

4. FÉLÉVI BEADANDÓ

PROGRAMOZÁSMÓDSZERTANBÓL

2005/06 TANÉVBEN

(REKURZÍV PROGRAMOK)

Általános tudnivalók

1. A megoldást rekurzívan kell elkészíteni, ami azt jelenti, hogy az eljárás(ok) algoritmus (és kódja) nem tartalmazhat ciklust.
2. A hatékonyság most most nem fontos.
3. Az adatokat a használó választása szerint vagy a klaviatúráról vagy egy egyszerű text fájlból kell beolvasni, és a képernyőre –és igény szerint– egy szintén egyszerű text fájlba kell kiírni. (A fájloknak egy sora egy kutya adatait, output esetében esetleg magyarázó adatot is tartalmaz. A pontos fájlleírásokat a dokumentációnak tartalmaznia kell! Igényes és teljes Felhasználói, ill. Fejlesztői dokumentációt kell mellékelni!)
4. A kutyáknak nevük is van!
5. Az olyan adatokat sem hagyhat figyelmen kívül, azaz –legalább– be kell olvasni, amelyek –esetleg– nem játszanak szerepet a feladat megoldásában.
6. **A beadandókkal szembeni elvárásokat és értékelését okvetlenül [olvassa el!](#)**

Feladatok

1. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely megadja az automatikusan kieső kutyák pontszámait, illetve jelzi, ha nincs ilyen kutya!
2. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely eldönti, van-e olyan kutya, amelyik minden kategóriában győztes!
3. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely eldönti, van-e olyan kategória, ahol holtverseny alakult ki a győztesek között!

4. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készíts Pascal programot, amely eldönti, hogy van-e olyan kutya, amely valamilyen szempont szerint tökéletes!
5. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely eldönti, van-e olyan kutya, amely minden kategóriában kiesett!
6. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely megadja a kategóriagyőzteseket (amelyik többször is nyert, azt is csak egyszer adja meg)!
7. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely megadja azokat a kutyákat, amelyek több kategóriában is győztek!
8. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely eldönti, van-e olyan kutya, amely abszolút győztes lett, de minden kategóriában van nála jobb!
9. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely megadja azokat a kutyákat, amelyek egyik kategóriában sem voltak sem győztesek, sem utolsók!
10. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely megadja azokat a kutyákat, amelyek valamilyen szempont szerint győztesek voltak, más szempont szerint pedig kiestek!

11. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely megadja azokat a kutyákat, amelyek egy szempont szerint győztesek voltak, de más szempont szerint nem nyertek!
12. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely megadja azokat a kutyákat, amelyek ha nyertek valamilyen kategóriában, akkor azt holtversenyben nyerték!
13. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely megadja azokat a kutyákat, amelyek egy kategóriában sem nyertek, de mindenhol elérték a minimális pontszámot!
14. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely eldönti, hogy van-e olyan kutya, amely több kategóriában is győzött!
15. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely eldönti, hogy létezik-e olyan A és B kutya, hogy A minden kategóriában több pontot kapott, mint B !
16. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely megadja azokat a kategóriákat, ahol nem volt kieső kutya!
17. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely megadja azokat a kategóriákat, ahol legalább 2 kutya ért el maximális pontszámot!

18. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely megadja azokat a kategóriákat, ahol legalább két kutya maradt a kiesések után!
19. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely megadja azokat a kategóriákat, amelyekben holtversenyt kellett hirdetni!
20. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely megadja azokat a kategóriákat, amelyekben volt tökéletes kutya!
21. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely megadja a valamilyen szempont szerint tökéletes kutyákat (amelyik több kategóriában is tökéletes volt, az is csak egyszer szerepeljen)!
22. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely megadja a valamilyen szempontból kiesett kutyákat (amelyik többször is kiesett, az is csak egyszer szerepeljen)!
23. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely eldönti, hogy volt-e valamelyik kategóriában holtverseny!
24. Egy N résztvevőjű kutyaszépségversenyen M különböző szempont szerint pontoznak minden kutyát, s az eredményt a $KUTYA(N,M)$ mátrix tartalmazza. Minden szempont alapján maximum $MAX(I)$ pont adható (nem negatív egész szám). Az összetett versenyből automatikusan kiesik az a kutya, amely valamelyik kategóriában nem éri el a kategóriánként megadott alsó ponthatárt ($MIN(I)$). Készítsen programot, amely eldönti, hogy létezik-e olyan A és B kutya, hogy A minden kategóriában kevesebb pontot kapott, mint B !