Elite-S029-Numbers

Solved Challenges **0/2**

Back To Challenges List

Largest Possible Odd Integer

ID:11123 **Solved By 652 Users**

The program must accept an integer **N** as the input. The program must print the largest possible odd integer using all the digits in N as the output. If it is not possible to form such an integer, the program must print no as the output.

Boundary Condition(s):
10 <= N <= 10^17

Input Format:
The first line contains N.

Output Format:
The first line contains the largest possible odd integer using all the digits in N or no.

Example Input/Output 1:
Input:
120087460153

Output:
876543210001

Explanation:
The largest possible odd integer using all the digits in 120087460153 is **876543210001**.

Example Input/Output 2:
Input:
246228

Output:
no

Max Execution Time Limit: 500 millisecs

Ambiance

C (gcc 8.x)

Reset

```c
#include<stdio.h>
#include<stdlib.h>
#define ULL unsigned long long int

int main()
{
    ULL N;
    scanf("%llu",&N);
    int digits[10]={0};
    while(N!=0)
    {
        digits[N%10]++;
        N/=10;
    }

    int unitDig=-1;
    for(int dig=1;dig<=9;dig+=2)
    {
        if(digits[dig]>0)
        {
            unitDig = dig;
            digits[dig]--;
            break;
        }
    }
    if(unitDig==-1)
    {
        printf("no");
        return;
    }
    int start = 1;
    for(int digit=1;digit<=9;digit++)
    {
        if(digits[digit]>0)
        {
            start=0;
            break;
        }
    }
    for(int digit=9;digit>=start;digit--)
    {
        while(digits[digit]-->0)
        {
            printf("%d",digit);
        }
    }
    printf("%d",unitDig);

}
```

Code did not pass the execution

– ×

3 Private (Hidden) Test Cases Failed.

8 Passed 3 Failed

MEM: 0.09765625 MB **CPU: 0.01**

Save Run

□ Run with a custom test case (Input/Output)