



A 2016/2017 tanévi Országos Középiskolai Tanulmányi Verseny első forduló javítási-értékelési útmutató

INFORMATIKA II. (programozás) kategória

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.)

1. feladat: Vállalkozás (27 pont)

Egy vállalkozás 1 napos munkákra kapott megbízást, 1 nap 1 munkát tud elvégezni. Ismerjük minden megbízás határidejét és az elvégzett munkáért kapott fizetséget.

Add meg, hogy az alábbi megrendelések (a számpárok első tagja a határidő, a második a fizetés) esetén maximum mennyit kereshet a vállalkozás és ehhez mely munkákat, mely napokon kell elvégeznie.

A. 1:(3,60), 2:(4,40), 3:(1,10), 4:(3,30), 5:(7,70), 6:(4,20)

B. 1:(4,400), 2:(8,200), 3:(1,200), 4:(4,700), 5:(9,300), 6:(6,700), 7:(7,900), 8:(7,400), 9:(10,100), 10:(2,500)

C. 1:(1,30), 2:(2,420), 3:(2,450), 4:(4,650), 5:(5,930), 6:(1,100), 7:(3,860), 8:(3,50), 9:(4,300), 10:(2,930)

Értékelés: Fizetés szerint csökkenő sorrendbe rendezzük az igényeket, majd mohó stratégiával mindegyiket a legutolsó szabad napra tesszük a határideje előtt.

A. 220, 6 4 1 2 5 (a 4-es és az 1-es lehet az első 3 nap bármelyikén, a 6-os és a 2-es az első 4 nap bármelyikén, és az 5-ös az 5-7. napok közül bármelyikén) 5+4 pont

B. 4400, 3 10 1 4 8 6 7 2 5 9 (1 4 fordítva is lehet, 8 6 7 sorrendje tetszőleges, de a 6 közülük nem lehet a harmadik helyen) 5+4 pont

C. 3820, 3 10 7 4 5 (3 10 fordítva is lehet) 5+4 pont

2. feladat: Sorrend (30 pont)

Egy könyvespolcon egy könyvsorozat N tagja helyezkedik el, tetszőleges sorrendben. A sorba-rendezés egy lépésében egy tetszőleges könyvet levehetünk a könyvespolcra és egy másik helyre visszatehetjük. (Nem számít külön lépésnek, hogy emiatt a könyv régi és új helye közötti könyveket egy hellyel eltoljuk, hogy a könyvek megint egymás mellett legyenek.)

Add meg, hogy az alábbi könyvsorozatok minimum hány lépésben rendezhetők növekvő sorrendbe és mely könyveket kell ehhez levenni a polcra! Több megoldás esetén bármelyik megadható.

A. 8 könyv, sorrendjük: 1,5,2,6,3,8,4,7

B. 10 könyv, sorrendjük: 8,2,5,4,1,6,3,9,7,10

C. 12 könyv, sorrendjük: 11,1,10,4,3,2,8,7,12,6,9,5

D. 20 könyv, sorrendjük: 2,17,5,9,6,16,8,3,4,10,1,12,7,18,13,20,14,15,11,19

Értékelés: Ha megkeressük a leghosszabb növekvő részsorozatot, akkor az abban levő könyveket nem kell mozgatni.

A. 3 lépésben rendezhető; mozgatni kell: 5,6,8 2+3 pont

B. 5 lépésben rendezhető; mozgatni kell: (1,3,5,7,8) vagy (1,3,5,8,9) vagy (1,3,4,7,8) vagy (1,3,4,8,9) 4+3 pont

C. 8 lépésben rendezhető; mozgatni kell: (2,3,5,6,7,9,10,11) vagy (2,4,5,6,7,9,10,11) vagy (3,4,5,6,7,9,10,11) vagy (2,3,5,6,8,9,10,11) vagy (2,4,5,6,8,9,10,11) vagy (3,4,5,6,8,9,10,11) vagy (2,3,5,6,7,10,11,12) vagy (2,4,5,6,7,10,11,12) vagy (3,4,5,6,7,10,11,12) vagy (2,3,5,6,8,10,11,12) vagy (2,4,5,6,8,10,11,12) vagy (3,4,5,6,8,10,11,12) vagy (2,3,5,7,8,10,11,12) vagy (2,4,5,7,8,10,11,12) vagy (3,4,5,7,8,10,11,12) 4+4 pont

D. 10 lépésben rendezhető; mozgatni kell: 1,3,4,7,9,11,16,17,18,20 5+5 pont

3. feladat: Rendezés (33 pont)

Egy síterepen az egyes csomópontok között lehetnek síelési útvonalak. Add meg az alábbi útvonalak alapján a csomópontok egy olyan sorrendjét, amiben minden csomópont megelőz minden olyan csomópontot, ahova síelési útvonalon el lehet jutni belőle! (Több megoldás esetén bármelyiket.) Add meg azt is, hogy melyik csomópontba hányféleképpen lehet eljutni olyan csomópontból, ahova nem vezet útvonal! (Amely csomópontba nem vezet útvonal, ott ez a szám 0.)

A. (5,2), (1,3), (1,4), (2,3)

B. (9,5), (6,5), (10,6), (10,9), (3,2), (3,10), (8,3), (1,8), (7,8), (4,1), (1,10)

Értékelés: A lehetséges sorrendek a kapcsolatokat leíró gráf topologikus rendezései.

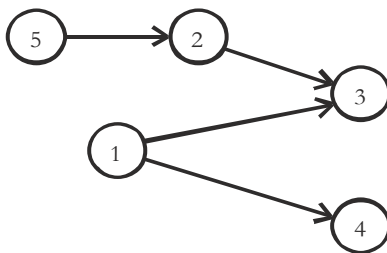
A. Jó sorrend pl.: (5,2,1,3,4)

az (5,1,2) megelőzi a (3)-at 1 pont

az (1) megelőzi (4)-et 1 pont

az (5,2) sorrend rögzített, az 1 hozzájuk képest bárhol lehet 2 pont

a (3,4) sorrendje tetszőleges, de a 3 csak az utolsó 2 helyen lehet 2 pont



Az 1-be 0, 2-be 1, 3-ba 2, 4-be 1, 5-be 0-féleképpen lehet eljutni 5*1 pont

B. Az eleje (4,1,7), (4,7,1), (7,4,1) lehet 2 pont

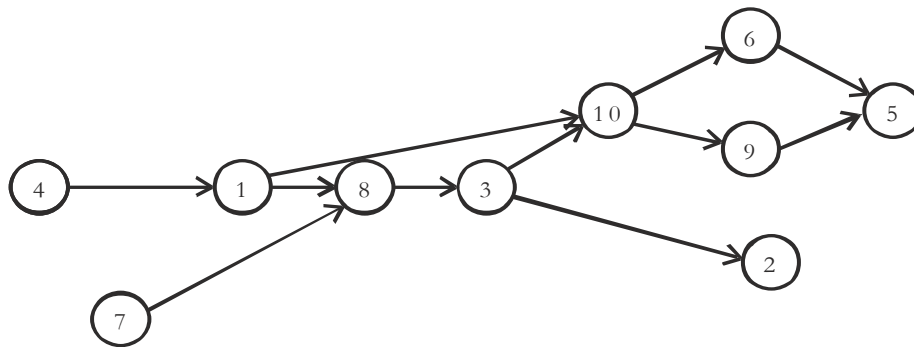
ezt mindenképpen a (8,3) követi 2 pont

a 2 ettől kezdve bárhol lehet 2 pont

a (10,6,9,5)-ből a 10-nek kell következnie először 2 pont

amit a 6 és 9 tetszőleges sorrendben követhet 2 pont

az 5 pedig e négy szám közül az utolsó 2 pont



Az 1-be 1, 2-be 2, 3-ba 2, 4-be 0, 5-be 6, 6-ba 3, 7-be 0, 8-ba 2, 9-be 3, 10-be 3-féleképpen lehet eljutni 10*1 pont

4. feladat: Titkosítás (40 pont)

Egy titkosítási eljárás úgy működik, hogy összekeverik a kapott számok kettes számrendszerbeli bitjeit. A keverést egy vektor írja le, amely i . elemében megadjuk, hogy a titkos kód i . bitjét a szám hányadik bitjéből kell venni (balról 0-tól sorszámozva a biteket). Például, ha a keverést a $(2,1,3,0)$ vektor írja le, akkor a 0000, 1100, 0111 számok titkos kódjai a 0000, 0101, 1110 számok lesznek.

Add meg a keverést leíró vektort, ha a számok és a kódjuk az alábbi (a számok és a kódok sorrendje nem feltétlenül azonos, több megoldás esetén bármelyik megadható):

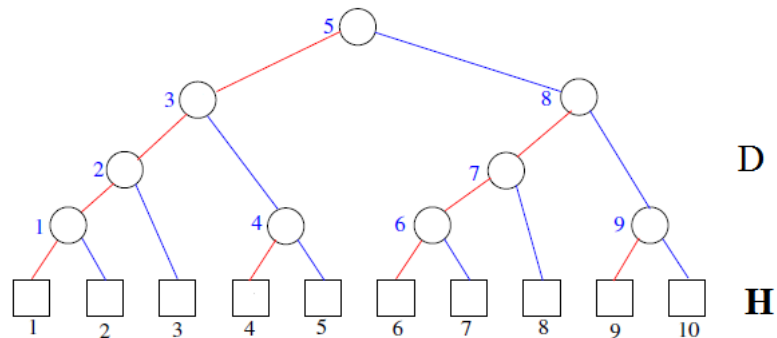
- A. Számok: 0011, 0110, 1100, 1010, 1011. Kódjuk: 1110, 1010, 1100, 1001, 0011.
- B. Számok: 1010, 0101, 0110, 1001. Kódjuk: 0011, 1100, 0101, 1010.
- C. Számok: 00110, 00101, 10000, 10010, 01001. Kódjuk: 01100, 10001, 01010, 00100, 10010
- D. Számok: 01011, 00110, 10010, 11010. Kódjuk: 11010, 10100, 10001, 10011.

Értékelés: Lehet olyan szám, aminél egyértelmű, hogy melyik a kódja – ha pl. egyetlen 1-es vagy egyetlen 0-s bitje van, abból egy helyről már lesz információ. Ezt az információt fel kell használni a többi számnál!

- A. $(2,3,0,1)$ 4*2 pont
- B. $(0, 2, 3, 1)$ vagy $(0, 3, 2, 1)$ vagy $(1, 2, 3, 0)$ vagy $(1, 3, 2, 0)$ vagy $(2, 0, 1, 3)$ vagy $(2, 1, 0, 3)$ vagy $(3, 0, 1, 2)$ vagy $(3, 1, 0, 2)$ 4*3 pont
- C. $(4,3,0,2,1)$ 5*2 pont
- D. $(3,4,2,1,0)$ 5*2 pont

5. feladat: Mit csinál? (25 pont)

Az alábbi algoritmus a 10 elemű H logikai vektor alapján számítja ki egy függvényértéket és tölti ki a 9 elemű D vektort. Az alábbi ábrán a H vektor elemeit négyzetekkel, a D vektorét pedig körökkel jelöljük:



A. Mi lesz az $\text{Épít}(1,10)$ függvény értéke és a D vektor tartalma, ha $H=(\text{hamis, igaz, hamis, igaz, igaz, hamis, igaz, hamis, igaz, igaz})$?

B. Fogalmazd meg általánosan is, hogy H-tól függően mi az $\text{Épít}(1,10)$ függvény értéke és a D vektor tartalma!

```

Épít (bal, jobb) :
  Ha bal < jobb
    akkor s = egészrész ((bal+jobb) / 2)
           b := Épít (bal, s); j := Épít (s+1, jobb)
           D(s) := j; Épít := b+j
    különben ha H(bal) akkor Épít := 1
           különben Épít := 0
  
```

Eljárás vége.

Értékelés:

- A. $D=(1,0,2,1,3,1,0,2,1)$, függvényérték=6 6+4 pont
- B. A függvényérték a H-ban levő igaz értékek száma 5 pont
 $D(i)$ az ábrán az i jelű körtől jobbra levő ágon szereplő (azaz az i gyökerű fa jobb részfájában levő) H-beli igaz értékek száma 10 pont

6. feladat: Mit csinál? (25 pont)

Nyolc ember (0-tól 7-ig sorszámozva) mindegyike 1 és 10 közötti minősítést szerzett. Szeretnének ebből valamilyen értéket kiszámolni, de senki sem szeretné, ha az ő minősítését egynél több másik ember ismerné. Kitaláltak egy algoritmust, amelyet ha mindenki végrehajt, akkor a kívánt érték a feltételeknek megfelelően kiszámolható. Az algoritmusban adott rendszer szerint üzenetet kell küldeniük egymásnak és az általuk tudott számot közölni a partnerrel, majd a saját és a kapott számból egy új értéket kell kiszámolni.

Az összes ember programja így néz ki:

Program:

```
i:=1; P:=Azonositom()
Ciklus amíg i*2<8
  Ha P mod (i*2)<i akkor Kuld(P+i,A); B:=Fogad(P+i)
  különben Kuld(P-i,A); B:=Fogad(P-i)
  Ha B<A akkor A:=B
  i:=i*2; Ki: A
Ciklus vége
```

Eljárás vége

A Kuld(kinek, mit) művelet nem blokkoló, hatására a kinek pufferébe kerül a mit érték (egy egész) és tovább folytatódik a program. A Fogad(kitol) blokkoló, azaz amíg a kitol nem hajtott végre Kuld(en, x) műveletet (ahol en a Fogad végrehajtója), addig várakozik. Az Azonositom() művelet megadja a végrehajtó ember azonosítóját (0..7).

A. Mik lesznek a 8 programban a kiírt számok, ha kezdetben a 8 program A változója: 7, 9, 4, 10, 10, 6, 2, 2?

Program	0.	1.	2.	3.	4.	5.	6.	7.
1. kiírás:								
2. kiírás:								
3. kiírás:								

B. Fogalmazd meg általánosan, mi lesz az egyes programok által kiszámított utolsó A érték!

Értékelés:

A. Minden helyes sor 5 pont

3*5 pont

Program	0.	1.	2.	3.	4.	5.	6.	7.
1.kiírás:	7	7	4	4	6	6	2	2
2. kiírás:	4	4	4	4	2	2	2	2
3. kiírás:	2	2	2	2	2	2	2	2

Az utolsó sor 5 pontja megadható az alábbi két válasz típus esetén is:

- Ha eggyel tovább menne, a harmadik sorba csupa 2-es kerülne
- Ebben a változatban 3. kiírás nincs

B. A= minimumot számol, egyre több számból

10 pont

7. feladat: Függvény (20 pont)

Egy képzeletbeli programozási nyelven az x -ekhez és y -okhoz z -ket rendelő f függvény típusát $f :: x \rightarrow y \rightarrow z$ alakban írjuk le, az a típusú elemekből álló lista típusát $[a]$ -val, az a és b típusú elemekből álló párt pedig (a, b) -vel jelöljük. Az üres listát $[]$ jelöli. Hosszabb listákat a $(:)$ művelettel állíthatunk elő egy elemből és egy listából. Például: $1 : [2, 3, 4] == [1, 2, 3, 4] == 1 : (2 : (3 : (4 : [])))$. A függvények működése különbözhet az argumentumok szerkezetétől függően (ezt mintaillesztésnek hívjuk).

Írtunk egy függvényt:

```
valami :: [a] -> [b] -> [(a,b)]
valami as [] = []
valami [] bs = []
valami (a:as) (b:bs) = (a,b) : valami as bs
```

A. Mi a valami $[1, 2, 3, 4] ['a', 'b', 'c', 'd', 'e', 'f']$ értéke?

B. Mit csinál a függvény, általánosan megfogalmazva?

A függvényt így módosítottuk:

```
valami :: [a] -> [b] -> [(a,b)]
valami as [] = []
valami [] (b:bs) = (0,b) : valami [] bs
valami (a:as) (b:bs) = (a,b) : valami as bs
```

C. Mi lesz most a valami $[1, 2, 3, 4] ['a', 'b', 'c', 'd', 'e', 'f']$ értéke?

D. Mit csinál a függvény, általánosan megfogalmazva?

Értékelés:

- | | |
|---|------------------|
| A. $[(1, 'a'), (2, 'b'), (3, 'c'), (4, 'd')]$ | 4 pont |
| B. Pozícionként párosítja a két kapott lista elemeit a rövidebb hosszágig. | 2 pont
4 pont |
| C. $[(1, 'a'), (2, 'b'), (3, 'c'), (4, 'd'), (0, 'e'), (0, 'f')]$ | 4 pont |
| D. Pozícionként párosítja a két kapott lista elemeit ha a második hosszabb, akkor az első listát nullákkal egészíti ki. | 2 pont
4 pont |

Összpontszám: 200 pont

Beküldési határ: 80 pont