

N Div K

Các bước làm:

- Xác định chữ số thứ k nằm ở nhóm các số ODD có mấy chữ số
- Xác định số ODD nào chứa kí tự thứ k trong nhóm đó
- In ra kí tự thứ k



Bài 4 – Chữ số thứ K

Xác định chữ số thứ k nằm ở nhóm các số ODD có mấy chữ số

Đếm số lượng số ODD trong mỗi nhóm

- Có 5 số ODD có 1 chữ số
- Có $5^2 = 10$ số ODD có 2 chữ số
- Có $5^3 = 125$ số ODD có 3 chữ số
- ...

Đếm số lượng kí tự trong mỗi nhóm

- → có 5 kí tự trong nhóm 1
- → có 25×2 kí tự trong nhóm 2
- → có 125×3 kí tự trong nhóm 3
- ...

→ Chạy vòng lặp while, quản lí số lượng kí tự trong nhóm 1 + nhóm 2 + ... cho đến khi lớn hơn hoặc bằng k để xác định chữ số thứ k nằm ở nhóm nào.

```
cnt = 5
sz = 1
while cnt * sz < k:
    k -= cnt * sz
    sz += 1
cnt *= 5
```

K trừ đi cho số lượng kí tự ở các nhóm trước đó để dễ tính toán bước sau



Bài 4 – Chữ số thứ K

Xác định số ODD nào chứa kí tự thứ k trong nhóm đó

Ví dụ: trong nhóm các số ODD có 2 chữ số:

11	31	51	71	91
13	33	53	73	93
15	35	55	75	95
17	37	57	77	97
19	39	59	79	99

- Có 5 số ODD bắt đầu bằng 1
- Có 5 số ODD bắt đầu bằng 3
- Có 5 số ODD bắt đầu bằng 5
- Có 5 số ODD bắt đầu bằng 7
- Có 5 số ODD bắt đầu bằng 9

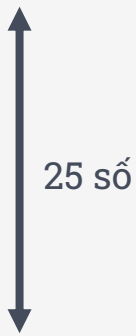


Bài 4 – Chữ số thứ K

Xác định số ODD nào chứa kí tự thứ k trong nhóm đó

Ví dụ: trong nhóm các số ODD có 3 chữ số:

111	311	511	711	911
113	313	513	713	913
115	315	515	715	915
1...	3...	5...	7...	9...
199	399	599	799	999



- Có 25 số ODD bắt đầu bằng 1
- Có 25 số ODD bắt đầu bằng 3
- Có 25 số ODD bắt đầu bằng 5
- Có 25 số ODD bắt đầu bằng 7
- Có 25 số ODD bắt đầu bằng 9

- ➔ Có 5^{i-1} số ODD có cùng chữ số bắt đầu trong nhóm các số ODD độ dài i
- ➔ Để biết chữ số bắt đầu của số chứa đáp án, ta lấy $k // 5^{i-1}$ thu được các kết quả 0, 1, 2, 3, 4 tương ứng với 1, 3, 5, 7, 9. Lặp lại như vậy cho đến hết



```
k = int(input())
cnt = 5
sz = 1
while cnt * sz < k:
    k -= cnt * sz
    sz += 1
    cnt *= 5
s = []
for i in range(sz):
    cnt //= 5
    for j in '13579':
        if cnt * sz < k:
            k -= cnt * sz
        else:
            s.append(j)
            break
print(s[k - 1])
```

