

## Gépkölcsönzés

Nemes Tihamér OITV 2010. 3. forduló 2. korcsoport 4. feladat

Mekk Elek vállalkozásának van egy nagy értékű munkagépe, amit más vállalkozók kölcsönözhetnek tőle. Egy vagy több, de összefüggő napokra lehet igényelni, minden napra azonos a bérleti díj. A következő  $M$  napra sok megrendelés érkezett. Minden megrendelés két számot tartalmaz,  $D$ : ahány napra bérelni akarja a megrendelő,  $H$ : az a határidő, ameddig a megrendelőnek el kell végeznie a munkát a géppel. Tehát neki olyan  $E$  naptól lehet adni a gépet, amelyre teljesül, hogy  $E+D-1 \leq H$ . Mekk Eleknek az a célja, hogy olyan megrendeléseket teljesítsen, amelyek összességében a lehető legtöbb napra kölcsönzik a gépet.

Készíts programot *gep* néven, amely a megrendelések alapján kiszámítja, hogy Mekk Elek legjobb esetben hány napra tudja bérbe adni a gépet! Továbbá, meg is ad egy megfelelő beosztást.

A *gep.be* szöveges állomány első sorában két egész szám van (egy szóközzel elválasztva), a munkanapok  $M$  ( $1 \leq M \leq 100$ ) száma és a megrendelések  $N$  ( $1 \leq N \leq 1000$ ) száma. A következő  $N$  sor mindegyike két egész számot tartalmaz (egy szóközzel elválasztva),  $D$ : az igényelt napok száma,  $H$ : a határidő. Az igényeket az  $1, \dots, N$  sorszámukkal azonosítjuk, az állomány  $I+1$ -edik sorában van az  $I$ -edik igény. A bemenet az igények határideje szerint nem-csökkenően rendezett.

A *gep.ki* szöveges állomány első sorába azt a legnagyobb  $K$  számot kell írni, amelyre teljesül, hogy összesen  $K$  napra bérbe lehet adni a gépet! A második sorban a kielégített igények  $L$  száma legyen! A következő  $L$  sor egy-egy megrendelés teljesítését tartalmazza, soronként két egész számot. Az első szám a megrendelés sorszáma, a második pedig az első nap sorszáma legyen, amelytől a megrendelő használhatja a gépet! A teljesített igények tetszőleges sorrendben megadhatók. Több megoldás esetén bármelyik megadható.

Példa:

<i>gep.be</i>	<i>gep.ki</i>
10 6	9
2 2	4
1 3	6 8
2 3	5 5
3 6	4 2
3 7	2 1
2 9	

### Megoldás

Vegyük egységnyiinek a napi bérleti díjat! A bevétel ekkor megegyezik a kölcsönzési napok számával.

A feladatot részfeladatokra bontjuk. Tekintsük egy részfeladatnak az  $i$ -edik napig teljesíthető kölcsönzésekből származó bevétel optimalizálását, az első  $j$  megrendelés alapján. Mivel a feladatra teljesül az optimális részproblémák tulajdonság (ellenőrizzük!), a dinamikus programozás stratégiáját alkalmazzuk.

Legyen  $Napok(j)$  a  $j$ -edik megrendelés időtartama,  $Határidő(j)$  pedig a határideje! Jelöljük  $Opt(i, j)$ -vel az  $i$ -edik napig teljesíthető megrendelésekből származó maximális bevételt, de csak az első  $j$  megrendelés alapján! A továbbiakban kihasználjuk, hogy a megrendelések határidő (és sorszám) szerint növekvő sorrendben szerepelnek a bemeneten.

- $Opt(i, j)$  értékét rekurzív módon határozzuk meg. A következő lehetőségek közül választhatunk.
- Az  $i$ -edik napra nem adjuk kölcsön a gépet (vagy nem is tudjuk kölcsönadni). Ekkor a bevétel:  $Opt(i-1, j)$ .
  - A  $j$ -edik megrendelést nem fogadjuk el (vagy nem is tudjuk elfogadni). Ekkor a bevétel:  $Opt(i, j-1)$ .
  - Elfogadjuk a  $j$ -edik megrendelést. Ekkor a  $j$ -edik megrendelőnek az  $i$ -edik napig adhatjuk kölcsön a gépet, azaz ő az  $i - Napok(j) + 1$ -edik napon viheti el. Ezért az előző,  $j-1$  megrendelés legfeljebb az  $i - Napok(j)$ -edik napig foglalhatja le a gépet. Így a bevétel:  $Opt(i - Napok(j), j-1) + Napok(j)$ .

Ez utóbbi esetre csak akkor kerülhet sor, ha  $i - Napok(j)$  értéke pozitív, illetve nem lépjük túl a  $j$ -edik megrendelés határidejét. Azaz:  $i < Határidő(j)$ .

A három lehetőség közül nyilván azt választjuk, amelyik a legnagyobb bevétellel jár:

$$Opt(i, j) = \text{MAX} \left\{ \begin{array}{l} Opt(i-1, j), \\ Opt(i, j-1), \\ Opt(i - Napok(j), j-1) + Napok(j) \end{array} \right. \quad \text{ahol} \quad \left\{ \begin{array}{l} i \leq Határidő(j) \text{ és} \\ i - Napok(j) > 0 \end{array} \right\}$$

A Gépkölcsönzés Excel-munkalap alapján ellenőrizzük néhány érték meghatározását!

	A	B	C	D	E	F	G	H	I
4		J:	0	1	2	3	4	5	6
5		Határidő(J):	0	2	3	3	6	7	9
6		Napok(J):	0	2	1	2	3	3	2
7	I:	0	0	0	0	0	0	0	0
8		1	0	0	1	1	1	1	1
9		2	0	2	2	2	2	2	2
10		3	0	2	3	3	3	3	3
11		4	0	2	3	3	4	4	4
12		5	0	2	3	3	5	5	5
13		6	0	2	3	3	6	6	6
14		7	0	2	3	3	6	7	7
15		8	0	2	3	3	6	7	8
16		9	0	2	3	3	6	7	9
17	10	0	2	3	3	6	7	9	

Az  $Opt(i, j)$  értékek meghatározása

A rekurzióhoz a következő, nyilvánvaló báziskritériumokat használjuk.

Ha nem teljesítünk egyetlen megrendelést sem, akkor nem lesz bevételünk:

$$Opt(i, 0) = 0 \quad 0 \leq i \leq M$$

A nulladik napra a bevétel 0, akárhány megrendelésünk is van:

$$Opt(0, j) = 0 \quad 0 \leq j \leq N$$

A fenti összefüggések alapján már könnyen felírhatjuk az algoritmust.

A rendeléseket listában, egy megrendelés adatait pedig struktúrában tároljuk:

STRUKTÚRA TMegrendelés

VÁLTOZÓ Sorszám, Napok, Határidő MINT Egész

STRUKTÚRA VÉGE

VÁLTOZÓ Megrendelések MINT Lista(Elemtípus: TMegrendelés)

VÁLTOZÓ N, M MINT Egész

Be: N, M, Megrendelések

Az  $Opt$  tömb elemeinek meghatározása:

VÁLTOZÓ  $Opt(0...M, 0...N)$  MINT Egész

CIKLUS  $I=0$ -tól  $M$ -ig

$Opt(I, 0) = 0$

CIKLUS VÉGE

CIKLUS  $J=0$ -tól  $N$ -ig

$Opt(0, J) = 0$

CIKLUS VÉGE

CIKLUS  $I=1$ -től  $M$ -ig

CIKLUS  $J=1$ -től  $N$ -ig

MINŐSÍT Megrendelések( $J$ )-vel

$Opt(I, J) = \text{Max}(Opt(I-1, J), Opt(I, J-1))$

HA  $I \geq \text{.Napok}$  ÉS  $I \leq \text{.Határidő}$  AKKOR

$Opt(I, J) = \text{Max}(Opt(I, J), Opt(I-\text{.Napok}, J-1) + \text{.Napok})$

ELÁGAZÁS VÉGE

MINŐSÍTÉS VÉGE

CIKLUS VÉGE

CIKLUS VÉGE

A maximális bevételt az  $Opt(M, N)$  tömbelem szolgáltatja.

Meg kell még határozni a teljesített megrendelések számát és listáját.

Ehhez visszafelé követjük a megrendeléseket. Az  $Opt(M, N)$ -ből kiindulva mindig meghatározzuk azt az előző  $Opt(I, J)$  értéket, melyből megkaptuk az újabb  $Opt(I, J)$ -t (lásd az alábbi ábrát és a Gépkiölsönzés Excel-munkalapot). Közben azok a  $J$  indexek jelzik a teljesített megrendelések sorszámát, melyekre nőtt az  $Opt(I, J)$  értéke.

	A	B	C	D	E	F	G	H	I
20		J:	0	1	2	3	4	5	6
21		Határidő(J):	0	2	3	3	6	7	9
22		Napok(J):	0	2	1	2	3	3	2
23	I:	0	0	0	0	0	0	0	0
24		1	0	0	1	1	1	1	1
25		2	0	2	2	2	2	2	2
26		3	0	2	3	3	3	3	3
27		4	0	2	3	3	4	4	4
28		5	0	2	3	3	5	5	5
29		6	0	2	3	3	6	6	6
30		7	0	2	3	3	6	7	7
31		8	0	2	3	3	6	7	8
32		9	0	2	3	3	6	7	9
33	10	0	2	3	3	6	7	9	

A teljesített megrendelések meghatározása

Az algoritmus:

VÁLTOZÓ Megkap MINT Szótár(Kulcstípus: Egész, Értéktípus: Egész)

VÁLTOZÓ I, J MINT Egész

I = M

J = N

CIKLUS AMÍG I > 0 ÉS J > 0

HA Opt(I, J) == Opt(I-1, J) AKKOR I = I - 1

EGYÉBKÉNT HA Opt(I, J) == Opt(I, J-1) AKKOR J = J - 1

EGYÉBKÉNT

Megkap.Add(Megrendelések(J).Sorszám, I-Megrendelések(J).Napok + 1)

I = I - Megrendelések(J).Napok

J = J - 1

ELÁGAZÁS VÉGE

CIKLUS VÉGE

Ki: Opt(M, N), Megkap.Elemszám, Megkap szótár elemei (kulcs-érték párok)

A teljes megoldást a *gep* program tartalmazza.

A teljesített megrendelések listáját megkaphattuk volna úgy is, hogy az *Opt* tömb elemeinek meghatározása közben egy újabb táblázatban feljegyezzük, honnan léptünk az (I, J) elemre.

Javasoljuk az Olvasónak, hogy hasonlítsa össze a megoldást a 2004-es OITV 3. fordulójának 5. feladatával (Ütemezés).

## További ütemezési feladatok

### 1. forduló feladatok

OITV 2006-1f2	Fazekas
OITV 2008-1f2	Ütemezés
OITV 2012-1f2	Munka
OKTV 2008-1f3	Ütemezés
OKTV 2009-1f3	Pakolás
OKTV 2013-1f3	Munkavállalás

### Feladatok a mohó algoritmusra

OKTV 2002-2f3	Ütemezés
OKTV 2007-3f3	Málna
OKTV 2010-2f3	Ütemezés
OKTV 2014-2f3	Gépek

### Feladatok a dinamikus programozásra

OITV 2013-3f2	Gépek
OKTV 2001-3f3	Kemence
OKTV 2011-3f3	Kemence
OKTV 2014-2f3	Fazekas

### Feladatok a gráfok alkalmazására

OKTV 1998-3f3	Terv
---------------	------