**CHANDIGARH UNIVERSITYUNIVERSITYINSTITUTEOFENGINEERING**

**DEPARTMENTOFCOMPUTERSCIENCEANDENGINEERING**

**Submittedby:** Aniket Kumar (20BCS5306)

| | |
|---|---|
| **Section/Group** | 20BCS_WM_703/B |
| **SubjectName** | Competitive Coding |
| **SubjectCode** | 20CSP-314 |
| **Branch** | B.E.CSE |
| **Semester** | 5th |

## LAB INDEX

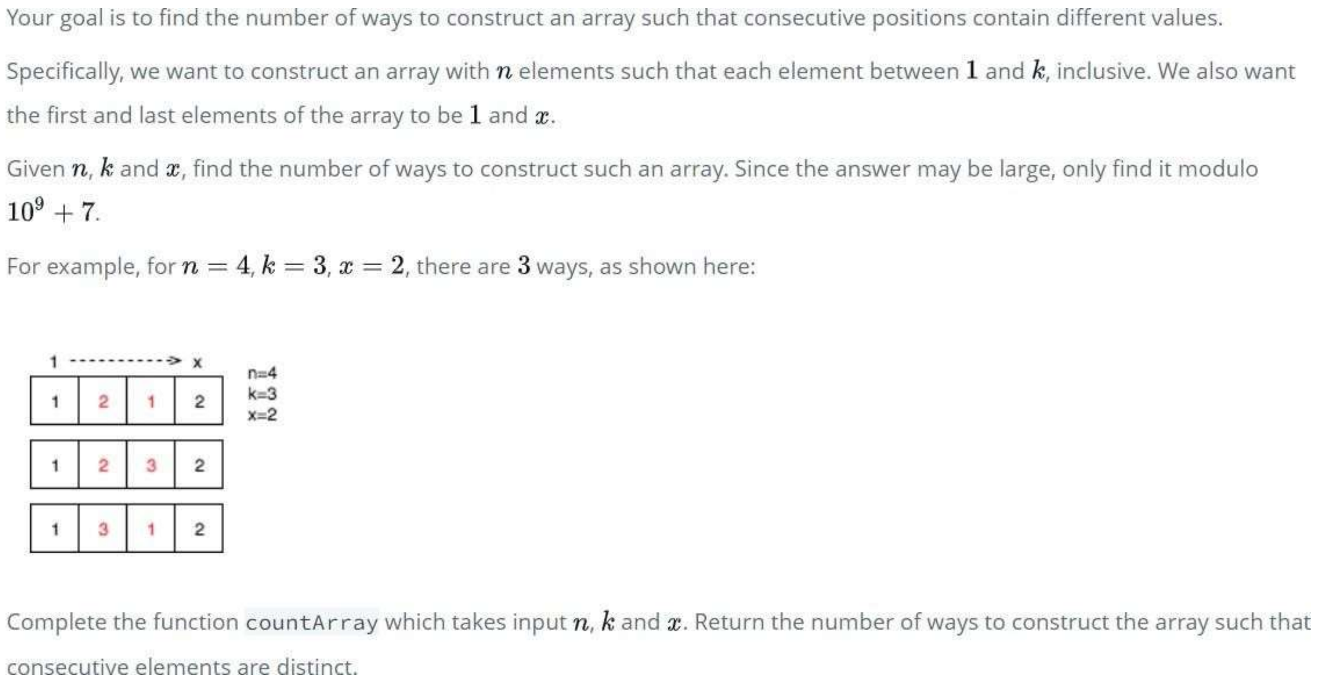
| Sr. No | Program | Date | Evaluation | | | | Sign |
|---|---|---|---|---|---|---|---|
| | | | LW (12) | VV (8) | FW (10) | Total (30) | |
| 1. | Todemonstratetheconceptofarrays | | | | | | |
| 2. | Todemonstratetheconceptofstackand queues | | | | | | |
| 3. | Todemonstrate theconcept ofLinkedlist | | | | | | |
| 4. | Todemonstratetheconcept ofSearchingandsorting | | | | | | |
| 5. | Todemonstrate theconcept ofGraphs | | | | | | |
| 6. | Todemonstrate theconcept ofTrees | | | | | | |
| 7. | Todemonstrate theconcept ofStrings | | | | | | |
| 8. | Todemonstratetheconceptof Dynamicprogramming | | | | | | |
| 9. | TodemonstratetheconceptofBacktracking | | | | | | |
| 10. | To demonstrate theconcept ofGreedyand Branchand Bound | | | | | | |

## EXPERIMENTNUMBER–8

## 8.1 AIMOFTHEEXPERIMENT–

DynamicProgramming

## TASKTOBEDONE/LOGISTICSUSED--



Your goal is to find the number of ways to construct an array such that consecutive positions contain different values.

Specifically, we want to construct an array with $n$ elements such that each element between $1$ and $k$, inclusive. We also want the first and last elements of the array to be $1$ and $x$.

Given $n$, $k$ and $x$, find the number of ways to construct such an array. Since the answer may be large, only find it modulo $10^9 + 7$.

For example, for $n = 4$, $k = 3$, $x = 2$, there are $3$ ways, as shown here:

1 ----------> x

| | | | |
|---|---|---|---|
| 1 | 2 | 1 | 2 |
| 1 | 2 | 3 | 2 |
| 1 | 3 | 1 | 2 |

n=4<br>
k=3<br>
x=2

Complete the function `countArray` which takes input $n$, $k$ and $x$. Return the number of ways to construct the array such that consecutive elements are distinct.

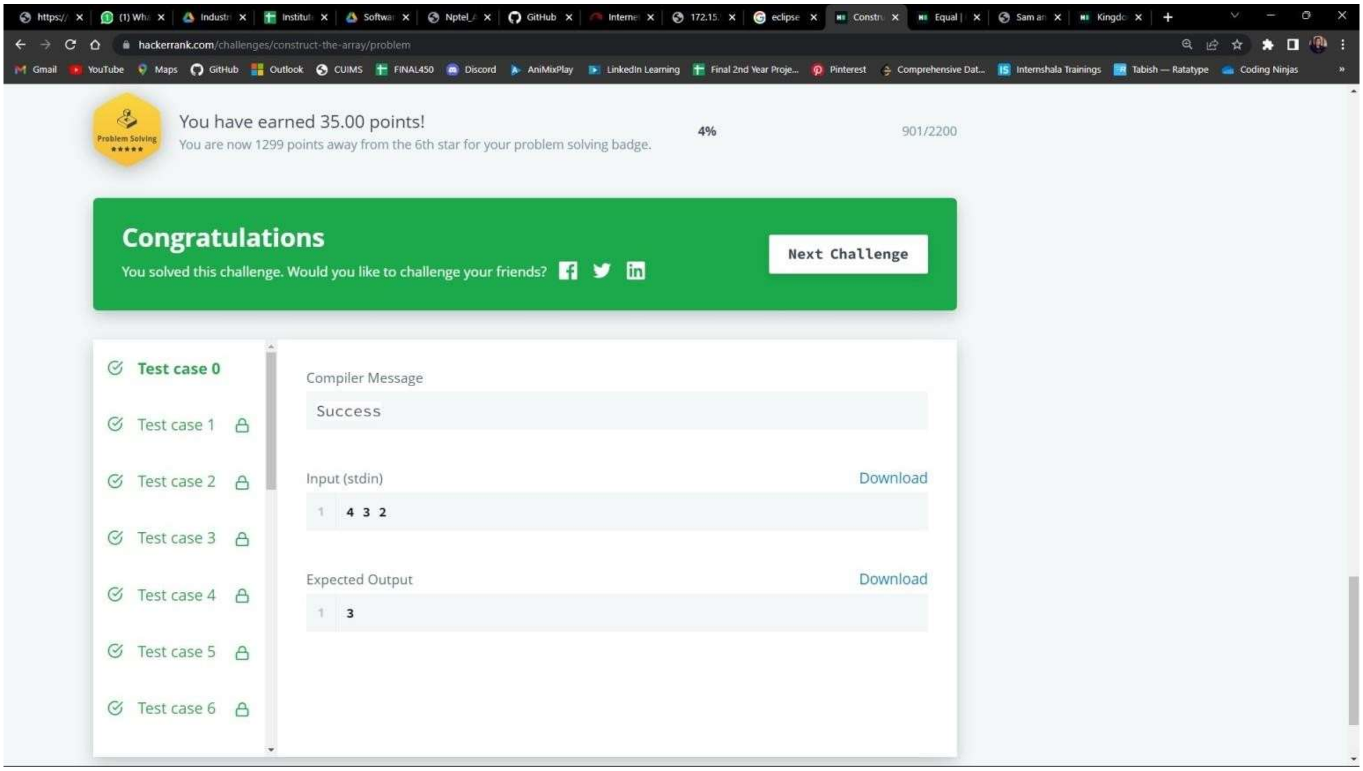
https://www.hackerrank.com/challenges/construct-the-array/problem?isFullScreen=false

## CODE:(INC++)

```
#include<bits/stdc++.h>u
singnamespace std;
```

```
#define ll long
long#defineMOD10000000
07

int
main(){ios_base::sync_with_stdio(
false);cin.tie(NULL);
intt=1;

while(t--){
    ll
    n,k,x;cin>>n>
    >k>>x;
    vector<ll>path(n+1),dp(n+1);p
    ath[1]=1;
    if(x==1)
    dp[1]=1;
    elsedp[1
    ]=0;
    for(ll
        i=2;i<=n;i++){path[i]=(path[i-
        1]*(k-1))%MOD;
        dp[i]=(path[i-1]-dp[i-1]+MOD)%MOD;
    }
    cout<<dp[n]<<endl;


}

return0;
}
```

**OUTPUT:**



You have earned 35.00 points!

You are now 1299 points away from the 6th star for your problem solving badge.

4%

901/2200

**Congratulations**

You solved this challenge. Would you like to challenge your friends?

Test case 0

Test case 1

Test case 2

Test case 3

Test case 4

Test case 5

Test case 6

Compiler Message

```
Success
```

Input (stdin)

```
4 3 2
```

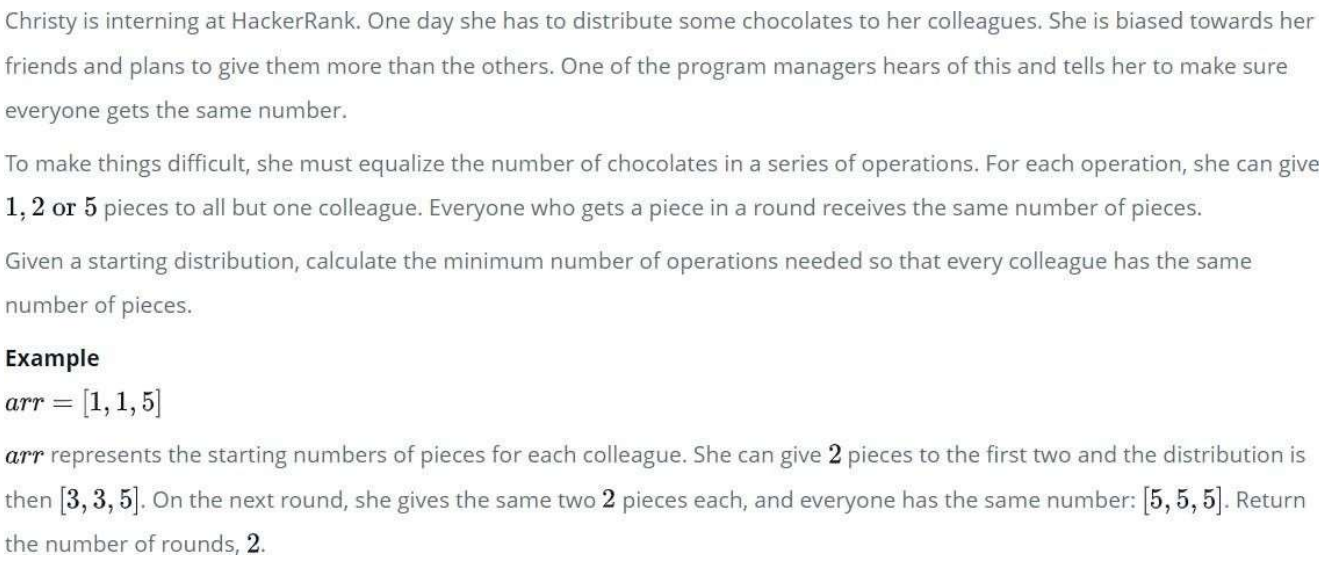
Expected Output

```
3
```

## 8.2 AIMOFTHEEXPERIMENT–

DynamicProgramming

## TASKTOBEDONE/LOGISTICSUSED–



Christy is interning at HackerRank. One day she has to distribute some chocolates to her colleagues. She is biased towards her friends and plans to give them more than the others. One of the program managers hears of this and tells her to make sure everyone gets the same number.

To make things difficult, she must equalize the number of chocolates in a series of operations. For each operation, she can give $1, 2$ or $5$ pieces to all but one colleague. Everyone who gets a piece in a round receives the same number of pieces.

Given a starting distribution, calculate the minimum number of operations needed so that every colleague has the same number of pieces.

**Example**

$arr = [1, 1, 5]$

$arr$ represents the starting numbers of pieces for each colleague. She can give $2$ pieces to the first two and the distribution is then $[3, 3, 5]$. On the next round, she gives the same two $2$ pieces each, and everyone has the same number: $[5, 5, 5]$. Return the number of rounds, $2$.

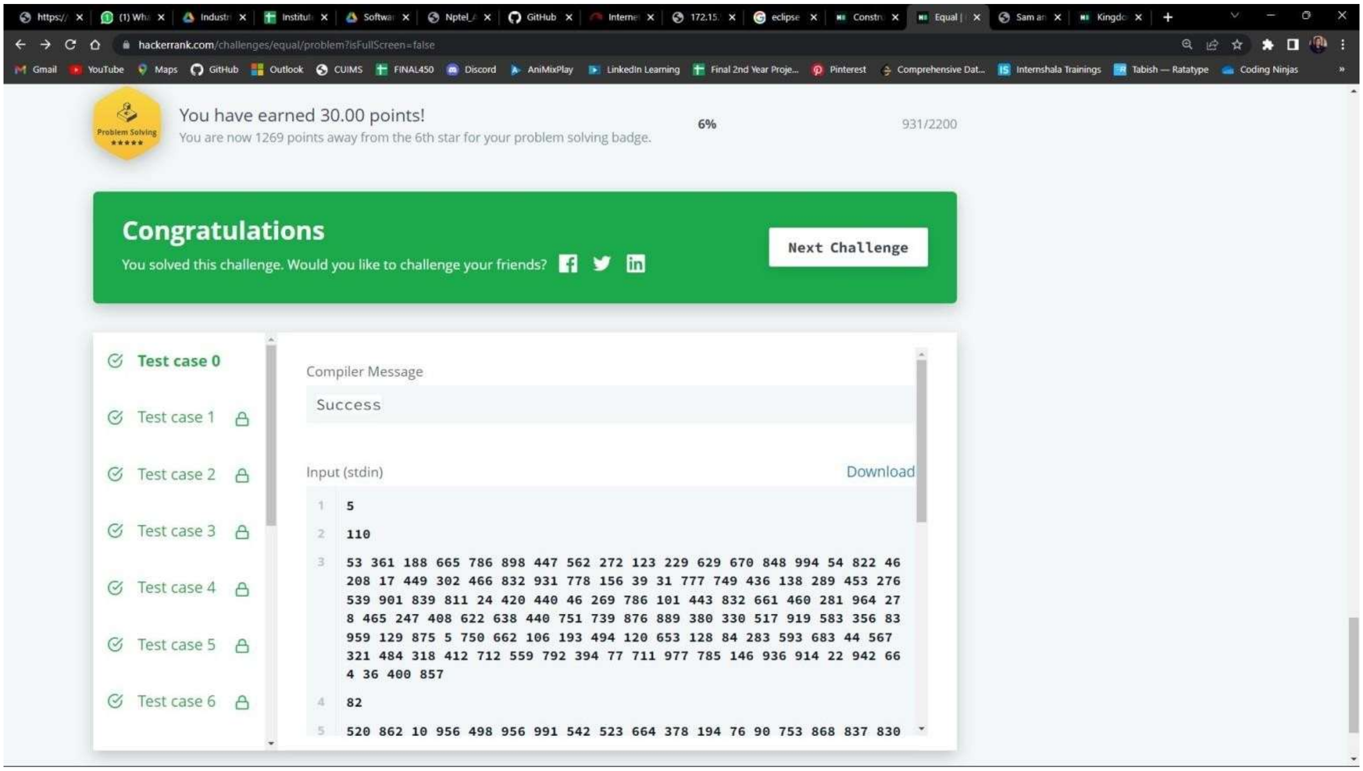
https://www.hackerrank.com/challenges/equal/problem?isFullScreen=false

**CODE:(INC++)**

```
#include<bits/stdc++.h>u
singnamespace std;
inta[3]={1,2,5};
intdp[2001];
intgetans(intx)
 {
      if(x==0)
       return
       dp[x]=0;if(dp[x
       ]!=-1)
         returndp[x];intm
       ini=10000001;
       for(inti=0;i<3;i++)
       {
           if(a[i]<=x)mini=min(mini,1+getans(x
              -a[i]));
       }
       returndp[x]=mini;
 }
int equal(vector<int> arr)
   {int mins=10000001;
   for(inti=0;i<arr.size();i++)
```

```
    {
        if(arr[i]<mins)
         mins=arr[i];
    }
    intans=10000001;
     for(intj=mins-4;j<=mins;j++)
     {
         intans1=0;
     for(inti=0;i<arr.size();i++)
     {
         ans1+=dp[arr[i]-j];
     }
      ans=min(ans,ans1);
     }
     returnans;
}

intmain()
{
    int
    t;cin>>
    t;
    memset(dp,-
     1,sizeof(dp));for(inti=0
     ;i<2001;i++)
     {
         getans(i);
     }
     while(t--)
     {
         int
         n;cin>>
         n;
         vector<int>arr(n);fo
         r(inti=0;i<n;i++)
           cin>>arr[i];cout<<equal
           (arr)<<endl;
     }
     return0;
}
```

**OUTPUT:**



You have earned 30.00 points!
You are now 1269 points away from the 6th star for your problem solving badge.

Congratulations
You solved this challenge. Would you like to challenge your friends?

Test case 0
Test case 1
Test case 2
Test case 3
Test case 4
Test case 5
Test case 6

Compiler Message

```
Success
```

Input (stdin)

```
5
110
53 361 188 665 786 898 447 562 272 123 229 629 670 848 994 54 822 46
208 17 449 302 466 832 931 778 156 39 31 777 749 436 138 289 453 276
539 901 839 811 24 420 440 46 269 786 101 443 832 661 460 281 964 27
8 465 247 408 622 638 440 751 739 876 889 380 330 517 919 583 356 83
959 129 875 5 750 662 106 193 494 120 653 128 84 283 593 683 44 567
321 484 318 412 712 559 792 394 77 711 977 785 146 936 914 22 942 66
4 36 400 857
82
520 862 10 956 498 956 991 542 523 664 378 194 76 90 753 868 837 830
```

## 8.3 AIMOFTHEEXPERIMENT–

DynamicProgramming

## TASKTOBEDONE/LOGISTICSUSED–

Samantha and Sam are playing a numbers game. Given a number as a string, no leading zeros, determine the sum of all integer values of substrings of the string.

Given an integer as a string, sum all of its substrings cast as integers. As the number may become large, return the value modulo $10^9 + 7$.

**Example**

$n = \text{'42'}$

Here $n$ is a string that has $3$ integer substrings: $4$, $2$, and $42$. Their sum is $48$, and $48 \bmod (10^9 + 7) = 48$.

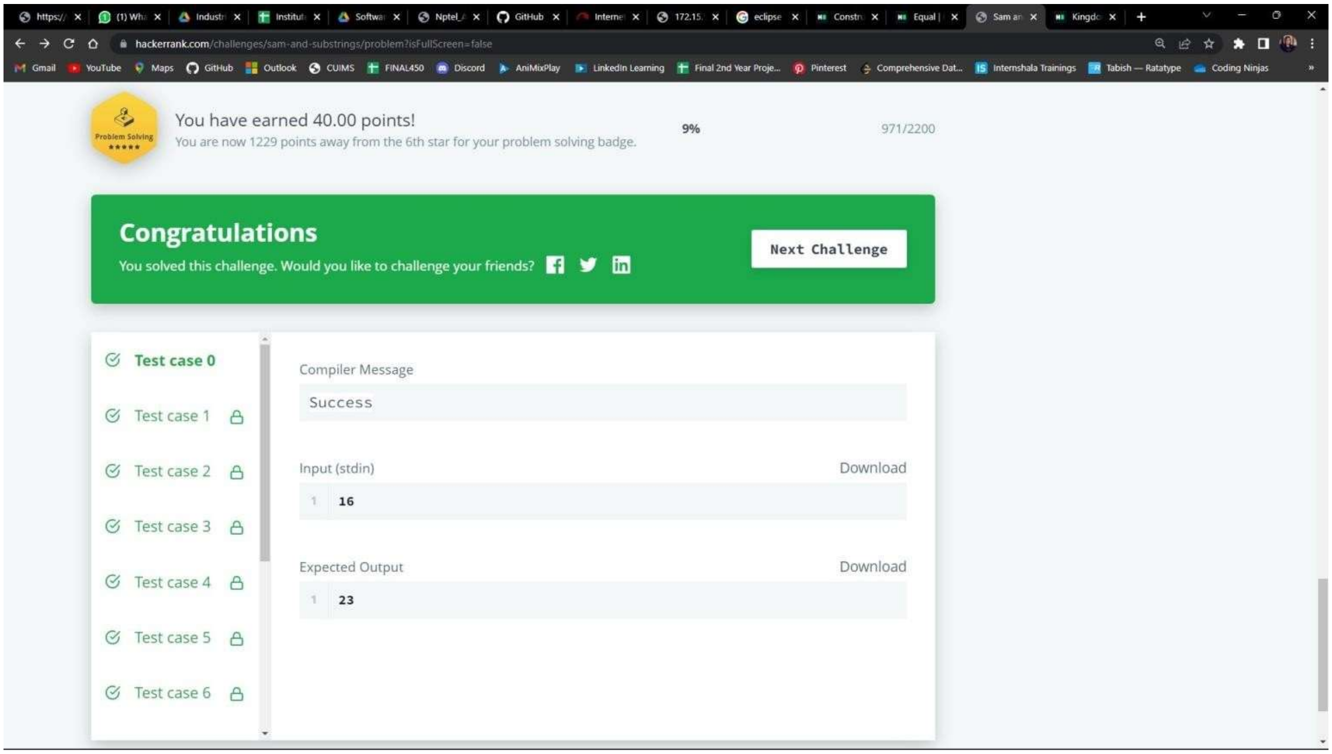
https://www.hackerrank.com/challenges/sam-and-substrings/problem?isFullScreen=false

## CODE:(INC++)

```
#include<bits/stdc++.h>
```

```
usingnamespacestd;

#definelllonglong#de
finearray

constintmxN=2e5,M=1e9+7;int
n;
stringt;lld
p[mxN];
intmain(){
    cin >>
    t;n=t.size();dp
    [0]=t[0]-'0';
    for(inti=1; i<n; ++i)
        dp[i]=(dp[i-1]*10%M+(t[i]-'0')*(i+1))%M;
    llans=0;
    for(int i=0; i<n;
        ++i)ans=(ans+dp[i]
        )%M;
    cout<<ans;
}
```

**OUTPUT:**



You have earned 40.00 points!

You are now 1229 points away from the 6th star for your problem solving badge.

9%

971/2200

Congratulations

You solved this challenge. Would you like to challenge your friends?

Test case 0

Test case 1

Test case 2

Test case 3

Test case 4

Test case 5

Test case 6

Compiler Message

```
Success
```

Input (stdin)

```
16
```

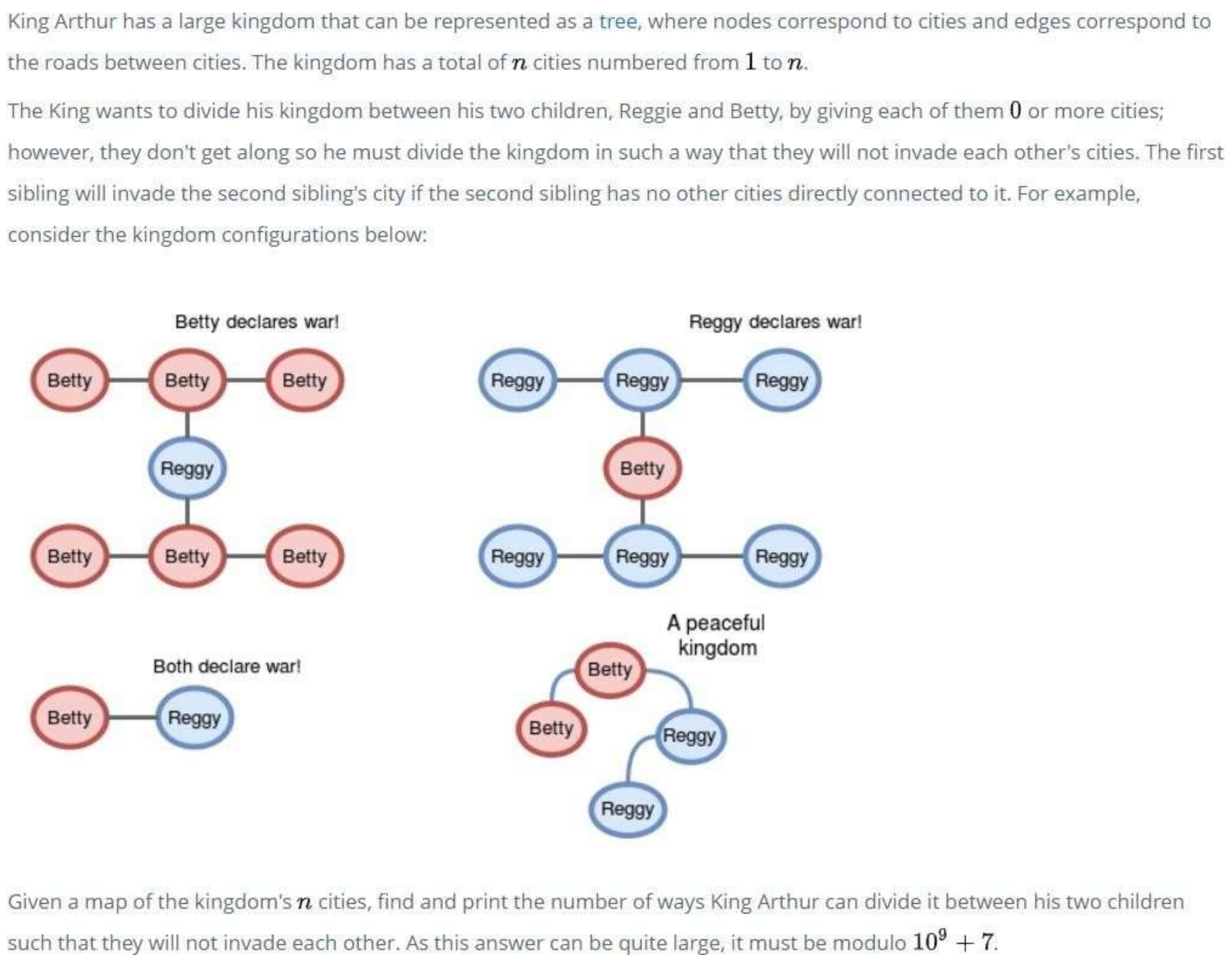
Expected Output

```
23
```

## 8.4 AIMOFTHEEXPERIMENT–

DynamicProgramming

## TASKTOBEDONE/LOGISTICSUSED--



King Arthur has a large kingdom that can be represented as a tree, where nodes correspond to cities and edges correspond to the roads between cities. The kingdom has a total of $n$ cities numbered from $1$ to $n$.

The King wants to divide his kingdom between his two children, Reggie and Betty, by giving each of them $0$ or more cities; however, they don't get along so he must divide the kingdom in such a way that they will not invade each other's cities. The first sibling will invade the second sibling's city if the second sibling has no other cities directly connected to it. For example, consider the kingdom configurations below:

Given a map of the kingdom's $n$ cities, find and print the number of ways King Arthur can divide it between his two children such that they will not invade each other. As this answer can be quite large, it must be modulo $10^9 + 7$.

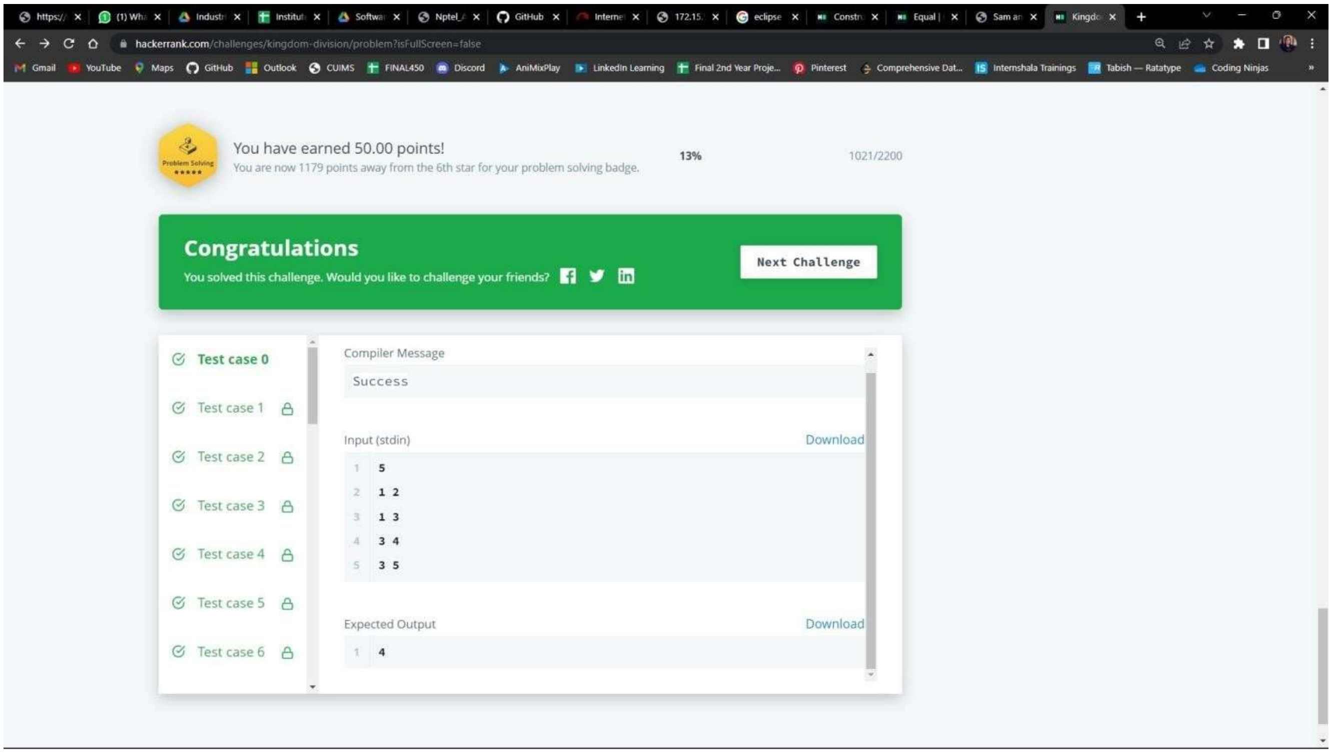
https://www.hackerrank.com/challenges/kingdom-division/problem?isFullScreen=false

## CODE:(inC++)

```
#include<iostream>#
include<vector>
```

```
#define NODE

100000usingnamespace

std;

vector<int>
adj[NODE+10];intmod=(int)
1e9+7;
intmemo[NODE+10][2][2];
intdone[NODE+10][2][2];

longlongdp(int,int,int,int,int);int

main(){
    intn,a,b;ci
    n>>n;

    for(int i = 1; i < n; i++)
        {cin >> a >>
        b;adj[a].push_back(b);ad
        j[b].push_back(a);
    }

    cout<<(dp(1,0,0,0,0)+dp(1,0, 0,1,0))%mod<<endl;
    return0;
}

longlongdp(ints,intp,intc,intcolor,intally){longlongres=

    0;

    if(c==adj[s].size())ret
        urnally;

    intt=adj[s][c];if

    (t == p)
        returndp(s,p, c+ 1, color,ally); /// skipbackedges

    if (done[t][color][ally] ==
        1)returnmemo[t][color][ally]
        ;

    ///makeitally
    res =(dp(t,s,0, color,1) *dp(s,p, c+ 1,color,1) )% mod;
    ///makeitenemy
    res=(res+(dp(t,s,0,1-
 color, 0) *dp(s, p, c+1,color, ally) )% mod)% mod;

    done[t][color][ally] =
    1;memo[t][color][ally]=(int)res;

    returnres;
}
```

## OUTPUT



You have earned 50.00 points!

You are now 1179 points away from the 6th star for your problem solving badge.

13% 1021/2200

**Congratulations**

You solved this challenge. Would you like to challenge your friends?

Next Challenge

✓ Test case 0
✓ Test case 1
✓ Test case 2
✓ Test case 3
✓ Test case 4
✓ Test case 5
✓ Test case 6

Compiler Message

```
Success
```

Input (stdin) Download

```
5
1 2
1 3
3 4
3 5
```

Expected Output Download

```
4
```

**LearningOutcomes(WhatIhavelearnt):**

- Itwillprovidethemodestexperiencethatallowsstudentstodevelopandimprovetheire xperimentalskillsand developability to analyzedata.
- Ability to demonstrate the practical skill on measurements and instrumentationtechniques of some Physics experiments. Students will develop the ability to useappropriatephysical conceptstoobtainquantitativesolutionstoproblemsinphysics.
- Studentswilldemonstratebasicexperimentalskillsbysettinguplaboratoryequipment safely and efficiently, plan and carry out experimental procedures, andreportverballyandin written languagetheresultsoftheexperiment.
- Studentswilldevelopskillsbythepracticeofsettingupandconductinganexperiment with dueregardsto minimizing measurementerror.