

XXIV НАЦИОНАЛНА ОЛИМПИАДА ПО ИНФОРМАТИКА

Областен кръг, 1 март 2008

Задача С3. ПОДРАВНЯВАНЕ

Първан е силен не само по Информатика, но и по Информационни технологии. Когато изучавали текстообработка в часовете по Информационни технологии му хрумнала следната задача. *Дума* наричаме всяка последователност от букви на латиницата и кирилицата, цифри и препинателни знаци, която не съдържа интервали. *Текст* е съставен от няколко *реда*, всеки от които съдържа последователност от думи, разделени с интервали. Подравняването в зададен брой, например 60, позиции е обичайна операция при работата с текст. Един ред на текста е *двустранно подравнен в 60 позиции*, когато е с дължина 60 (без да се брои знакът за нов ред), започва и завършва със знак, различен от интервал, като недостигащите знаци, за да стане дължината на реда точно 60, се допълват от интервали, *равномерно разпръснати* между думите. Равномерно разпръснати интервали означава, че или всички последователности от интервали са с равна дължина, или, ако това е невъзможно, дължината на част от последователностите от интервали е с 1 по-голяма от дължината на останалите последователности. При това, всички по-дълги последователности от интервали се срещат от началото на реда до определена позиция и след това следват всички по-къси последователности от интервали. Напишете програма **justify**, която по зададен текст подравнява двустранно редовете му в 60 позиции.

Вход

На първия ред на стандартния вход е зададено цяло положително число N – брой на редовете в текста, които трябва да бъдат двустранно подравнени. Следват самите редове – N на брой. Дължината на всеки от редовете не е по-голяма от 60. В текста няма да има дума по-дълга от 29 знака. Между две думи на текста може да има повече от един интервал.

Изход

На стандартния изход програмата трябва да изведе двустранно подравнени в 60 позиции първите $N-1$ реда от входа. След тях трябва да се изведе последният ред, като между думите му трябва да има само по един интервал (т.е. последният ред трябва да е подравнен вляво).

ПРИМЕР

Вход

4

При това, всички по-дълги последователности са преди всички по-къси. Напишете програма `justify`, която по зададен текст го подравнява двустранно в 60 позиции на ред.

Изход

При това, всички по-дълги последователности са преди всички по-къси. Напишете програма `justify`, която по зададен текст го подравнява двустранно в 60 позиции на ред.