### Programmēšanas uzdevumu frizieri

**Atmiņas ierobežojums**: 4 MIB **laika ierobežojums**: 0,2 sekunžu **ievades fails**: hair.in **izvade**: hair.out

#### Apraksts

Lielā pilsētā (vairāk nekā miljons iedzīvotāju, bet ne vairāk kā miljards) ir viena un tikai viena frizētava, kurā ir tikai daži frizieri (to skaits ir līdz ). Katram frizierim ir unikāls numurs $[1. 9] $, nodrošinot efektīvāku servisu. Studija mēra laiku noteiktās laika vienībās $[1\,..\, 2\, 000\, 000\, 000 $,] un laika skaitīšana sākas studijas atvēršanas brīdī.

Lai gan klientu skaits ir milzīgs un pieprasījums pēc frizieriem ir ļoti liels, katram frizierim būtu jāveic obligāti pārtraukumi. Obligātā pārtraukuma laiks katram frizierim ir tad, kad simtiem ciparu laika numurā sakrīt ar friziera numuru. Piemēram, frizierim ar numuru ir jāietur pārtraukums laika intervālos Ls [500. 599] $, $[1500.. 1599] $, $[2500.. 2599] $utt. Pārtraukuma laikā frizierim ir aizliegts apkalpot klientu. Turklāt klienta tikšanos nevar sadalīt pa posmiem, t. i., klientu var apkalpot tikai viens frizieris bez pārtraukumiem. Līdz ar to frizieris nevar sākt apkalpot klientu, ja pakalpojumu nevar pabeigt pirms pārtraukuma sākuma.

Klients nekavējoties jāapkalpo, ja ir neaizņemts frizieris un viņam/viņai nav nekādu ierobežojumu attiecībā uz šo darbu. Pabeidzot darbu ar pašreizējo klientu, frizierim nekavējoties jāmēģina sākt nākamā klienta apkalpošanu. Precīzāk: klienta parādās brīdī, kad , un viņa iecelšanai ir nepieciešams laiks (kalpošanas ilgums) . Friziera pašlaik ir bez maksas. Tādējādi šī tikšanās notiks laika intervālā $[T\_1..T\_1 + D\_1-1] $. Tikšanās ir pabeigta brīdī, kad ir $T\_1 + D\_1-1 $. Ja klienta jau ir parādījies pirms vai tieši laika periodā $T\_1 + D\_1 $, friziera var sākt strādāt ar klienta laika periodā $T\_1 + D\_1 $.