

Plan editorial

La editura ED este pe cale să apară un compendiu de probleme de informatică planificat a avea P pagini, numerotate de la 1 la P .

Autorii intenționează să plaseze problemele din culegere, numerotate cu 1, 2, 3,...,etc, în ordinea apariției lor, respectând următorul algoritm: pe prima pagină a culegerii este scrisă o singură problemă (cea cu numărul 1). Pe a doua pagină sunt scrise exact două probleme (cele cu numerele 2 și 3, în această ordine). Pe cea de-a treia pagină sunt scrise exact trei probleme (cele cu numerele 4, 5 și 6, în această ordine) ș.a.m.d. În sfârșit, pe cea de a P -a pagină vor apărea exact P probleme.

Pe lângă aceasta, editura trebuie să știe din timp amplitudinea spațiului editorial rezervat (a numărului *minim* necesar de pagini) astfel încât, în condițiile prezentate, cartea să cuprindă și problema cu numărul n .

Cerință

Scrieți un program care, pe baza numerelor naturale P și n , semnificând numărul de pagini al cărții și respectiv indexul maxim al problemei care se dorește publicată, determină valoarea C ($C \in \mathbb{N}$), reprezentând numărul total de *cifre* care au fost utilizate în numerotarea tuturor problemelor publicate în carte, doar dacă numărul de pagini P este suficient pentru a cuprinde problema cu index maxim n . Se menționează că pot exista și pagini albe.

Dacă însă nu există pagini suficiente (P) pentru a cuprinde numărul n indicat mai sus, atunci programul nu va calcula nimic și va afișa valoarea 0.

Date de intrare

De la intrare (stream-ul *stdin*) sunt preluate cele două numere naturale P și n , separate printr-un spațiu. Semnificația acestora este prezentată anterior.

Linia citită se încheie cu caracter *newline* ('\n') obținut prin apăsarea tastei *Enter*.

Date de ieșire

Cu semnificația și în condițiile menționate în cerință, programul va afișa la consolă (pe stream-ul *stdout*) fie numărul natural C (dacă numărul de pagini P este suficient) fie 0 (zero) (dacă condiția privind numărul de pagini nu este îndeplinită).

Linia tipărită se va încheia cu caracterul *newline* ('\n').

ATENȚIE la respectarea cerinței problemei: afișarea rezultatelor trebuie făcută EXACT în modul în care a fost indicat! Cu alte cuvinte, pe stream-ul standard de ieșire nu se va afișa nimic în plus față de cerința problemei; ca urmare a evaluării automate, orice caracter suplimentar afișat, sau o afișare diferită de cea indicată, duc la un rezultat eronat și prin urmare la obținerea calificativului „Respins”.

Restricții și precizări

1. Valoarea numărului de pagini, $1 \leq P \leq 200$
2. Numărul problemei pe care trebuie să o cuprindă cartea, $1 \leq n \leq 20100$
3. **Atenție:** În funcție de limbajul de programare ales, fișierul ce conține codul trebuie să aibă una din extensiile .c, .cpp, .java, sau .m. Editorul web **nu va adăuga automat** aceste extensii și lipsa lor duce la imposibilitatea compilării programului!
4. **Atenție:** Fișierul sursă trebuie numit de candidat sub forma: `<nume>.<ext>` unde *nume* este numele de familie al candidatului și *ext* este cea aleasă conform punctului anterior. Atenție la restricțiile impuse de limbajul Java legate de numele clasei și numele fișierului!

Exemple

Date intrare	Date ieșire	Explicații
5 15	21	<p>Pentru a conține și problema cu numărul $n=15$, culegerea trebuie să aibă minimum 5 pagini. Din datele de intrare știm deja numărul de pagini pe care-l are cartea ($P=5$ pagini). Prin urmare avem suficiente pagini pentru a publica și problema cu index 15.</p> <p>Pentru cazul considerat, pe fiecare pagină problemele apar astfel:</p> <ul style="list-style-type: none">• 1 (pagina 1, o singură problemă)• 2, 3 (pagina 2, două probleme)• 4, 5, 6 (pagina 3, trei probleme)• 7, 8, 9, 10 (pagina 4, patru probleme)• 11, 12, 13, 14, 15 (pagina 5, cinci probleme) <p>În această situație, cerința numărului minim de pagini fiind îndeplinită, putem calcula numărul cifrelor folosite pentru generarea indecșilor problemelor publicate. Găsim: $C=21$ de cifre. Valoarea C va fi afișată la ieșire și problema se încheie.</p>
6 23	0	<p>Trebuie verificată condiția privind numărul minim de pagini necesare. Astfel, pentru a include problema cu indexul $n=23$, compendiul ar trebui să aibă cel puțin 7 pagini.</p> <p>Deoarece avem doar $P=6$ pagini disponibile, în conformitate cu cerințele, programul nu va calcula nimic (deoarece $P < 7$) și va tipări doar zero (0), încheindu-se imediat după această tipărire.</p>

Timp efectiv de lucru: 120 de minute