

АНАЛИЗ НА РЕШЕНИЕТО НА ЗАДАЧА РАЗКАЗИ

Предложената задача е модификация на класическата задача за раницата, в която се търси само максималния товар, който може да се побере в раницата, но не и предметите, които участват във формирането на този максимален товар.

Тук ще изложим една от идеите, които може да се използват за решаване на задачата.

Да означим с $p[i]$ и $c[i]$ съответно броя на страниците и хонорара за i -тия разказ, а с $best[i][j]$ – максималната печалба, която авторът ще получи, ако включва в книгата си само разкази с номера от 1 до i (не задължително всички) и книгата съдържа най-много j страници ($i = 1, 2, \dots, N; j = 0, 1, 2, \dots, M$). Ясно е, че отговорът на задачата ще бъде стойността на $best[N][M]$. Намирането му може да стане като последователно се пресмятат стойностите на $best[1][j], best[2][j], \dots, best[N][j]$, т.е. първо решаваме задачата, като използваме само първия разказ, след това – като имаме право да използваме първия и втория и т.н. Може да считаме, че $best[i][0] = 0$ за $i = 1, 2, \dots, N$. При определянето на $best[i][j]$ използваме, че при неговото формиране разказът с номер i или участва, или не участва. Ако разказът с номер i не участва при формирането на $best[i][j]$, то $best[i][j] = best[i-1][j]$. Ако разказът с номер i участва при формирането на $best[i][j]$, то $best[i][j] = c[i] + best[i-1][j-p[i]]$.

Тъй като след обработката на информацията, свързана с роман с номер i , стойностите на $p[i]$ и $c[i]$ повече не са ни необходими, вместо масиви $p[i]$ и $c[i]$ е достатъчно да използваме само две променливи p и c .

Решение, което използва само горните съображения може да намерите във файла book-1.cpp).

Може да намалим използваната памет, ако забележим, че при пресмятането на $best[i][j]$ са необходими само стойности на $best[i-1][j]$ и на $best[i-1][j-p[i]]$. Следователно няма нужда от двумерен масив. Достатъчно е да използваме едномерен масив за запомняне на пресметнатите стойности. Тук обаче за всяко i стойностите на елементите на този масив трябва да се пресмятат отзад напред, т.е. за $j = M, M-1, \dots, 2, 1$. Такова решение може да намерите във файла book.cpp.

Автор: Младен Манев