Országos Programozó Verseny – Neumann János Egyetem GAMF Kar – 2025

2. forduló (online)

A feladatok megoldásának szabályai

- A 2. forduló három feladatot tartalmaz és összesen 11 kérdést.
- Minden kérdésre egy nemnegatív egész szám vagy egy string a válasz (az adott kérdésnél szerepel, hogy melyik). Ezeket az eredményeket kell beküldeni. A csapatnak ügyelnie kell arra, hogy pontosan a kérdésnél leírt módon küldje be a választ.
- Beküldési határidő: ***2025. március 8.*** *(23:59)*
*- A megoldások beküldését az alábbi linken található űrlapon keresztül kell elvégezni (****kizárólag a versenyre nevezett csapatoknak****):*

https://forms.gle/8xE45eRDotHFR6Vb6

- Minden nevezett csapat egyetlen alkalommal küldheti be (fordulónként) a megoldásait. Ha egy csapat többször is beküld megoldást, akkor a legkorábbit vesszük figyelembe a pontozásnál. Tehát csak akkor érdemes a megoldásokat beküldeni, ha valamennyi kérdésre megvan a válasz, vagy a csapat már nem tud vagy nem akar több feladatot megoldani. Kérjük, hogy nevezésnél megadott csapatnéven (karakterhelyesen) küldjék be a válaszokat! Ha a csapat elfelejtette a csapatnevét, akkor a nevezéskor kapott visszaigazoló e-mailben azt megtalálja.
- A megoldások részleteit nem kell beküldeni, csak a kérdésekre adott válaszokat (nemnegatív egész számok vagy stringek).
- A forduló helyes megoldásait és a csapatok pontszámait minden forduló lezárta után ismertetjük.

A 2. forduló feladatai

1. feladat

Egy hagyományos (12 órás ciklust mérő), analóg óra nagy- és kismutatója által bezárt szögről lesz szó. A feladat során feltételezzük, hogy mindig egész perceket mutat az óra (azaz a másodpercmutató mindig éppen a 12-esen áll). A két mutató által bezárt szögön mindig 0°-nál nem kisebb és 180°-nál nem nagyobb szöget értünk.

a) Az *idopontok.txt* állományban soronként két egész érték található szóközzel elválasztva, az időpontok szerint szigorúan monoton növekvő sorrendben. Az első érték az óra ($0 \leq h \leq 23$), a második a perc ($0 \leq m \leq 59$) értékét adja meg. Határozza meg, hogy az időpontok közül melyik esetben a legnagyobb a két mutató szöge! (7 pont)
A válasz egyetlen szövegérték legyen, két-két számjegyen megadva az óra és a perc értékét, közöttük kettősponttal (pl. *09:08*)!

b) Adja meg, hogy a fájlban közvetlenül egymást követő időpontokat figyelembe véve mekkora – abszolútértékben – a két mutató szögének legkisebb változása! (2 pont)
A válasz egyetlen számérték legyen! (pl. *21.6* vagy *32*) Tizedes határoló (ha szükséges) a pont.

c) A *szogek.txt* állományban egymás után, szóközzel elválasztva szögértékek állnak (fokban megadva). A tizedes határoló a pont. Az órát az 1. napon 00:00-kor kezdjük figyelni, ekkor a két mutató szöge 0° – a fájlban az első érték 0. A következő érték azt az időpontot határozza meg, amikor a két mutató éjfél után először zár be ekkora szöget. A harmadik érték azt az időpontot adja meg, amikor a két mutató a második időpont után először zár be ekkora szöget, és így tovább; lehetséges, hogy közben új nap kezdődik. (6 pont)

A válasz egyetlen szövegérték legyen! Elsőként adja meg, hogy hányadik napon fejeződött be a megfigyelés, ezt kövesse egy | karakter, majd – az a) részben leírt formátumban – az utolsó megadott időpontot (pl. *5*|*13:08*)!

2. feladat

A feladatban egy kockapóker mérkőzés adatait kell elemezni. Az alábbiakban leírt szabályrendszer részben eltér a hagyományos kockapóker játéktól, az egyszerűbb elemezhetőség miatt.
A játékban három játékos vett részt: alfa, beta, gamma (ebben a sorrendben dobtak).
Az első játékos egyszerre 5 db szabályos dobókockával dob. A dobás után eldönti, hogy melyik kockákat tartja meg és melyekkel dob újra. Akár mind az 5 kockával is újradobhat, de mind az 5-öt meg is tarthatja. Ezután a kiválasztott kockákkal újra dob (ha volt ilyen). Majd ismét dönthet, hogy az új dobások közül melyiket tartja meg és melyikkel dob újra. A korábban megtartott kockákkal már nem dobhat. Ezután harmadszor is dob a kiválasztott kockákkal. (Ha úgy döntött, hogy a második dobás után mind megtartja, akkor már nincs harmadik dobás.) Tehát minimálisan 5 kockát lehet dobni, maximálisan pedig 15-öt.
Ha a játékos befejezte a dobásait, akkor a kockákon látható pontértékek alapján eldönti, hogy a póker szabályai szerint mit ért el.
A lehetőségek:

1 pár: két kockán azonos számok vannak, a másik három kockán pedig különbözőek (egymástól és az azonosaktól is különbözniük kell) Pl.: 1, 1, 2, 3, 4

2 pár: két-két kockán azonos, de egymástól különböző számok vannak, az 5. kockán lévő szám is különbözik a többitől Pl.: 1, 1, 2, 2, 3

terc: három kockán van azonos szám, a másik kettőn különböző Pl.: 1, 1, 1, 2, 3

full: három kockán azonos szám és két kockán is azonos szám van, de különbözik a másik háromtól Pl.: 1, 1, 1, 2, 2

póker: négy kockán azonos szám van, az ötödiken különböző Pl.: 1, 1, 1, 1, 2

kissor: a számok a kockákon (sorrendtől függetlenül): 1, 2, 3, 4, 5

nagysor: a számok a kockákon (sorrendtől függetlenül): 2, 3, 4, 5, 6

A dobások után a játékos eldönti, hogy az öt kockáján lévő számok alapján mit dobott. Nem megengedett például egy fullt tercként elszámolni, vagy egy pókert 2 párként. A játékot az nyeri, aki először dobja meg mind a 7 lehetőséget (*1 pár, 2 pár, terc, full, póker, kissor, nagysor*). Ha a játékos dobása után olyan számok láthatók a kockákon, amelyek egyik lehetőségre sem jók, vagy már korábban dobott ilyet, akkor nem tudja elszámolni a dobása eredményét.
Az első játékos után következik a második, majd utána a harmadik. Ezzel egy kör zárul és indul a következő kör az első játékossal.
Amint valamelyik játokos mind a 7 lehetőséget megdobta, akkor a játék véget ér és Ő győzött. (A kört nem kell befejezni.)

A melléklet két fájl a *dobasok.txt* és a *dontesek.txt* egy teljes kockapóker játék adatait tartalmazza 3 játékos között. Tudjuk, hogy a játék valamelyik játékos győzelmével ért véget.
A *dobások.txt* fájlban 1 és 6 közötti véletlen számok szerepelnek határoló karakter nélkül. A játék során ezek voltak a dobások.
A *dontesek.txt* fájlban 1-esek és 0-k szerepelnek határoló karakter nélkül, amelyek az adott játékos döntését jelentik. Az 1-es a kocka megtartását, míg a 0 a kockával történő újra dobást jelenti.
A két fájlban azonos számú adat található és az adatok sorszáma kapcsolja össze őket. Tehát a *dontesek.txt* fájl 13. adata a 13. dobás megtartását vagy újra dobását jelenti.
Például:

Ha dobások rendre: 2325161245 a döntések pedig: 1010000111, akkor ez a következő játékot jelenti: az első öt dobás a 2 3 2 5 1 eredményt adta. Ebből a játékos megtartotta az 1 és 3. kockát, a másik hárommal újra dobott. Az új dobás eredménye a 6 1 2 volt. Innen megtartotta a 3. kockát és

a másik kettővel dobott. Ennek az eredménye 4 5 volt. Ezeket már meg kellett tartania, mert nem volt több dobási lehetősége. Így eredményként a 2 2 2 4 5 számokat kapta, ami egy terc.

A *dobások.txt* és *dontesek.txt* fájlok tartalma alapján oldja meg a következő feladatokat!

a) Hány teljes kör volt a játékban? Teljes kör alatt azt értjük, amikor mindhárom játékos dobott. A válasz egy szám. (2 pont)

b) Melyik játékos nyerte a játékot? A játékosok nevei a dobási sorrendnek megfelelően: alfa, beta, gamma. A válasz a három szó valamelyike (kisbetűvel, ékezet nélkül) (2 pont)

c) A harmadik játékos (gamma) hányadik körben dobott fullt? A körök számolását 1-gyel kezdje! A válasz egy szám. (5 pont)

d) Milyen számokból állt a legnagyobb póker? Pl.: egy 5-ösökből álló póker nagyobb, mint a 3-asokból álló póker. A válasz egy szám. (4 pont)

e) A három játékos hány párt dobott összesen a játék során? Azokat a párokat is számolja, amelyeket nem számoltak el, mert korábban már dobtak párt. A válasz egy szám. (4 pont)

3. feladat

A titkosírások egyik legegyszerűbb – bár könnyen megfejthető módja, ha a betűket egy táblázatba rendezzük, és a betű helyett annak koordinátáit használjuk (ld. bal oldali ábra). A koordináták sorrendjében megállapodhatnak a titkosírás használói (akár betűnként váltogathatnák is). Ha az első szám a sort, a második pedig az oszlopot azonosítja, akkor a *411543* számsorozat a *PER* szót adja meg. Ha a sorokat és az oszlopokat számok helyett magánhangzókkal helyettesítjük (ld. jobb oldali ábra), akkor az előző számsor helyett az *OAAEOI* betűsorozattal is leírhatjuk a *PER* szót.

	1	**2**	**3**	**4**	**5**
1	A	B	C	D	E
2	F	G	H	I	J
3	K	L	M	N	O
4	P	Q	R	S	T
5	U	V	X	Y	Z

	A	**E**	**I**	**O**	**U**
A	A	B	C	D	E
E	F	G	H	I	J
I	K	L	M	N	O
O	P	Q	R	S	T
U	U	V	X	Y	Z

a) A *szoveg.txt* fájlban található magánhangzók alapján fejtse meg a titkosírással írt szöveget. A szavakat most szóköz vagy sortörés választja el egymástól (igazi titkosírásnál ezzel nem segítenénk a megfejtést). Írásjelet nem használtunk, és csak a táblázatban szereplő betűk nagybetűs alakja szerepel a szövegben. Az első betűpár első betűje a sort, a második betűje az oszlopot azonosítja a jobb oldali táblázatban, a következő betűpárnál fordítva, majd minden betűpár után újra felcserélve. A szóközöknél, sortöréseknél nem cseréljük fel a koordinátákat. (Tehát értelmezze a szöveget úgy, mintha egyetlen szóból állna!) (6 pont)
Válaszként adja meg, hány szó szerepel az első és az utolsó olyan szó között, amelynek megfejtett alakjában szerepel a *Q* karakter (ha pl. az első ilyen szó a szöveg 6. szava, az utolsó ilyen szó pedig a szöveg 15., akkor a válasz 8). Kihasználhatja, hogy nincs olyan szó, amelyben két *Q* karakter is lenne.

b) A kód kitalálását nehezíti, ha a magánhangzókkal kódolt szöveget mássalhangzókkal egészítjük ki úgy, hogy az így kapott szöveg értelmesnek tűnjön, de továbbra is csak a magánhangzók hordoznak jelentést. A *szoveg2.txt* fájlban lévő (korlátozottan) értelmes szövegben elrejtett üzenetet kell megfejteni. A koordináták sorrendjét nem cseréltük fel a titkosítás során. (4 pont)
Válaszként a megfejtett szöveget kell beküldeni (szóközök nélkül nagybetűsen).

c) Kódolja a *szoveg3.txt* fájlban lévő szöveget koordinátaváltás nélkül, első koordináta alatt a sort értve. Ezután a *szavak.txt* fájlban található szavak segítségével egészítse ki értelmes szavakká a kapott magánhangzókat. A szóban a magánhangzók sorrendje egyezzen meg koordináták sorrendjével! A *szavak.txt* fájlban minden magánhangzópárhoz pontosan egy szót talál, rendezetlenül, a szavak külön sorokban állnak (valódi titkosításnál egy nagy szóadatbázisból válogatnánk véletlenszerűen, nem csak két magánhangzót tartalmazó szavakkal). (7 pont)
Válaszként a titkosított, kiegészített szövegeket kell beküldeni, szóközökkel tagolva.