Editorial plan

The ED publishing house is about to publish an Informatics compendium. It is planned to have *P* pages, denoted as *1, 2,…, P*.

The intention of the authors is to distribute the problems in the compendium in their order (*1, 2, 3, ...*, etc.), based upon the following algorithm: on the first page of the compendium only one problem will be printed (the one with index 1), the 2nd page contains exactly two problems (those with indices 2 and 3 – in this order), then three problems will be printed on the 3rd page (those with indices 4, 5 and 6 – in this order), and so on. Finally, on the *P*th page exactly *P* problems will appear.

Besides this, the publishing house must know beforehand the amplitude of the editorial space – meaning the *minimum* number of pages – so that, given the above presented conditions, the problem with index *n* will is printed as well. In other words, the final published book should contain as many pages as needed so that the problem having the index *n* (the *n*th problem) belongs to the book.

Requirements

Write a program that, given the natural numbers *P* and *n*, where *P* is the number of pages of the book and *n* the maximum index of the problem still belonging to the book, computes the value *C* ($C \in \mathrm{N}$), meaning the total number of *digits* needed for numbering all of the published problems. This value should be computed only if the number of pages *P* is large enough to enable the publishing of the problem with index *n*. Note that there can be blank pages as well – which do not affect the computations.

But if there are not enough pages to include the *n*th problem as well, then the program will compute nothing and will only print `0`.

Input data

The program reads in two natural numbers, *P* şi *n*, separated by space on a single line. Their meanings have already been presented above.

The input line ends in *newline* ('*\n*') (obtained by pressing the *Enter* key).

Output data

Following the requirements above, your program will either print out (using *stdout* stream) the natural number *C* – if we have enough pages, or 0 (zero) if the condition related to the number of pages is not met.

The printed line ends in *newline* ('*\n*').

WARNING! Pay attention to the problem’s requirements: the results must be printed EXACTLY as indicated! In other words, nothing more will be printed onto the standard output stream other than required information; following the automatic evaluation, any supplementary printed character or a kind of printing other than the one required, lead to an erroneous result and therefore to a ‘Fail’ status of your examination (meaning you are rejected).

Restrictions and further explanations

1. The accepted range for the number of pages is: 1 <= *P* <= 200
2. The correct range of the problem index (*n*) is: 1 <= *n* <= 20100
3. **Warning**: depending upon the chosen programming language the file containing your code must use one of the standard extensions: `.c`, `.cpp`, `.java` or `.m`. The web editor **will not automatically add** these extensions. Therefore, if an extension is missing, you will not be able to compile your program!
4. **Warning**: the source filename has to obey the following template: `<nume>.<ext>`, where `nume` is the surname of the student while the extension (`ext`) is chosen according to the

previous warning. Beware of the restrictions imposed by Java programming language related to the class name and the file name!

Examples

Input data	Output data	Explanations
5 15	21	In order to include the problem with index *n*=15, the book must have at least 5 pages. From the input data we already know the number of pages of the book (*P*=5 pages). It follows that there are enough pages in order to publish the 15th problem as well (since *P* is greater or equal to 5) so we can compute the total number of digits, *C*. For this case, on each page the problems appear as follows: • 1 (page 1, only one problem is printed) • 2, 3 (page 2, two problems are printed) • 4, 5, 6 (page 3, three problems) • 7, 8, 9, 10 (page 4, four problems) • 11, 12, 13, 14, 15 (page 5, five problems are printed). The total number of digits used to index *all* the published problems (15 problems in this case) is: *C*=21. The program will print out this value and then will conclude itself.
6 23	0	In this context, in order to include the problem with index *n*=23 the compendium should have at least 7 pages. Since we only have *P*=6 pages available, according to the requirements, the program will not compute anything (since *P*<*7*) and will only print out a zero (0), immediately concluding itself after the print.

Effective working time: 120 minutes