



Examenul național de bacalaureat 2022  
Proba E. d) INFORMATICĂ Limbajul C/C++

SIMULARE

**Limbajul C/C++ Filieră teoretică, profil real, specializare matematică-informatică /  
matematică-informatică intensiv informatică**

- *Minden tétel kötelező. Hivatalból 10 pont jár.*
- *A teljes munkaidő 3 óra.*
- *A kért megoldásokban a használt azonosítóknak teljesíteniük kell a megkötéseket (bold), vagy a megkötések hiányában az azonosítóknak a jelentésüknek megfelelő elnevezést kell adni (esetleg rövidített formában). A bemeneti adatokat helyeseknek kell tekinteni, nincs szükség az ellenőrzésükre.*
- *A gráfok esetében, bármely él végpontjai különböznek egymástól és bármely két él legalább egy végpontban különbözik egymástól.*

**I. TÉTEL**

**(20 pont)**

**Az első 5 feladat esetében írjátok a vizsgalapra a helyes válasz betűjelét. Minden helyes válasz 4 pontot ér.**

1. A feltétel, hogy az **x** és **y** egész számok ne egyszerre legyenek 0:

- a. **`x != 0 && y != 0`**      b. **`!(x==0 && y==0)`**      c. **`x + y != 0`**      d. **`x * y != 0`**

2. Adott a mellékelt szomszédossági listával meghatározott irányítatlan gráf. Határozzátok meg azt a legkisebb csomópontszámot, amelyet letörölve a gráfból, a kapott gráf euler-gráf lesz.

- a. 0      b. 4      c. 2      d. 3

1: 2, 3, 4  
2: 1, 3, 5  
3: 1, 2, 4  
4: 1, 3  
5: 2, 6  
6: 5

3. A backtracking módszer alkalmazásával létrehozzuk ABC sorrendben a **mouse** szó összes anagrammáját, amelyben nincs 2 magánhangzó egymás mellett. Határozzátok meg, hány anagramma lesz a **esumo** és **osemu** megoldások között.

- a. 3      b. 0      c. 2      d. 1

4. Adott az alábbi **f** alprogram. Melyik meghívásra lesz az alprogram által visszatérített érték **8**?

- a. **`f(10)`**      b. **`f(34)`**  
c. **`f(128)`**      d. **`f(256)`**

```
int f(int n)
{
    if(n==1) return 0;
    if(n%2==0) return 1+f(n/2);
    else return 1+f(3*n+1);
}
```

5. Egy 6 csomópontból álló gyökeres fa csúcsait 1-től 6-ig számozzuk, az apák tömbje pedig  $t=\{x, y, 4, z, 0, 1\}$ . Milyen értékei lehetnek a tömb **x**, **y**, **z** értékeinek úgy, hogy a fának egyetlen levele legyen?

- a. 2, 3, 6      b. 1, 3, 4      c. 2, 3, 5      d. 2, 5, 3



## II. TÉTEL

(40 pont)

1. Adott a mellékelt algoritmus pszeudokódban. Az  $x \% y$  az  $x$  és  $y$  egész szám osztási maradékát, az  $[x]$  az  $x$  valós szám egész részét jelöli.

a) Mit ír ki az algoritmus, ha  $a = 2$  és  $n = 7$ ? (6 pont)

b) Írjatok 2-2 különböző értéket, amit be lehet olvasni az  $a$  és  $n$  változóba úgy, hogy a kiírt érték 1 legyen. (6 pont)

c) Írjatok meg az algoritmusnal megfelelő C/C++ programot. (10 pont)

d) Írjatok egy, az adott algoritmussal egyenértékű, algoritmust, amelyben egy más ismétlődő struktúrát használtok. (6 pont)

```
beolvas a, n
(a egész szám, n természetes szám)
p ← 1
amедdig n>0 végezd el
    ha n%2=0 akkor
        a ← a * a
        n ← [n/2]
    különben
        p ← p * a
        n ← n-1
kiír p
```

2. Helyettesítsétek a pontozott részt egy programrészszel úgy, hogy az létehozzon a memóriában egy  $2*n$  sorból és  $2*n$  oszlopból álló kétdimenziós tömböt, amelynek elemei természetes számok. A tömb elemei 0-tól indexálódnak és szimmetrikusak lesznek a vízszintes és függőleges középvonalhoz viszonyítva. (6 pont)

```
for(int i=0;i<n;i++)
    for(int j=0;j<n;j++)
        .....
```

|   |   |   |   |   |   |
|---|---|---|---|---|---|
| 0 | 1 | 2 | 2 | 1 | 0 |
| 1 | 2 | 3 | 3 | 2 | 1 |
| 2 | 3 | 4 | 4 | 3 | 2 |
| 2 | 3 | 4 | 4 | 3 | 2 |
| 1 | 2 | 3 | 3 | 2 | 1 |
| 0 | 1 | 2 | 2 | 1 | 0 |

Az adott példában, ha  $n=3$ , a tömb elemei a mellékelt ábrán láthatók:

3. Adottak az alábbi deklarációk, ahol az  $e$  változó egy diák családnevét, keresztnévét és születési dátumát tárolja. Írjatok egy C/C++ programrészt amely kiírja a diák nevét a következő formátumban **i.nume** (a keresztnév kezdőbetűje és a családnév egy ponttal elválasztva), ha a születési év nagyobb mint 2001, illetve a születési dátumot **zi.luna.an** formátumban (pontokkal elválasztva) ellenkező esetben. (6 pont)

```
struct data {
    int zi, luna, an;
};
struct elev{
    char nume[21], pren[21];
    data dn;
}e;
```



### III. TÉTEL

(30 pont)

1. Bármely **páros természetes szám**,  $n > 2$  felírható 2 prím természetes szám összegeként (*Goldbach tétele*). Írjatok egy **C/C++** alprogramot amely egy **n páros természetes számot** kap paraméterként az  $[4, 10^9]$  intervallumból és visszatéríti az **x** és **y** prím számokat, amelyek összegeként felírható az **n** szám.

**Példa.** Ha  $n=290$  egy lehetséges megoldás  $n = 7+283$  (a két visszatérített érték 7 és 283), ha pedig  $n=100$  egy lehetőség (nem az egyetlen)  $n = 3+97$  (a két visszatérített érték 3 és 97).  
(10 pont)

2. Két **A** és **B** szó rímel ha az **A** szó utolsó magánhangzójával kezdődő szóvég megegyezik a **B** szó utolsó magánhangzójával kezdődő szóvéggel. Például, a **savant** és **gigant** rímel, de a **sport** és **cert** nem rímel.

Írjatok egy **C/C++** programot amely beolvas a billentyűzetről egy legtöbb 255 karakterből (kisbetűk és szóközők) álló karakterláncot és kiírja a képernyőre az egymással rímelő szópárosokat. Minden szó hossza minimálisan 3 lesz és tartalmaz legalább egy magánhangzót. A program a szópárosokat a képernyő különböző soraiba fogja kiírni, vagy a **NU EXISTA** üzenetet, ha a beolvasott karakterlánc nem tartalmaz egymással rímelő szavakat.  
(10 pont)

Például, ha a beolvasott karakterlánc

**true is bool and school is cool**

a következő szópárosokat fogjuk kiírni:

**bool school**

**bool cool**

**school cool**

3. Egy **n** természetes szám **p-összetett**, ha felírható **p** egymásutáni természetes szám összegeként. Írjatok egy **C/C++** programot amely a **BAC.TXT** állományból beolvas legtöbb 1000000 nullától különböző, legtöbb 2 számjegyű természetes számot és kiírja a képernyőre a legnagyobb **p-összetett** számot, amelyet az állományban található különböző számok összegeként lehet felírni, illetve az összeget alkotó tagok számát. A két kiírt számot egymástól szóközzel választjuk el.

**Példa:** ha a **BAC.TXT** állomány az alábbi értékeket tartalmazza:

12 1 2 7 3 4 7 6 8 10

Akkor a képernyőre kiírt értékek: **21 3**

a) Tervezzetek egy idő és memória hatékony algoritmust és magyarázzátok el a saját szavaitokkal, miben rejlik a választott algoritmus hatékonysága.  
(2 pont)

b) Írjatok meg az a) pontnál vázolt algoritmusnak megfelelő **C/C++** programot.  
(8 pont)