

Problem - Sub A Add B

- **Question Analysis:**

Chef এর কাছে একটা **string S** আছে। ২ টা **parameters A , B** দেওয়া আছে। **Chef** নিচের **operation** টা করবে।

- যতক্ষণ **string** এর **size $\geq A$** , ততক্ষণ **A length** এর কোনো **Substring** - কে **B length Substring** দিয়ে **replace** করে দিবে।
আর **$B < A$** এটা **guranteed**

আমাকে বলতে হবে, **final size** কত হবে **String** এর।

- **Hints:**

- **Constraint** অনেক কম। **Brute Force** চিন্তা করতে পার।
 - **Actually operation** এ হচ্ছে টা কি?
আমরা , **A size** কে **B size** দিয়ে **replace** করে দিচ্ছি, মানে -
যদি, **string** এর **current length - n** হয়, তাহলে -
 $n - A + B$ করতামি আমরা।
 - তারমানে, যতক্ষণ **$n \geq A$** থাকবে, আমরা এই **operation** টা করতে থাকব।
-

Problem - Minimum Distinct

- **Question Analysis:**

আমাকে একটা **array A** দেওয়া আছে। আমি **at most K times operation** টা করতে পারব।

Operation:

- ২ টা **index i, j choose** কর , যাতে - **$1 \leq i < j \leq N$ * * ***

আমার কাজ হচ্ছে - **operation** টি **at most k times use** করে, **array** তে **minimum distinct elements** কয়টা বেছে থাকে , সেটা বের করা।

- **Hints:**

- **$K \leq N$**
- যেহেতু আমার **K limited** আমি চাইব, যত বেশি সংখ্যক **operation use** করে, **distinct number** এর **total count** কমানো।

- একটা **Xoss level** এর **observation** -

কোনো একটা **index j** এর উপর **operation** চালাতে পারব, তার **prefix index - i** এর help নিয়ে।

1 2 3 4 5 6 7 8

[index-4 can only be replaced by taking help from 1,2,3 indices]

আর **index - 1** কখনো **replace** করা **possible** না। তাই, আমি যায় করি না কেনো, **index-1** এর **element** টা থেকে যাবেই, আর তাহলে

minimum answer at least 1

তাহলে - এটা ধরে নাও, যাকেই তুমি **replace** করবা, **index-j** , **i-th index** হিসাবে **1st element** কেই **choose** করবা।

- আরেকটা **observation** , তোমার যেহেতু , **limited K operations** আছে, তাহলে, তোমার **must target** হবে, **1st element** এর help নিয়ে, **right side** যে **element-** এর **frequency** কম, তাকে আগে, **replace** করা, একটা একটা করে।
-

Problem - Max Minus Min

- **Question Analysis:**

আমাকে একটা **array A** দেওয়া আছে। আমি **any times operation** টা করতে পারব।

Operation:

- index **i** choose কর , **$A_i = 2 * A_i$** কর।

আমাকে **minimize** করতে হবে **$\max(\text{array} - A) - \min(\text{array} - A)$** .

- **Hints:**

- 2 দিয়ে **multiply** করা মানে, **bitwise least significant bit** এ 0 append করা।
- এখন, আমাদের কাজ **minimize** করা, তাহলে -

$$\begin{array}{rcllclcl} 57 & = & 1 & 1 & 1 & 0 & 0 & 1 \\ 3 & = & & & & & 1 & 1 \end{array}$$

3 কে আমরা ততক্ষণ, 2 দিয়ে গুণ করব, যতক্ষণ, এটা 57 থেকে সমান বা , **exact** ছোট মান না হয়। এমনভাবে -

$$\begin{array}{rcllclcl} 57 & = & 1 & 1 & 1 & 0 & 0 & 1 \\ 48 & = & 1 & 1 & 0 & 0 & 0 & 0 \end{array}$$

এতে লাভ হলো, **most significant bit** আমার **cancel out** হয়ে যাবে **minus** করলে।

- এভাবে, আমরা সব, **element** যেগুলো, **maximum element** থেকে ছোট, সেগুলোকে **convert** করব। তারপর, একটা **ans = (current_max - current_min)**
- আমরা কিন্তু আমাদের **ans** কিন্তু আরো, **minimize** করার **chance** থাকতে পারে।

ধর-

$$\begin{array}{ccc} 3 & 5 & 8 \end{array}$$

এখানে, কিন্তু, আমি যদি উপরের নিয়মে, **array convert** করি -

6 5 8

তাহলে - **answer** = $(8-5) = 3$.

আমরা যদি, আরেকবার **2 multiply** করি। তাহলে -

12 10 8

এখন - **answer** = $(12-8) = 2$

- তারমানে, যা দাঁড়াচ্ছে - আরেকবার, **2** গুণ করে , আমার আরো **best minimum ans** টা **find** করতে হবে।
-