

Filiera teoretică, profil real, specializare matematică-informatică / matematică-informatică intensiv informatică
Filiera vocatională, profil militar, specializare matematică-informatică

- Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.
- Timpul de lucru efectiv este de 3 ore.
- Identificatorii utilizați în rezolvări trebuie să respecte precizările din enunț (**bold**), iar în lipsa unor precizări explicite, notațiile trebuie să corespundă cu semnificațiile asociate acestora (eventual în formă prescurtată). Datele de intrare se consideră corecte, validarea lor nefiind necesară.
- În grafurile din cerințe oricare arc/muchie are extremități distincte și oricare două arce/muchii diferă prin cel puțin una dintre extremități.

Az 1-től 5-ig sorszámozott ítemek esetén, írja a vizsgalapra a helyes válasznak megfelelő betűjelet. Minden helyes válasz 4 pontot ér.

- Jelöld meg, mennyi lesz a tömb főátlóján lévő elemek összege a mellékelt kódrész lefutása után.

3. A backtracking módszert használva generáljuk az összes lehetséges módon George Enescu 3 különböző műalkotásából összeállítható listát a {Oedip, Poema română, Rapsodia română nr. 1, Rapsodia română nr. 2, Simfonia nr. 1} halmazból. Két lista különböző, ha legalább egy műalkotásban vagy a műalkotások sorrendjében különböznek. Az első négy megoldás a generálás sorrendjében: (Oedip, Poema română, Rapsodia română nr. 1), (Oedip, Poema română, Rapsodia română nr. 2), (Oedip, Poema română, Simfonia nr. 1), (Oedip, Rapsodia română nr. 1, Poema română). Jelöld meg az utolsó előtti megoldást.

4. Egy 10 csomóponttal rendelkező gyökeres fának, amelynek csomópontjai 1-től 10-ig vannak sorszámozva, az ősektora (szülők tömbje) a következő: $(5, 1, 10, 7, 0, 4, 1, 7, 4, 5)$. Jelöld meg a 7-es csomópont összes leszármazottjainak a számát.

5. Egy 6 csomóponttal rendelkező irányított gráf, amelynek csomópontjai 1-től 6-ig vannak sorszámozva, a mellékelt szomszédsági listával van ábrázolva. Jelöld meg a gráf összes különböző részgráfjainak a számát. Két részgráf különböző, ha a hozzájuk tartozó szomszédsági mátrix különböző.

- a. 2^6 b. 2^8 c. 2^{12} d. 2^{28}

II. TÉTEL

(40 pont)

- Adott a mellékelt algoritmus pszeudokódban.

<p>a. Írd le az algoritmus által kiírt értéket, ha a változókba beolvasott értékek 2 és 3, ebben a sorrendben. (6p.)</p> <p>b. Írj le két értékkészletet az $[1, 10^2]$ intervallumból, amelyeket, ha beolvasunk az algoritmusba, akkor az utolsó kiírt érték 20 lesz. (6p.)</p> <p>c. Írd le az adott algoritmusnak megfelelő C/C++ programot. (10p.)</p> <p>d. Írj az adott algoritmussal megegyező pszeudokód algoritmust, amelyben az egyik minden...végezd el struktúrát egy más ismétlődő struktúra helyettesít. (6p.)</p>	<pre> beolvas n,k (természetes, nem nulla számok) minden i←1,n végezd el minden j←k,1,-1 végezd el kiír j, ' ' ■ minden j←1,k végezd el kiír j, ' ' ■ k←k-1 ■ </pre>
--	--
- Adott a mellékelt **f** alprogram. Írd le mi történik vissza az **f(0)** hívás esetén, hát az **f(23575209)** hívás esetén. (6p.)

	<pre> int f (int n) { if (n!=0) return n%2+f(n/100); return 0; } </pre>
--	---
- Adott egy 9 csomóponttal rendelkező irányítatlan gárf, amelynek csomópontjai 1-től 9-ig vannak sorszámozva, a következő élekkel: [1,2], [1,6], [2,6], [3,7], [4,5], [4,9]. Írd le a gráfba szereplő összefüggő komponensek számát és azt, hogy ezek közül hány rendelkezik maximális csomópontszámmal. (6p.)

III. TÉTEL

(30 pont)

- A **suma** alprogramnak két paramétere van:
 - n**, melyen keresztül egy természetes számot kap a $[0, 10^9]$ intervallumból;
 - s**, melyen keresztül visszaadja a **n** szám különböző, páros számjegyeinek összegét.
 Írd le a teljes alprogramot.
Példa: ha **n=67638825**, a hívás után: **s=16** ($16=6+8+2$), de ha **n=15**, a hívás után: **s=0**. (10p.)
- Egy szó **sufix**-a egy másik szónak, ha az a szó első betűinek elhagyásával származtatható. Írj C/C++ programot, amely a billentyűzetről beolvas két természetes számot **n** és **k** ($n \in [2, 20]$, $k \in [1, n]$) és utána **n** darab különböző, legtöbb 20 karakterből és csak az angol ábécé kisbetűiből álló szavat.
 Az adatok megadásakor, minden szó után Entert kell ütni. Írj programot, amely kiírja a képernyőre egy-egy szóközzel elválasztva azokat a szavakat, melyeknek a **k**. beolvasott szó sufixa, mint ahogy a példában is látható. Ha nincs ilyen szó a beolvasottak között, akkor a **nu exista** üzenet jelenjen meg a képernyőn.
Példa: ha **n=7**, **k=3** és a mellékelt szavakat olvassuk be, akkor a képernyőn a következő jelenik meg: **paratirisi hiritisi**

	<pre> isihast paratirisi isi meremetisire acolisitor hiritisi paraponisit </pre>
--	--
- Nevezzük **10-sorozat**-nak azokat a természetes számokból álló sorozatban megjelenő egymás utáni tagokat, melyek mind 10-nek többszörösei. A sorozat **hossza** a sorozatban szereplő tagok számával egyenlő.
 A **bac.txt** szöveges állomány egy számsorozatot tartalmaz, mely legtöbb 10^6 természetes számot tartalmaz a $[0, 10^9]$ intervallumból, egy-egy szóközzel elválasztva. Legalább egy tag a sorozatban 10-nek többszöröse kell legyen. Írjon ki a képernyőre két értéket, szóközzel elválasztva, reprezentálva a leghosszabb 10-sorozat hosszát az állományban talált sorozatban, illetve az eltérő hosszúságú 10-sorozatok számát. Tervezzon egy memória felhasználás és a futási idő szempontjából hatékony algoritmust.
Példa: ha az állomás a mellékelt sorozatot

	7 3 <u>200 100 10</u> 9 6 <u>100 1000 40</u> 1002 <u>20 30</u>
--	--

 tartalmazza, akkor a kiírt értékek 3 2
 - Írd meg a megtervezett algoritmusnak megfelelő C/C++ programot. (2p.)
 - Írd le saját szavaiddal a megtervezett algoritmust és indokold meg ennek a hatékonyságát. (8p.)