

farma – descrierea soluției

prof. Emanuela Cerchez
Colegiul Național "Emil Racoviță" Iași

Rezolvarea cerinței 1 nu necesită comentarii. Pentru cerința 2 putem proceda în modul următor.

Să numerotăm substanțele active de la 1 la n (denumirile acestora sunt irelevante în problemă). Să notăm cu $lg[i]$ =numărul de medicamente existente pentru substanța activă i . Problema constă în generarea elementelor produsului cartezian

$$\{1, 2, \dots, lg[1]\} \times \{1, 2, \dots, lg[2]\} \times \dots \times \{1, 2, \dots, lg[n]\}$$

Nu toate elementele produsului cartezian ne interesează, ci doar acelea pentru care medicamentele corespunzătoare sunt compatibile două câte două. Iar din toate soluțiile posibile, ne interesează cea pentru care suma elementelor este minimă.

Problema poate fi abordată în moduri diferite:

1. Generăm toate elementele produsului cartezian printr-un algoritm de tip succesiv. Pentru fiecare element generat, verificăm dacă medicamentele sunt compatibile două câte două. În caz afirmativ, comparăm suma necesară pentru soluția curentă cu suma minimă și o reținem dacă este cazul.

Această abordare obține 65 de puncte (sursa `farma_gen`)

2. Pentru a optimiza generarea observăm că pe poziția i în soluție nu are rost să selectăm un medicament care este incompatibil cu cel puțin unul deja selectat pe pozițiile $1, 2, \dots, i-1$. Putem genera soluțiile ținând cont de această observație utilizând o stivă, implementată ca un vector `sol` cu n elemente.

La pasul curent i , trebuie să selectăm un element care poate fi pus în soluție pe poziția i .

Dacă $i > n$, soluția este completă, comparăm suma necesară pentru cumpărarea ei cu suma minimă și o reținem dacă este cazul.

Dacă $i \leq n$, atunci incrementăm `sol[i]` (plasăm un nou element pe poziția i în soluție).

Apar următoarele cazuri posibile:

`sol[i] > lg[i]` (în acest caz am epuizat valorile posibile pentru poziția i , cobor pe stivă revenind la poziția precedentă ($i--$)).

`sol[i] <= lg[i]`, iar valoarea `sol[i]` curentă este compatibilă cu toate valorile `sol[1], sol[2], ..., sol[i-1]`, atunci urcăm un nivel pe stivă și ($i++$) și inițializăm `sol[i]` cu 0.

`sol[i] <= lg[i]`, dar valoarea `sol[i]` curentă nu este compatibilă cu valorile `sol[1], sol[2], ..., sol[i-1]`; în acest caz rămân pe stivă la același nivel i , urmând la pasul următor să incrementăm valoarea curentă.

Această generare este implementată în `farma_ok_slow` și obține 75 de puncte

3. Putem optimiza generarea în continuare făcând următoarea observație. Dacă suma necesară cumpărării medicamentelor în soluția parțial generată depășește deja suma minimă curentă, abandonăm generarea cu valoarea curentă și revenim la pasul precedent, pentru că e clar că această soluție nu va fi convenabilă.

Pentru a obține mai rapid o sumă mai mică, vom sorta crescător după preț medicamentele din lista fiecărei substanțe active.

Această soluție obține 100 de puncte (`farma_ok`).

O implementare recursivă pentru această idee este concisă și eficientă (`farma_rec`) și obține de asemenea 100 de puncte, dar depășește nivelul clasei a VIII-a.

Sunt posibile și abordări Greedy care obțin punctaje parțiale. De exemplu dacă nu ținem cont de incompatibilități și pentru fiecare substanță activă selectăm medicamentul cel mai ieftin obținem 10 puncte pentru cerința 2.

Dacă abordăm Greedy ținând cont de incompatibilități (adică mă opresc la prima soluție care îndeplinește condiția de compatibilitate, evident, după sortarea medicamentelor crescător după preț) se obțin 20 de puncte la cerința 2.

În toate soluțiile prezentate este necesară verificarea rapidă a compatibilității medicamentelor selectate în soluție. Pentru această propunem următoarea codificare a medicamentelor. Substanțele active sunt numerotate de la 1 la n ($n < 10$), iar medicamentele din lista substanței active i de la 1 la $lg[i]$ ($lg[i] < 10$). Identificăm în mod unic un medicament printr-un număr de două cifre format din numărul substanței active și numărul de ordine al medicamentului în lista de medicamente a substanței active.

Pentru a reține relațiile de incompatibilitate între medicamente utilizăm o matrice $d_{100 \times 100}$.

$d[c1][c2]=d[c2][c1]=1$, dacă medicamentele care au numerele de identificare $c1$ și $c2$ sunt incompatibile, respectiv 0 în caz contrar.

Atenție! Codificarea medicamentelor se va face după sortarea acestora după preț, pentru că la sortare medicamentele își schimbă pozițiile în lista de medicamente.