

Српска информатичка олимпијада, Београд – 26.6.2021.

1.

Књига је прозор у свет

Свака књига која се продаје у књижари описана је шифром и ценом. Ада има одређени износ динара и жели да купи што скупље књиге. Ада куповину обавља тако што купује најскупље књиге, док има новца. Ако нема новца за најскупљу књигу, узима најскупљу књигу за коју има новца. Написати програм који ће приказати шифре књига које је купила Ада и преостали износ новца (ако јој је преостало новца).

Улаз

У првој линији стандардног улаза налази се износ новца (реалан број са две децимале не већи од 20000) који има Ада, у другој број књига, N ($N < 1000$), а затим се, у сваке две линије стандардног улаза, уносе, редом, шифра (ниска карактера до 20 карактера) и цена (реалан број са две децимале не већи од 20000) књиге, свака у посебном реду, за свих N књига.

Излаз

У свакој линији стандардног излаза исписују се шифре и цене купљених књига (раздвојене размаком), ако их има. У последњој линији приказује се преостали износ новца, ако постоји. Испис бројева вршити са прецизношћу од две децимале.

Пример1

Улаз

1250.75

5

sifra1

1010.30

sifra2

357.35

sifra3

725.45

sifra4

1125.50

sifra5

115.75

Излаз

sifra4 1125.50

sifra5 115.75

9.50

Пример2

Улаз

10000

6

sifra1

3010

sifra2

3005

sifra3

5725

sifra4

1265

sifra5

2075

sifra6

385

Изназ

sifra3 5725.00

sifra1 3010.00

sifra4 1265.00

Решење:

Овај задатак је по алгоритамском моделу сличан задатку Највреднији предмети из Методичка збирка задатака из основа програмирања (Python,)

https://petlja.org/biblioteka/r/Zbirka-python/najvredniji_predmeti

Књиге можемо сортирати по цени у нерастућем поретку. С обзиром да библиотечке функције подразумевано сортирају низ уређених парова или низ структура растуће, можемо формирати низ структура или низ уређених парова тако да учитану цену књиге помножимо са -1 и памтимо њену негативну вредност. Друга могућност је да осмислимо функцију поређења две структуре.

Након сортирања одређујемо књигу која ће се купити, обилазећи редом књиге од најскупље до најјефтиније. Ако је текућа књига скупља од износа којим располаже Ада, игнорише се, а иначе Ада купује ту књигу, те на стандарни излаз исписујемо шифру купљене књиге и цену. У том случају, смањује се износ којим располаже Ада. Поступак понављамо док Ада не потроши цео износ или док не обрадимо све књиге

из низа. Ако је Ади преостало новца, на крају исписујемо и износ којим располаже Ада.

2.

Удружење тенисера

Нова структура АТП (асоцијације тениских професионалаца) у којој ради програмер Ноле има облик стабла са кореном, тако да сваком чвору дрвета одговара један тенисер. Неки тенисер се сматра функционером других тенисера ако је он корен подстабла које садржи друге чворове (наследнике) којима је он функционер . У новој структури АТП удружења, сваки тенисер је понудио списак тенисера који он жели да види на месту свог функционера.

Зараде у АТПу се утврђују према следећим захтевима:

- Плата је позитиван цео број, вишеструко већа од званичне минималне зараде и не мања од званичне минималне зараде;
- Плата сваког функционера мора бити већа од збира зарада његових директних подређених тенисера.

Напишите програм који прави нову структуру АТПа тако да се сваком тенисеру додели функционер са списка који је он предложио, а износ плата свих тенисера буде што мањи.

Улаз

Први ред стандардног уноса садржи две позитивне целобројне вредности N и K , одвојене размаком ($2 \leq N \leq 5000$, $1 \leq K \leq 550$).

Број N представља број тенисера, а број K представља минималну зараду.

Тенисери се нумеришу природним бројевима: 1, 2, 3, ..., N . Потом на стандардном улазу следи N редова који описују списак жеља тенисера.

Ред i садржи цео број a_i , праћен списком a_i целих бројева. Списак садржи бројеве тенисера које би i -ти тенисер прихватио као своје функционере

$$\left(\sum_{i=1}^N a_i \leq 10000 \right).$$

Излаз

У једином реду стандардног излаза приказати најнижи могући укупан износ плата, међу свим важећим опцијама. Гарантовано је да су улазни подаци изабрани тако да постоји бар једно решење.

Пример:

Улаз

4 100

1 4

3 1 3 4

2 1 2

1 3

Излаз

800

Појашњење

АТП компанија има 4 тенисера. Минимална зарада је 100.

Тенисер №1 жели за функционера тенисера №4.

Тенисер №2 жели за функционера №1 или №3 или №4.

**Тенисер №3 жели за функционера №1 или №2.
Тенисер №4 жели за функционера №3.**

Решење:

Анализирајмо пример дат у формулацији задатка:

**АТП компанија има 4 тенисера. Минимална зарада је 100.
Тенисер №1 жели за функционера тенисера №4.
Тенисер №2 жели за функционера №1 или №3 или №4.
Тенисер №3 жели за функционера №1 или №2.
Тенисер №4 жели за функционера №3.**

Матрица суседства за релацију „пожељни функционер“ изгледа овако:

	1	2	3	4
1	0	0	0	1
2	1	0	1	1
3	1	1	0	0
4	0	0	1	0

То значи да транспонована матрица суседства показује ко коме може бити шеф (на пример, тенисер 3 може бити функционер тенисерима 2 и 4):

	1	2	3	4
1	0	1	1	0
2	0	0	1	0
3	0	1	0	1
4	1	1	0	0

У даљем тексту са BFS означавамо алгоритам обиласка датог графа (матрице) по ширини.

Ако изберемо чвор 1 за корен стабла и применимо BFS, добијамо:

ниво 1: 1
ниво 2: 2, 3
ниво 3: 4

Ако изберемо чвор 2 за корен стабла и применимо BFS, добијамо:

ниво 1: 2
ниво 2: 2, 3
ниво 3: 4
ниво 4: 1

Ако изберемо чвор 3 за корен стабла и применимо BFS, добијамо:

ниво 1: 3
ниво 2: 2, 4
ниво 3: 1

Ако изберемо чвор 4 за корен стабла и применимо BFS, добијамо:

ниво 1: 4
ниво 2: 1, 2
ниво 3: 3

За сваку од опција израчунавамо укупну плату, посебно памтећи минималну вредност.

Дакле, потребно је имплементирати следећи алгоритам:

- 1 Из улазних података креирамо транспоновану матрицу суседства.
- 2 Понављајмо у петљи по чворовима (од 1 до N):
Покренимо BFS почевши од избраног чвора
Израчунавамо вредност износа за укупну зараду.
Упоређујемо ту вредност са досадашњим минимумом и задржавамо мању вредност.

3.

Број појава речи

Стари Римљани су оставили много писаних трагова по Србији. На локалитету С је пронађен писани извор у облику матрице састављене од малих латиничних слова. Димензије писаног извора су $N \times M$. Млади археолог Драгослав је уочио реч која је такође написана као матрица малих латиничних слова и то димензије $R \times C$. Напишите програм који проналази број појава те речи у писаном извору (тј. друге матрице у првој матрици), ако знате да је Драгослав уочио да места на којима се јавља друга матрица се могу преклапати.

Улаз

Из првог реда стандардног улаза прочитајте позитивне целе бројеве R и C - димензије матрице која је једнака прочитаној речи ($1 \leq R, C \leq 100$). Из следећих R редова се читавају низови малих латиничних слова дужине C . Из $(R + 2)$. реда се читавају цели позитивни бројеви N и M - димензије матрице, која образује писани извор. Из последњих N редова се читавају низови малих латиничних слова дужине M ($1 \leq N, M \leq 5000$).

Излаз

На стандардни излаз испишите један број - пронађени број појава дате дводимензионалне речи у писаном извору.

Пример

Улаз

2 2

dd

dr

3 3

ddd

ddr

drd

Излаз

2

Појашњење

Појава речи почиње на позицијама (1,2) и (2,1):

ddd

ddr

drd

и

ddd
ddr
drd

Решење:

Алгоритам грубе силе има временску сложеност $O(NMR)$ и није примењив за дата ограничења. За сваку позицију у писаном извору проверавамо да ли се јавља дводимензионална реч. Најједноставнији начин да проверите да ли се реч појављује на датој позицији је да се помоћу два угнежђена циклуса тестира подудараре одговарајућих слова. Потребна су укупно четири угнежђена циклуса или сложеност $O(NMR)$.

Али, оптимизација овог алгоритма циља да убрза процедуру провере да ли се реч јавља на датој позицији. Мудро је да се за ову врсту провере користи релативно стандардна техника – хеширање стрингова. Прво проналазимо хеш за сваки ред речи, а затим проналазимо хеш сваког подстринга са Ц знакова у писаном извору. Стога је приликом провере подудараре речи довољно обићи само врсте и видети да ли је одговарајући хеш у писаном извору исти као хеш одговарајућег реда у дводимензионалној речи. Слично томе, могли би да погледамо хешеве у колонама. Овде је сложеност или $O(NMR)$ или $O(NMC)$.

Још ефикасније решење захтева више труда у имплементацији с обзиром да не постоји библиотечка подршка у многим програмским језицима. Ако за сваку подстринг са Ц узастопних знакова можемо рећи који ред речи одговара или не одговара ниједном реду писаног извора, накнадна провера ће бити лака - на колонама ћемо спровести КМП алгоритам провере упаривања стрингова. Важно је да спроведемо поступак провере ако имамо линије које су исте у дводимензионалној речи. Овај задатак је еквивалентан следећем - имамо Р речи и морамо да кажемо где и које речи се јављају за унети текст (ред из писаног извора). Тако долазимо до идеје да користимо Ахо-Корасик алгоритам за Р-редова дводимензионалне речи.