



OKTATÁSI HIVATAL

**A 2023/2024. tanévi  
Országos Középiskolai Tanulmányi Verseny  
első forduló**

**DIGITÁLIS KULTÚRA II. (PROGRAMOZÁS) KATEGÓRIA  
Javítási-értékelési útmutató**

Kérjük a tisztelt tanár kollégákat, hogy a dolgozatokat – az egységes értékelés érdekében – szigorúan az alábbi útmutató szerint pontozzák, a megadott részpontoszámokat ne bontsák tovább! Vagyis, ha egy részmegoldásra pl. 3 pontot javasolunk, akkor arra vagy 0, vagy 3 pont adható. (Az útmutatótól eltérő megoldások is lehetnek jók.) Az egyes részmegoldásokat az útmutatóban pontosvesszővel választjuk el.

**Összpontszám: 400 pont**

**Beküldési ponthatár: 200 pont**

**1. feladat: Mi a különbség? (54 pont)**

Bendegúz egy feladatra két megoldást is írt és azt gondolta, hogy ugyanazt számolják ki egy legalább 5 elemű A tömbre.

első:

```
S:=0
Ciklus i=3-tól N-1-ig
    S:=A[i]+A[i-1]+A[i-2]+S
Ciklus vége
Ki: S
```

Eljárás vége.

második:

```
S1:=0; S2:=0; S3:=0
Ciklus i=2-től N-ig
    S1:=S1+A[i-1]
    Ha i>1 és i<N-1 akkor S2:=S2+A[i-1]
    Ha i>2 és i<N-2 akkor S3:=S3+A[i-1]
Ciklus vége
Ki: S1+S2+S3
```

Eljárás vége.

- A.** Mit számol az első és a második megoldása és mi a különbség köztük?
- B.** Tetszőleges bemenetre mit számít az első? Add meg az eredményt az alábbi tömbre is: (1,2,3,4,5,6).
- C.** Tetszőleges bemenetre mit számít a második? Add meg az eredményt az alábbi tömbre is: (1,2,3,4,5,6).
- D.** Fogalmazd meg általánosan a különbséget a két eredmény között!

Az Országos Középiskolai Tanulmányi Versenyek megvalósulását az NTP-TMV-M-23 projekt támogatja



KULTURÁLIS ÉS INNOVÁCIÓS  
MINISZTERIUM

 **Nemzeti  
Tehetség Program**

**Értékelés:**

- A. Mindkettő A elemeit adja össze 4 pont
- B. Az első 1-től N-1-ig; plusz 2-től N-2-ig; plusz 3-tól N-3-ig; első=15+9+3=27 5+5+5+5 pont  
(Képlettel:  $A[1]+2*A[2]+3*A[3]+\dots+3*A[N-3]+2*A[N-2]+A[N-1]$ )
- C. A második 1-től N-1-ig; plusz 1-től N-3-ig; plusz 2-től N-4-ig; második=15+6+2=23 5+5+5+5 pont  
(Képlettel:  $2*A[1]+3*A[2]+\dots+3*A[N-4]+2*A[N-3]+A[N-2]+A[N-1]$ )
- D. A különbség:  $A[N-2]+A[N-3]-A[1]-A[2]$  10 pont

**2. feladat: Mit csinál? (40 pont)**

A következő programrészlet bemenete az A[1..N] tömb, értékei 0 és M-1 közötti egész számok.

```

Ciklus I=0-tól M-1-ig
  B[I]:=0
Ciklus vége
Ciklus I=1-től N-ig
  B[A[I]]:=B[A[I]]+1
Ciklus vége
(*)
Ciklus I=1-től M-1-ig
  B[I]:=B[I]+B[I-1]
Ciklus vége
(**)
Ciklus I=1-től N-ig
  C[B[A[I]]]:=A[I]; B[A[I]]:=B[A[I]]-1
Ciklus vége
(***)

```

- A.** Mit tartalmaz a megoldásban a B[0..M-1] tömb a (\*) ponton?
- B.** Mit tartalmaz a megoldásban a B[0..M-1] tömb a (\*\*) ponton?
- C.** Mit tartalmaz a megoldásban a B[0..M-1] és a C[1..N] tömb a (\*\*\*) ponton?

**Értékelés:**

- A.  $B(j)$ = az A tömb j értékű elemeinek száma 10 pont
- B.  $B(j)$ = az A tömb j értékű vagy annál kisebb elemei száma 10 pont
- C.  $B(j)$ = az A tömb j-nél kisebb elemei száma 10 pont
- C = az A tömb elemei nagyság szerint növekvően 10 pont

**3. feladat: Kódolás (40 pont)**

Számítógéppel titkosítottak egy értelmes magyar SZÓ-t. A program lefutása után közölték velünk az előállított KÓD-ot, de a használt KULCS-ot nem.

A kódoláskor használt algoritmus a következő volt:

```

Kódolás:
KÓD[0]:=KULCS
Ciklus I=1-től HOSSZ-ig
  KÓD[I]:=SZÓ[I] művelet KÓD[I-1]
Ciklus vége
Eljárás vége.

```

Az 1-től HOSSZ-ig indexelt SZÓ tömb a kódolandó, a 0-tól HOSSZ-ig indexelt KÓD tömb pedig a kódolt szót tartalmazza.

A KÓD (1-től HOSSZ-ig) ez lett: CJNKL. Az alábbi igazságtáblával megadott műveletet a számítógép az operandusokon bitenként végzi el (A és B az operandusok, C az eredmény).

Művelet:

A	0	1	0	1
B	0	0	1	1
C	0	1	1	0

A használható betűk és kódjaik:

- 0000	A 0001	B 0010	C 0011
D 0100	E 0101	F 0110	G 0111
H 1000	I 1001	J 1010	K 1011
L 1100	M 1101	N 1110	O 1111

Mi volt az eredeti, értelmes magyar szó (megjegyzés: az első betű kitalálásához az algoritmus nem ad segítséget), és mi volt a KULCS? Magyarázd meg, hogyan jöttél rá a megoldásra!

**Értékelés:**

A. SZÓ = HIDEG 10 pont

B. KULCS = K 10 pont

C. A művelet  $KÓD[I-1]$  operandusából és  $KÓD[I]$  eredményéből kiszámítható  $SZÓ[I]$  operandusa, és így a szó hátulról visszafelé haladva helyreállítható, 10 pont

vagy

az itt alkalmazott XOR művelet asszociatív és kommutatív; ha két operandusa azonos, akkor 0 az eredménye; ha egyik operandusa 0, akkor az eredménye azonos a másik operandusával. Ezért  $SZÓ[I] = KÓD[I]$  művelet  $KÓD[I-1]$ , ami az első betű kivételével mindig igaz

D. Az első betűre pedig  $KULCS = KÓD[0] = SZÓ[1]$  művelet  $KÓD[1]$  igaz 10 pont

#### 4. feladat: Automata (60 pont)

Egy automata különböző egész számokat tartalmazó sorozatot dolgoz fel. Egyesével balról-jobbra haladva veszi a sorozat számait és egy sor adatszerkezet felhasználásával végez műveletet. Kezdetben a sor üres. Az aktuális számmal az alábbi műveletek közül az első olyat hajtja végre, amelyre a feltétel teljesül:

- ha a sor üres, a végére teszi
- ha a sor utolsójánál nagyobb, a végére teszi
- ha az utolsó kiírtnál kisebb, akkor hibajelzéssel megáll
- ha a sor elsőjénél kisebb, akkor kiírja
- amíg a sor elsőjénél nagyobb, addig kiírja a sor elsőjét (és a kiírt elemet ki is veszi a sorból); ilyenkor, ha a sor üressé válik, akkor a bemeneti értéket a sorba teszi, különben kiírja
- az utolsó érték feldolgozása után a sorban levőket kiírja

A. Minden elem érkezése után add meg a sorban levő elemeket és a kiírt értékeket az alábbi bemenetre: 3 4 6 1 8 2 5 9 7.

B. Mi a feltétele annak, hogy hibajelzéssel áll meg a feldolgozás? Adj meg egy konkrét sorozatot, amire hibajelzéssel megáll!

**Értékelés:**

A. 1. sor=(3), kimenet=() 2+2 pont

2. sor=(3,4), kimenet=() 2+2 pont

3. sor=(3,4,6), kimenet=()	2+2 pont
4. sor=(3,4,6), kimenet=(1)	2+2 pont
5. sor=(3,4,6,8), kimenet=(1)	2+2 pont
6. sor=(3,4,6,8), kimenet=(1,2)	2+2 pont
7. sor=(6,8), kimenet=(1,2,3,4,5)	2+2 pont
8. sor=(6,8,9), kimenet=(1,2,3,4,5)	2+2 pont
9. sor=(8,9), kimenet=(1,2,3,4,5,6,7)	2+2 pont
10. sor=(), kimenet=(1,2,3,4,5,6,7,8,9)	2+2 pont
B. 3,2,1-re hibajelzés lesz	8 pont
minden olyan sorozatra hibajelzést ad, amiben van olyan $i < j < k$ index, hogy $A[i] > A[j] > A[k]$	12 pont

### 5. feladat: Assembly (54 pont)

Az assembly nyelv a számítógép processzora által futtatott egyszerű gépi utasítások szöveges formája. Ciklusokat és elágazásokat a kód megfelelő részére – melyet egy szöveges címke jelöl – való feltételes ugrással képezhetünk benne. Változókra nem névvel utalunk, hanem az alapján, hogy a processzor melyik regiszterében (belső tárolóegység) található.

Az alábbi program az A és B memóriacímen lévő, 64 bites pozitív egész számokból számol ki egy 64 bites, nemnegatív egész értéket, majd azt a C címen tárolja. A segédszámításokhoz hat darab 64 bites regisztert (x0-x5) használ.

```

ldr    x0, A           ; az x0 regiszterbe az A címen található értéket betölti
ldr    x1, B           ; az x1 regiszterbe a B címen található értéket betölti
orr    x4, x0, x1      ; az x4 regiszterbe az x0-n és x1-en végzett bitenkénti VAGY
                        ; eredményét tölti
ctz    x4, x4          ; az x4 regiszterbe az x4 érték végén található 0 bitek számát tölti
ctz    x2, x0          ; az x2 regiszterbe az x0 érték végén található 0 bitek számát tölti
lsr    x0, x0, x2      ; az x0 regiszter tartalmát x2-szor jobbra lépteti
ctz    x3, x1          ; az x3 regiszterbe az x1 érték végén található 0 bitek számát tölti
lsr    x1, x1, x3      ; az x1 regiszter tartalmát x3-szor jobbra lépteti
X:
cmp    x0, x1          ; összehasonlítja x0 és x1 értékét
beq    Z               ; a Z címkére ugrik, ha egyenlők
bhi    Y               ; az Y címkére ugrik, ha x0 > x1
mov    x5, x1          ; az x5 regiszterbe x1 értékét tölti
mov    x1, x0          ; az x1 regiszterbe x0 értékét tölti
mov    x0, x5          ; az x0 regiszterbe x5 értékét tölti
Y:
sub    x0, x0, x1      ; x0 értékéből kivonja x1 értékét
ctz    x5, x0          ; az x5 regiszterbe az x0 érték végén található 0 bitek számát tölti
lsr    x0, x0, x5      ; az x0 regiszter tartalmát x5-szor jobbra lépteti
b      X               ; az X címkére ugrik
Z:
lsl    x0, x0, x4      ; az x0 tartalmát x4-szer balra lépteti
str    x0, C           ; a C memóriacímen tárolja x0 értékét

```

Kiegészítés: egy kettes számrendszerbeli szám eggyel jobbra léptetése azt jelenti, hogy a legalacsonyabb helyiértékű bitet töröljük, az elejére pedig 0-t teszünk, például:  $1011_b \gg 1 == 0101_b$ . Balra léptetéskor pedig a legnagyobb helyiértékű bitet töröljük és a végére 0-t írunk, például 4 biten:  $1011_b \ll 1 == 0110_b$ .

- A.** Mi lesz a C címen, ha  $A=12$ ,  $B=9$ ?
- B.** Mi lesz a C címen, ha  $A=15$ ,  $B=7$ ?
- C.** Mi lesz a C címen, ha  $A=27$ ,  $B=3$ ?
- D.** Hogy függ a C címen található érték A-tól és B-től? Írd le általánosan!
- E.** Hogyan viszonyul  $x_4$  értéke a bemeneti értékek végén található 0 bitek számához?
- F.** Mit csinál az X és Y címkék közti kódrészlet? Írd le szövegesen!

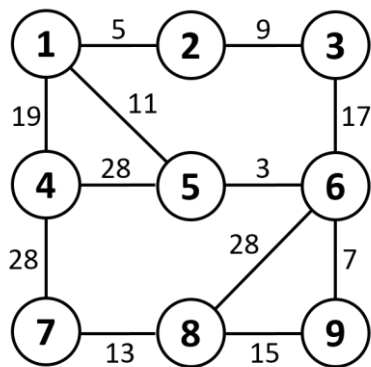
**Értékelés:**

- A.  $C=3$  6 pont
- B.  $C=1$  6 pont
- C.  $C=3$  6 pont
- D. A C-n található érték az A és B értékek legnagyobb közös osztója 12 pont
- E. Azok minimuma. 10 pont
- F. Ha  $x_0$  és  $x_1$  értékek egyenlők, kilép a ciklusból; ha  $x_0 < x_1$ , a két regiszter értékét felcseréli 7+7 pont

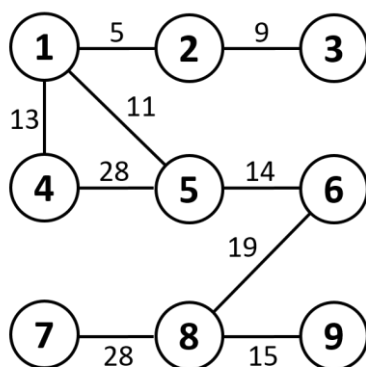
**6. feladat: Utak (80 pont)**

Bergengóciában a városokat földutak kötik össze. A király elhatározza, hogy az utakat leaszfaltozzák, de sajnos nincs annyi pénze, hogy minden utat aszfaltozzanak. A királynak K forintja van. A következő stratégiát választotta: Először aszfaltozzák a fővárosból kiinduló legkevesebbe kerülő utat. Ezután minden lépésben a legkisebb költségű olyan U és V város közötti utat választják, ahol U elérhető a fővárosból (az 1-es sorszámú városból) már aszfaltozott úton, de V még nem.

Add meg az alábbi két városhálózatra, hogy ezzel a szabállyal mely utak lesznek leaszfaltozva, az aszfaltozás sorrendjében és a királynak mennyi pénze marad!



- A.**
- A1.** A királynak 30 forintja van kezdetben.
- A2.** A királynak 50 forintja van kezdetben.



B.

B1. A királynak 40 forintja van kezdetben.

B2. A királynak 80 forintja van kezdetben.

**Értékelés:**

A1. 1-2, 2-3, 1-5, 5-6; marad 2 forint utanként 3 pont; jó sorrend 4 pont; jó maradék 4 pont

A2. 1-2, 2-3, 1-5, 5-6, 6-9, 8-9; marad 0 forint utanként 2 pont; jó sorrend 4 pont; jó maradék 4 pont

B1. 1-2, 2-3, 1-5, 1-4; marad 2 forint utanként 3 pont; jó sorrend 4 pont; jó maradék 4 pont

B2. 1-2, 2-3, 1-5, 1-4, 5-6, 6-8; marad 9 forint utanként 2 pont; jó sorrend 4 pont; jó maradék 4 pont

**7. feladat: Siklóernyő (72 pont)**

Alex siklóernyőzni szeretne a Tatra hegycsúcsain. A hegycsúcsok a Tátrában négyzetrácsosan N sorban és azon belül M oszlopban helyezkednek el és mindegyiknek ismert a magassága. A siklóernyőzés során Alex elindul az egyik kiválasztott hegycsúcsról és átrepül valamelyik (nem feltétlenül szomszédos), a jelenleginél szigorúan kisebb magasságú hegycsúcsra, mely a jelenlegi csúccsal azonos sorban vagy azonos oszlopban helyezkedik el. Egy csúcsról akkor is átmehet egy alacsonyabb hegycsúcsra, ha a kettő között van magasabb hegycsúcs - ezeket a levegőben kikerüli.

Alex a siklóernyőzés során több alkalommal is átrepülhet a jelenlegi csúcsról egy következőre (függetlenül attól, hogy másik repüléssel már járt ott), és bármelyiken be is fejezheti a siklóernyőzést, feltéve, hogy legalább egy átrepülést végrehajtott már.

Add meg a lentebb megadott magasságok esetén **minden egyes hegycsúcsra**, hogy onnan kiindulva

**A.** Hány különböző siklóernyőzést tehet, mely során pontosan egy alkalommal repül át másik csúcsra?

**B.** Hány különböző siklóernyőzést tehet, mely során pontosan két alkalommal repül át másik csúcsra?

**C.** Hány különböző siklóernyőzést tehet összesen?

2	1	3	5
1	1	1	4
9	2	7	6

**Értékelés:**

- A. Minden helyes válasz 1 pontot ér.

Szabály: a helyes érték a cellával azonos sorban és oszlopban lévő, kisebb magasságértékeket tartalmazó cellák száma.

2	0	3	4
0	0	0	3
5	2	4	3

- B. Minden helyes válasz 2 pontot ér.

Szabály: a helyes érték a cellával azonos sorban és oszlopban lévő, kisebb magasságértékeket tartalmazó cellákra az A-beli válaszok összege.

0	0	2	8
0	0	0	0
11	0	8	9

- C. Minden helyes válasz 3 pontot ér.

Szabály: első megoldás naív számolással a B módszerét megismételni (max. 7 lépés -> 7 táblázat) és összegezni az értékeket. Második megoldás nulláról számolva növekvő sorrendben haladunk a magasságértékeken. Tetszőleges cellára a helyes válasz a vele azonos sorban és oszlopban lévő, kisebb magasságértékű cellákra a helyes válasz értéke plusz az ilyen cellák darabszáma.

2	0	5	14
0	0	0	3
64	2	33	22