



Kalakeya

Problem Code: **AMR15C**

Submit (/submit/AMR15C)



[Tweet](#) [Like](#) [Share](#) Be the first of your friends to like this.

All submissions for this problem are available.

The **Kalakeyas** were a powerful, ferocious and cruel clan of Danavas. They were known to be really strong and they did not have any war strategy. They would just attack the enemy randomly and overpower them with sheer number of soldiers. However, we all know that Baahubali and Bhallaladeva defeated the Kalakeyas by following the **Thrishul** strategy, and successfully defended their kingdom Maahishmati. We also know that Baahubali was very smart, and the truth is that he predicted how the Kalakeyas would attack and devised a counter strategy for the same, the night before the war. This is what he found:

The Kalakeyas had **N** forts, numbered **1** to **N** and Baahubali had **N** soldiers, numbered **1** to **N**. Baahubali discovered that he can permute his soldiers in any way to get a permutation of **1** to **N** => **P₁**, **P₂**, ..., **P_N**. He would then send his soldiers to attack the forts in the following way: soldier **P₁** attacks **fort 1**, soldier **P₂** attacks **fort 2**, ..., soldier **P_N** attacks **fort N**. It is easy to note that each soldier attacks exactly one fort and no two soldiers attack the same fort. Baahubali also got to know about a secret key of the Kalakeyas, which is an integer **K**. A soldier **X** can destroy a fort **Y**, iff **abs(X - Y) >= K**. For more details on the **abs()** function, check [here](https://en.wikipedia.org/wiki/Absolute_value) (https://en.wikipedia.org/wiki/Absolute_value).

Your task is to determine whether Baahubali's soldiers can be permuted in some way, such that all forts can be destroyed. In other words, for a permutation **P₁**, **P₂**, ..., **P_N**, Baahubali's soldiers can destroy all the forts iff **abs(P_i - i) >= K**, for all **1 <= i <= N**. If this is possible, you are also required to output the **lexicographically** smallest such permutation. If it is not possible, output **-1**.

Note: A permutation **A₁**, **A₂**, ..., **A_N** is said to be lexicographically smaller than a permutation **B₁**, **B₂**, ..., **B_N**, if and only if at the first **i** where **A_i** and **B_i** differ, **A_i** comes before **B_i**. You can refer [here](https://en.wikipedia.org/wiki/Lexicographical_order) (https://en.wikipedia.org/wiki/Lexicographical_order) for a more detailed definition of lexicographic ordering.

Input

The first line of input consists of a single integer **T** denoting the number of test cases. Each of the following **T** lines contain two space separated integers **N** and **K** denoting the values mentioned in the statement above.

Output

For each test case, output a single line containing **N** space separated integers (which should be a permutation of **[1..N]**, if Baahubali's soldiers can break all the forts. If it is not possible to break all the forts, output **-1** (quotes for clarity).

Constraints

- $1 \leq T \leq 1000$
- $1 \leq N \leq 10^5$
- $0 \leq K \leq N$
- The sum of **N** over all test cases in a single test file will not exceed 10^5

Example

My Submissions

(/status/AMR15C,asprazz2658)

All Submissions

(/status/AMR15C)

Successful Submissions



Input :

```
3
2 2
3 0
3 1
```

Output :

```
-1
1 2 3
2 3 1
```

Explanation

For the first test case, $N = 2$ and $K = 2$. It is impossible to permute **[1, 2]** in any way such that **$\text{abs}(P[1]-1) \geq 2$ and $\text{abs}(P[2]-2) \geq 2$** . Hence, output is **-1**.

For the second test case, $N = 3$ and $K = 0$. We can just set **$P[i] = i$** , and hence the answer is **1 2 3**

For the third case, the valid permutations are **[2, 3, 1]** and **[3, 1, 2]**. The answer is **[2, 3, 1]** since it is lexicographically smaller than **[3, 1, 2]**.

Author: 5★ [suh_ash2008 \(/users/suh_ash2008/\)](/users/suh_ash2008/)

Editorial: <http://discuss.codechef.com/problems/AMR15C>
(<http://discuss.codechef.com/problems/AMR15C>)

Tags: [acmamr15 \(/tags/problems/acmamr15/\)](/tags/problems/acmamr15/) [constructive \(/tags/problems/constructive/\)](/tags/problems/constructive/) [greedy \(/tags/problems/greedy/\)](/tags/problems/greedy/) [implementation \(/tags/problems/implementation/\)](/tags/problems/implementation/) [medium \(/tags/problems/medium/\)](/tags/problems/medium/) [suh_ash2008 \(/tags/problems/suh_ash2008/\)](/tags/problems/suh_ash2008/)

Date Added: 11-10-2015

Time Limit: 1 secs

Source Limit: 50000 Bytes

Languages: ADA, ASM, BASH, BF, C, C99 strict, CAML, CLOJ, CLPS, CPP 4.3.2, CPP 6.3, CPP14, CS2, D, ERL, FORT, FS, GO, HASK, ICK, ICON, JAVA, JS, LISP clisp, LISP sbcl, LUA, NEM, NICE, NODEJS, PAS fpc, PAS gpc, PERL, PERL6, PHP, PIKE, PRLG, PYPY, PYTH, PYTH 3.5, RUBY, SCALA, SCM chicken, SCM guile, SCM qobi, ST, TCL, TEXT, WSPC

[Submit \(/submit/AMR15C/\)](/submit/AMR15C/)

Comments ▶

[CodeChef is a non-commercial competitive programming community](#)

[About CodeChef \(http://www.codechef.com/aboutus/\)](http://www.codechef.com/aboutus/) | [About Directi \(http://www.directi.com/\)](http://www.directi.com/) | [CEO's Corner \(http://www.codechef.com/ceoscorner/\)](http://www.codechef.com/ceoscorner/) | [C-Programming \(http://www.codechef.com/c-programming/\)](http://www.codechef.com/c-programming/) | [Programming Languages \(http://www.codechef.com/Programming-Languages/\)](http://www.codechef.com/Programming-Languages/) | [Contact Us \(http://www.codechef.com/contactus/\)](http://www.codechef.com/contactus/)

© 2009 [Directi Group \(http://directi.com/\)](http://directi.com/). All Rights Reserved. CodeChef uses SPOJ © by [Sphere Research Labs \(http://www.sphere-research.com/\)](http://www.sphere-research.com/)
In order to report copyright violations of any kind, send in an email to [copyright@codechef.com \(mailto:copyright@codechef.com\)](mailto:copyright@codechef.com)

Directi (<http://directi.com>)
Intelligent People. Uncommon Ideas.

The time now is: 10:46:27 PM
Your IP: 117.212.144.114

[CodeChef \(http://www.codechef.com/\)](http://www.codechef.com/) - A Platform for Aspiring Programmers

CodeChef was created as a platform to help programmers make it big in the world of algorithms, **computer programming** and **programming contests**. At CodeChef we work hard to revive the geek in you by hosting a **programming contest** at the start of the month and another smaller programming challenge in the middle of the month. We also aim to have training sessions and discussions related to **algorithms**, **binary search**, technicalities like **array size** and the likes. Apart from providing a platform for **programming competitions**, CodeChef also has various algorithm tutorials and forum discussions to help those who are new to the world of **computer programming**.

Practice Section (<https://www.codechef.com/problems/easy>) - A Place to hone your 'Computer Programming Skills'

Try your hand at one of our many practice problems and submit your solution in a language of your choice. Our **programming contest** judge accepts solutions in over 35+ programming languages. Preparing for coding contests were never this much fun! Receive points, and move up through the CodeChef ranks. Use our practice section to better prepare yourself for the multiple **programming challenges** that take place through-out the month on CodeChef.

Compete (<https://www.codechef.com/problems/easy>) - Monthly Programming Contests and Cook-offs

Here is where you can show off your **computer programming skills**. Take part in our 10 day long monthly coding contest and the shorter format Cook-off **coding contest**. Put yourself up for recognition and win great prizes. Our **programming contests** have prizes worth up to INR 20,000 (for Indian Community), \$700 (for Global Community) and lots more CodeChef goodies up for grabs.

Programming Tools

[Online IDE \(<https://www.codechef.com/ide>\)](https://www.codechef.com/ide)

[Upcoming Coding Contests \(<http://www.codechef.com/contests#FutureContests>\)](http://www.codechef.com/contests#FutureContests)

[Contest Hosting \(<http://www.codechef.com/hostyourcontest>\)](http://www.codechef.com/hostyourcontest)

[Problem Setting \(<http://www.codechef.com/problemsetting>\)](http://www.codechef.com/problemsetting)

[CodeChef Tutorials \(<http://www.codechef.com/wiki/tutorials>\)](http://www.codechef.com/wiki/tutorials)

[CodeChef Wiki \(<https://www.codechef.com/wiki>\)](https://www.codechef.com/wiki)

Practice Problems

[Easy \(<https://www.codechef.com/problems/easy>\)](https://www.codechef.com/problems/easy)

[Medium \(<https://www.codechef.com/problems/medium>\)](https://www.codechef.com/problems/medium)

[Hard \(<https://www.codechef.com/problems/Hard>\)](https://www.codechef.com/problems/Hard)

[Challenge \(<https://www.codechef.com/problems/challenge>\)](https://www.codechef.com/problems/challenge)

[Peer \(<https://www.codechef.com/problems/extcontest>\)](https://www.codechef.com/problems/extcontest)

[School \(<https://www.codechef.com/problems/school>\)](https://www.codechef.com/problems/school)

[FAQ's \(<https://www.codechef.com/wiki/faq>\)](https://www.codechef.com/wiki/faq)

Initiatives

[Go for Gold \(<http://www.codechef.com/goforgold>\)](http://www.codechef.com/goforgold)

[CodeChef for Schools \(<http://www.codechef.com/school>\)](http://www.codechef.com/school)

[Campus Chapters \(\[http://www.codechef.com/campus_chapter/about\]\(http://www.codechef.com/campus_chapter/about\)\)](http://www.codechef.com/campus_chapter/about)

[Domain Registration in India \(<http://www.bigrock.in/>\)](http://www.bigrock.in/) and [Web Hosting \(<http://www.bigrock.com/web-hosting/>\)](http://www.bigrock.com/web-hosting/) powered by BigRock