

X


<https://swayam.gov.in>

https://swayam.gov.in/nc_details/NPTEL

souravsharma2468@gmail.com ▾

NPTEL (<https://swayam.gov.in/explorer?ncCode=NPTEL>) » **Design and analysis of algorithms (course)**

Announcements (announcements)

About the Course (https://swayam.gov.in/nd1_noc20_cs27/preview) Ask a Question (forum)

Progress (student/home) Mentor (student/mentor)

Course outline

How does an NPTEL online course work?

 Week 1 :
Introduction

 Week 1 :
Analysis of algorithms

Week 1 Quiz

 Week 2 :
Searching and sorting

Week 2 Quiz

 Week 2
Programming Assignment

 Week 3 :
Graphs

Week 3 Programming Assignment: Road Trips and Museums

Due on 2020-02-24, 23:59 IST

- Select your language (C/C++/Java/Python3)
- Paste your code into the submission window.
- There are some public test cases and some (hidden) private test cases.
- "Compile and run" will evaluate your submission against the public test cases.
- "Submit" will evaluate your submission against the hidden private test cases. There are 10 private testcases in all, each with equal weightage. You will get feedback about which private test cases pass or fail, though you cannot see the actual test cases.
- Ignore warnings about "Presentation errors".

Road Trips and Museums

Indian National Olympiad in Informatics 2018
<https://www.iarcs.org.in/inoi/2018/>

Lavanya and Nikhil have **K** months of holidays ahead of them, and they want to go on exactly **K** road trips, one a month. They have a map of the various cities in the world with the roads that connect them. There are **N** cities, numbered from 1 to **N**. We say that you can reach city B from city A if there is a sequence of roads that starts from city A and ends at city B. Note that the roads are bidirectional. Hence, if you can reach city B from city A, you can also reach city A from city B.

Lavanya first decides which city to start from. In the first month, they will start from that city, and they will visit every city that they can reach by road from that

Week 3 Quiz

Week 3 Programming Assignment

Week 3 Programming Assignment: Road Trips and Museums (/noc20_cs27/progasslavaranya?name=105)

Week 4 : Weighted graphs

Week 4 Quiz

Week 4 Programming Assignment

Week 5: Data Structures: Union-Find and Heaps

Week 5 : Divide and Conquer

Week 5 Quiz

Week 6: Data Structures: Search Trees

Week 6: Greedy Algorithms

Week 6 Quiz

Week 6 Programming Assignment

Week 7: Dynamic Programming

Week 7 Quiz

particular city, even if it means that they have to pass through cities that they have already visited previously. Then, at the beginning of the second month, Nikhil picks a city that they haven't visited till then. In the second month, they first fly to that city and visit all the cities that they can reach from that city by road. Then, in the third month, Lavanya identifies a city, and they fly there and visit all cities reachable from there by road. Then in the fourth month it is Nikhil's turn to choose an unvisited city to start a road trip, and they alternate like this. Note that the city that they fly to (that is, the city from where they start each month's road trip) is also considered as being visited.

Each city has some museums, and when they visit a city for the first time, Lavanya makes them visit each of the museums there. Lavanya loves going to museums, but Nikhil hates them. Lavanya always makes her decisions so that they visit the maximum number of museums possible that month, while Nikhil picks cities so that the number of museums visited that month is minimized.

Given a map of the roads, the number of museums in each city, and the number K , find the total number of museums that they will end up visiting at the end of K months. Print -1 if they will have visited all the cities before the beginning of the K^{th} month, and hence they will be left bored at home for some of the K months.

Solution hint

Use BFS/DFS to identify the connected components of the underlying undirected graph. Sort the components based on how many museums they contain and make appropriate choices for Lavanya and Nikhil each month.

Input format

- The first line of each testcase contains three integers: N , M and K , which represents the number of cities, number of roads and the number of months.
- The i^{th} of the next M lines contains two integers, u_i and v_i . This denotes that there is a direct road between city u_i and city v_i .
- The next line contains N integers, the i^{th} of which represents the number of museums in city i .

Output format

If they can go on K road trips, output a single line containing a single integer which should be the total number of museums they visit in the K months. Output -1 if they can't go on K road trips.

Constraints

- $1 \leq T \leq 3$
- $1 \leq N \leq 10^6$
- $0 \leq M \leq 10^6$
- $1 \leq K \leq 10^6$
- $1 \leq u_i, v_i \leq N$

**Week 7
Programming
Assignment****Week 8: Linear
Programming
and Network
Flows****Week 8:
Intractability****Week 8 Quiz****Text Transcripts****Books****Download
Videos**

- There is no road which goes from one city to itself. ie. $u_i \neq v_i$.
- There is at most one direct road between a pair of cities.
- $0 \leq$ Number of museums in each city ≤ 1000
- Sum of **N** over all testcases in a file will be $\leq 1.5 * 10^6$

Sample input 1

```
10 10 3
1 3
3 5
5 1
1 6
6 2
5 6
2 5
7 10
4 7
10 9
20 0 15 20 25 30 30 150 35 20
```

Sample output 1

```
345
```

Sample input 2

```
10 10 2
1 3
3 5
5 1
1 6
6 2
5 6
2 5
7 10
4 7
10 9
20 0 15 20 25 30 30 150 35 20
```

Sample output 2

```
240
```

Sample input 3

```

10 10 5
1 3
3 5
5 1
1 6
6 2
5 6
2 5
7 10
4 7
10 9
20 0 15 20 25 30 30 150 35 20

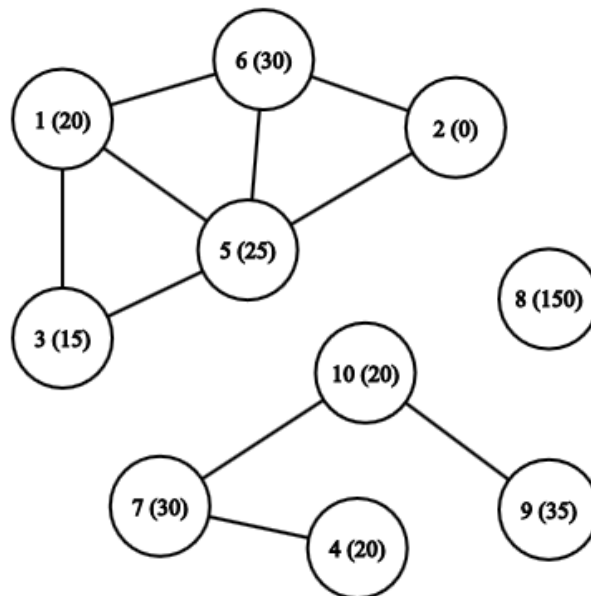
```

Sample output 3

-1

Explanation

Notice that in all the three testcases, everything is the same, except for the value of **K**. The following figure represents the road map in these testcases. Each node denotes a city, with a label of the form " $n (m)$ ", where n is the city number, between 1 and **N**, and m is the number of museums in this city. For example, the node with label "5 (25)" represents city 5, which has 25 museums.



Testcase 1: Lavanya will first choose to fly to city 8. In the first month, they visit only that city, but they visit 150 museums.

Then in the second month, Nikhil could choose to fly to city 3, and they visit the cities 1, 2, 3, 5 and 6, and visit $20 + 0 + 15 + 25 + 30 = 90$ museums that month. Note that Nikhil could have instead chosen to fly to city 1 or 2 or 5 or 6, and they would all result in the same scenario.

Then, Lavanya could choose city 7, and in the third month they will visit the cities 7, 4, 10 and 9. Note that Lavanya could have chosen to fly to city 4 or 10 or 9, and they would all result in the same scenario.

In total, they have visited 345 museums in the three months (which is in fact all the museums), and that is the answer.

Testcase 2: It is same as the previous testcase, but now they have only 2 months. So they visit only $150 + 90 = 240$ museums in total.

Testcase 3: It is same as the previous testcase, but now they have 5 months of holidays. But sadly, they finish visiting all the cities within the first three months itself, and hence the answer is -1.

Private

Test

cases Input
used for
evaluation

Expected Actual
Output Output Status

Test Case
1

```
10000 0 1521
712 383 959 764 131 522 978 634 7
94 165 597 387 623 745 912 167 31
284 254 572 315 250 294 158 911 6
06 851 391 715 306 903 776 353 14
5 685 22 129 985 717 478 405 11 6
83 106 630 352 763 917 985 465 25
3 505 813 944 543 233 973 141 491
840 2 474 652 818 495 810 86 727
278 675 621 407 505 355 523 377 4
44 180 482 319 96 376 999 754 902
439 354 387 202 534 237 397 435 1
76 248 830 132 968 746 165 450 92
0 646 142 450 226 880 838 734 583
814 958 563 229 768 467 156 814 9
97 392 12 18 762 470 487 34 254 7
13 732 819 367 894 217 810 237 86
2 925 172 43 949 301 541 431 771
628 895 159 880 915 995 449 548 4
11 513 792 224 407 933 376 853 91
1 468 373 152 70 998 765 7 532 92
4 798 909 694 761 405 882 494 306
687 254 621 924 717 749 358 57 20
8 71 198 573 269 122 350 657 804
909 744 436 346 310 448 792 779 7
7 435 943 999 689 508 327 2 31 46
8 58 493 534 817 200 29 6 975 420
557 324 946 349 831 63 346 390 79
263 90 270 829 937 312 939 919 57
0 775 512 788 207 623 866 334 306
27 359 334 447 267 892 851 40 998
550 560 165 952 666 437 616 21 59
5 923 117 821 277 991 174 720 696
```

7619
24\n

76
19
24
\n

Pa
ss
ed

```

5 667 58 20 650 478 563 279 635 4
1 818 403 712 338 132 424 412 417
96 437 2 722 599 934 92 582 791 3
8 744 232 153 600 184 33 24 932 2
38 966 81 186 637 293 167 192 325
492 250 124 437 295 653 469 935 3
62 706 452 59 755 955 275 788 996
695 878 66 865 182 127 918 809 87
9 64 72 74 568 870 460 403 171 25
0 609 936 779 575 150 388 771 383
954 961 882 510 431 552 551 782 1
48 60 396 47 541 610 235 250 809
718 708 686 998 524 374 754 484
418 814 999 875 55 837 187 420 2
55 439 653 564 669 334 452 516 91
4 820 951 88 447 674 733 752 424
20 904 537 382 730 540 138 628 4
38 588 975 229 16 695 679 998 644
293 292 781 204 869 923 470 936 6
76 182 173 365 715 987 127 226 19
6 142 752 111 763 822 143 430 279
100 713 598 798 407 839 282 282 3
32 801 304 771 592 160 962 286 77
7 838 775 527 697 417 748 350 131
692 947 774 254 313 948 987 886 1
23 19 853 735 45 94 916 823 218 8
47 959 309 874 587 802 700 482 32
0 35 718 751 856 688 806 130 860
974 826 516 557 990 383 699 158
616 354 770 618 928 686 519 81 1
72 78 759 579 539 405 558 62 516
687 550 842 77 415 11 273 156 84
9 96 607 992 476 928 244 318 414
788 120 60 867 228 336 919 6 298
664 889 952 635 249 702 629 425 8
89 588 579 540 86 354 98 361 376
118 838 100 326 470 386 264 454
575 998 464 924 783 795 660 753
980 89 343 250 978 711 429 438 1
73 676 482 844

```

The due date for submitting this assignment has passed.

10 out of 10 tests passed.

You scored 100.0/100.

Assignment submitted on 2020-02-23, 10:58 IST

Your last recorded submission was :

```

1 #include <bits/stdc++.h>
2 using namespace std;
3
4 #define print(x) cerr << #x << " is " << x << endl;
5 #define int long long int
6 #define vi vector<int>

```

```

7 #define vvi vector<vector<int>>
8 #define all(v) v.begin(), v.end()
9 #define pb push_back
10
11
12 vvi g;
13 vi val, vis;
14 int sum=0;
15
16 void dfs(int u){
17     if(vis[u])return;
18     vis[u]=1,sum+=val[u];
19     for(int v: g[u])
20         if(not vis[v])dfs(v);
21 }
22
23
24 int32_t main()
25 {
26     ios::sync_with_stdio(false);
27     cin.tie(0);
28
29     int n,m,k;
30     cin>>n>>m>>k;
31
32     g=vvi(n+1),val=vi(n+1);
33
34     for(int i=0;i<m;++i){
35         int u,v;
36         cin>>u>>v;
37         g[u].pb(v),g[v].pb(u);
38     }
39
40     for(int i=1;i<=n;++i)cin>>val[i];
41
42
43     vi ans;
44     vis=vi(n+1,0);
45     for(int i=1;i<=n;++i){
46         if(not vis[i]){
47             sum=0;
48             dfs(i);
49             ans.pb(sum);
50         }
51     }
52
53     sort(all(ans));
54     int fans=0;
55     if(ans.size()<k)cout<<"-1\n";
56     else{
57         int n=ans.size();
58         int p=k/2;
59         for(int i=0;i<p;++i)fans+=ans[i];
60         if(k%2)p++;
61
62         for(int i=n-1;i>=n-p;--i)fans+=ans[i];
63
64         cout<<fans<<"\n";
65     }
66
67     return 0;
68 }
69

```

