

Egy futárszolgálat a futárjainak a megtett utak alapján ad fizetést. Az egyik futár egy héten át feljegyezte fuvarjai legfontosabb adatait, és azokat eltárolta egy állományban

A fájlban minden sor egy-egy út adatait tartalmazza egymástól szóközzel elválasztva.

- nap sorszáma, ami 1 és 7 közötti érték lehet
- a napon belüli fuvarszám,
- az adott fuvar során megtett utat jelenti kilométerben, egészre kerekítve

Például:

```
1 7 14
3 3 10
1 1 3
1 2 3
```

A 3. sor például azt mutatja, hogy a hét első napján az első fuvar 3 kilométeres távolságot jelentett.

A felhasználó által megadott adatok helyességét, érvényességét nem kell ellenőriznie, feltételezheti, hogy a rendelkezésre álló adatok a leírtaknak megfelelnek.

1. Olvassa be az `adatok.txt` állományban talált adatokat.

tétel: másolás

A szokásos módon egy sor kiolvasása és a konstruktor hívása.
A szeletelés a `class` feladata!

2. Kérjen be a felhasználótól egy tetszőleges napot és fuvarszámot.

Írja ki a képernyőre, ennek a fuvarnak az adatait!

tétel: lineáris keresés

be: (nap)

be: (fuvarszám)

előre a listán, amíg ($i < N$ és $!(\text{getNap}() = \text{nap}$ és $\text{getFuvar}() = \text{fuvarszám})$)

ha $i < N$ akkor ki: (nap, fuvarszám, `getKm()`)

különben ki: („nincs ilyen adat“)

3. Tudjuk, hogy a futár minden héten tart legalább egy szabadnapot. Írja ki a képernyőre, hogy a hét hányadik napjain nem dolgozott a futár!

tétel: kiválogatás tömbbe

új napok[8]

1.végig a listán, `napok[getNap()]++`

2.végig a napok[]-n, ha `napok[i]=0`, akkor ki(*i*)

4. Írja ki a képernyőre, hogy a hét melyik napján volt a legtöbb fuvar! Amennyiben több nap is azonos, maximális számú fuvar volt, elegendő ezek egyikét kiírnia.

tétel: kiválogatás tömbbe,
maximum kiválasztás

új napok[8]

1.végig a listán, `napok[getNap()]++`

2.`maximum(napok[])`, ki(`maximum, i`)

5. Számítsa ki és írja a `km.txt` állományba a mintának megfelelően, hogy az egyes napokon hány kilométert kellett tekerni!

tétel: összegzés – tömbbe –

új napok[8]

1.végig a listán, `napok[getNap()] = napok[getNap()] + getKm()`

2.végig a napok[]-n → ki: fájlba(`napok[i]`)

6. A futár az egyes fuvarokra az út hosszától függően sávosan kap fizetést az alábbi szerint:

500 Ft	700 Ft	900 Ft	1400 Ft	2000 Ft
1 – 2 km	3 – 5 km	6 – 10 km	11 – 20 km	21 – 30 km

Kérjen be a felhasználótól egy tetszőleges napot és fuvarszámot, és határozza meg, hogy mekkora díjazás jár érte!

tétel: lineáris keresés

be: (nap)

be: (fuvarszám)

előre a listán, amíg ($i < N$ és $!(\text{getNap}() = \text{nap}$ és $\text{getFuvar}() = \text{fuvarszám})$)

ha $i < N$ akkor ki: (nap, fuvarszám, **getFt()**)

különben ki: („nincs ilyen adat”)

7. Határozza meg, és írja ki a képernyőre, hogy a futár mekkora összeget kap a heti munkájáért!

tétel: összegzés

új napok[8]

1.végig a listán, $\text{napok}[\text{getNap}()] = \text{napok}[\text{getNap}()] + \text{getFt}()$

2.végig a napok[]-n \rightarrow ki: (napok[i])

8. Mekkora összeget kapott a pénteki utolsó fuvarért?

tétel: megszámlálás, kiválasztás

1.végig a listán, ha $\text{getNap}() = 5$, akkor $\text{db}++$

2.végig a listán amíg $!(\text{getNap}() = 5$ és $\text{getFuvar}() = \text{db}) \rightarrow$ ki: (**getFt()**)