'''

Pranav is playing with numbers 2 and 5 only.

He is given an integer N consist of 2 and 5 only.

Pranav is trying to find the maximum number by

altering atmost one digit in the given N

Help pranav to find the maximum number.

Input Format:

-------------

An integer N, consist of 2 and 5

Output Format:

--------------

Print the maximum number.

Sample Input-1:

---------------

5225

Sample Output-1:

----------------

5525

Explanation:

---------------

alter the 1st digit-> 2225

alter the 2nd digit-> 5525.

alter the 3rd digit-> 5255.

alter the 4th digit->5222.

So, The maximum number is 5525.

Sample Input-2:

---------------

52525

Sample Output-2:

----------------

55525

Sample Input-3:

---------------

2555

Sample Output-3:

----------------

5555

Write your python code below

'''

n=input()

l=[]

l.append(int(n))

for i in range(0,len(n)):

if(n[i]=='2'):

k=n[0:i]+'5'+n[i+1:len(n)]

l.append(int(k))

# n[i]='2'

else:

# n[i]='2'

k=n[0:i]+'2'+n[i+1:len(n)]

l.append(int(k))

# n[i]='5'

print(max(l))

A string is called well-weighted string,if and only if

the string has equal number of 'A's and 'B's in it.

You are given a string S, divide S in to the maximum number of well-weighted strings.

You should not leave any letter/part of the string.

Return the maximum number of well-weighted strings.

Input Format:

-------------

A string contains only A's and B's

Output Format:

--------------

Print an integer, maximum number of well-weighted strings

Sample Input-1:

---------------

ABBBBAAABA

Sample Output-1:

----------------

3

Explanation:

--------------

Well weighted strings, AB, BBBAAA, BA.

Sample Input-2:

---------------

ABAABBABAB

Sample Output-2:

----------------

4

Explanation:

--------------

Well weighted strings, AB, AABB, AB, AB.

Sample Input-3:

---------------

ABAAABBABB

Sample Output-3:

----------------

2

import java.util.\*;

public class Main{

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

String s=sc.next();

ArrayList<Integer> k=new ArrayList<>();

// int l=0;

// }

// public static int fun(s){

int u=0;

int len=0;

int counta=0;

int countb=0;

int ocount=0;

while(u<s.length()){

if(counta==countb && counta!=0){

ocount+=1;

counta=0;

countb=0;

}

if(s.charAt(u)=='A'){

counta+=1;

}

else{

countb+=1;

}

u+=1;

}

if(counta==countb && counta!=0){

ocount+=1;

}

System.out.println(ocount);

}

}

There is a switch-board made by an electrician,

If you turn on any two adjacent switches, it will cause short-circuit

and damage the switch-board.

You are given N integers(only 0's and 1's), Indiactes current status of

the switch board with N switches, where 1 indiactes switch is ON and

0 indiactes switch is OFF. And an integer K, more number of switches

to be turned ON.

Return true if and only if you can turn ON all the K switches, without

causing any damage to switch-board. Otherwise return fasle.

Input Format:

-------------

Line-1: Two integers N and K, number of switches, and more number of switches to be ON

Line-2: N space separated integers, only 0's and 1's.

Output Format:

--------------

Print a boolean value.

Sample Input-1:

---------------

5 1

1 0 0 0 1

Sample Output-1:

----------------

true

Sample Input-2:

---------------

5 2

1 0 0 0 1

Sample Output-2:

----------------

Flase

import java.util.\*;

public class Main{

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

int n1=sc.nextInt();

int n2=sc.nextInt();

int[] arr=new int[n1];

String s="";

for(int i=0;i<n1;i++){

arr[i]=sc.nextInt();

s+=String.valueOf(arr[i]);

}

int count=0;

// System.out.println(s);

String[] srr=s.split("1");

// System.out.println(Arrays.toString(srr));

if(srr[0].length()>=2){

count+=(srr[0].length()/2);

}

if(srr[srr.length-1].length()>=2 ){

if(arr[n1-1]!=1){

count+=(srr[srr.length-1].length()/2);

}

else{

if(srr[srr.length-1].length()%2==0){

count+=(srr[srr.length-1].length()/2)-1;

}

else{

count+=srr[srr.length-1].length()/2;

}

}

}

for(int i=1;i<srr.length-1 ;i++){

if(srr[i].length()>2){

if(srr[i].length()%2==0){

count+=(srr[i].length()/2)-1;

}

else{

count+=srr[i].length()/2;

}

}

}

if(count>=n2){

System.out.println(true);

}

else{

System.out.println(false);

}

}

}