

Examenul național de bacalaureat 2026

Proba E. d)

INFORMATICĂ Limbajul C/C++ (Simulare județeană)

Varianta 2

Filiera teoretică, profil real, specializare științe ale naturii

- Toate subiectele sunt obligatorii. Se acordă zece puncte din oficiu.
- Timpul de lucru efectiv este de trei 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.

I. TÉTEL (20 pont)

Az 1-től 5-ig számozott itemek esetén, írja a vizsgalapra a helyes válasznak megfelelő betűt. Minden helyes válasz 4 pontot ér.

- Jelölje meg azt a kifejezést, amelynek értéke a legkisebb a többi kifejezéshez képest:
a. $25 \cdot 2026 / (5 \cdot 5)$ b. $25 / 5 \cdot 2026$ c. $(25 \cdot 2026) / 5$ d. $(25 \cdot 2026) / 5 \cdot 5$
- Adottak **A** és **B** egydimenziót tömbök elemei: **A** = (1, 3, 4, 6, 20, 45), és **B** = (74, 63, 45, 44, 26, 10, 2, 1). Miután csökkenő sorrendben összefésültük őket a következő tömböt kapjuk:
a. (1, 3, 4, 6, 20, 45, 74, 63, 45, 44, 26, 10, 2, 1) b. (74, 63, 45, 45, 44, 26, 20, 10, 6, 4, 3, 2, 1, 1)
c. (74, 63, 45, 44, 26, 20, 10, 6, 4, 3, 2, 1) d. (74, 63, 45, 44, 26, 10, 2, 1, 1, 3, 4, 6, 20, 45)
- Adja meg azt a C/C++ kifejezést, amelyik egyenértékű ezzel: $!(x \geq y \ \&\& \ x < z) \ || \ (x > z)$
a. $x < y \ || \ x > z \ \&\& \ x < z$ b. $x < y \ \&\& \ x > z \ \&\& \ x < z$ c. $x < y \ || \ x > z \ || \ x > z$ d. $x < y \ || \ x > z \ \&\& \ x > z$
- Egy egydimenziós tömb növekvő sorrendbe rendezéséhez a buborékos rendezés algoritmusát használjuk. Ha a kezdeti tömb (47, 23, 12, 17, 11, 19), akkor adja meg a tömböt a második végigjárás után:
a. (23, 12, 17, 11, 19, 47) b. (12, 11, 17, 19, 23, 47)
c. (12, 17, 11, 19, 23, 47) d. (11, 12, 17, 19, 23, 47)
- A mellékelt C/C++ programrészletben minden változó egész típusú. Adja meg azokat a kifejezéseket, amelyeket a pontozott részekre behelyettesítve az így kapott programrészlet végrehajtása után a következő eredmény jelenjen meg: 3 1 5 2

- $p=1; p++;$
- $p=0; p++;$
- $p=0; p--;$
- $p=1; p+=2;$

```
x=75; f=2;
while(x!=1)
{
    .....
    while(x%f==0)
    {
        x/=f;
        .....
    }
    if(p>0) cout<<f<<" "<<p<<" ";
    f++;
}
```

II. TÉTEL (40 pont)

1. Adott a mellékelt algoritmus pszeudokódban.

Az **a** természetes szám és a **b** természetes szám osztási maradékát **a%b**-vel, az egész hányadosát pedig **[a/b]**-vel jelöltük.

- Adja meg mi jelenik meg az algoritmus végrehajtása után, ha a beolvasott számok, ebben a sorrendben vannak:
6, 1020, 135, 249, 5678, 9, 64. (6p.)
- Ha az **n** változóba beolvasott érték 5, adja meg az összes különböző számot a $[10^2, 10^4]$ intervallumból, amelyeket az **a** változóba beolvasva az algoritmus, végrehajtása után, két egyenlő értéket ír ki. (6p.)
- Írja meg az adott algoritmusnak megfelelő C/C++ programot. (10p.)
- Írjon az adott algoritmussal egyenértékű pszeudokód algoritmust, amelyben a **minden...végezd el** szerkezetet egy előltesztelő ismétlő szerkezettel helyettesíti. (6p.)

```
beolvas n, a
(nemnulla természetes számok)
n1 ← 0; n2 ← 0
minden i ← 2, n végezd el
    beolvas b (természetes szám)
    c ← b
    ha b > 9 akkor
        ismételd
        b ← [b/10]
    ameddig b < 10
        ha a%2 = b%2 akkor
            n1 ← n1 + 1
        különben
            n2 ← n2 + 1
    a ← c
kiír n1, " ", n2
```

2. Az **x** és **y** változók természetes számokat tárolnak az $[1, 10^4]$ intervallumból. Írjon egy C/C++ utasítássorozatot, amelynek végrehajtása után a képernyőn megjelenik a nagyobb érték, ha mindkét érték páros, a kisebbik érték, ha **x** és **y** is páratlanok vagy a **c**, ha a két érték különböző paritású (az egyik szám páros, a másik szám páratlan).
Példa: ha **x=24**, **y=2026** a képernyőre kiírt szám **2026**, ha **x=25**, **y=127** a kiírt szám **25**, ha pedig **x=24**, **y=127** akkor a kiírt érték **0**. **(6p.)**

3. A mellékelt programrészletben az **x** és **y** változók típusa **char**, az **i** pedig **int** típusú. Adja meg, mi jelenik meg a megadott programrészlet végrehajtása után. **(6p.)**
- ```
x='a'; y='C';
for(i=1; i<=3; i++)
{ cout<<x<<y; | printf("%c%c",x,y);
 x=x+1;
 y=y-1;
}
```

### III. TÉTEL

**(30 pont)**

1. Olvassa be az **n** természetes számot. Határozza meg a szám azon számjegyeinek összegét, amelyek paritása megegyezik a rangjukkal. A számjegyek rangjait jobbról balra számozzuk, **0**-tól kezdve. Ha a számnak nincsenek olyan számjegyei, amelyek megfelelnek a kért tulajdonságnak, akkor a "**nu există**" üzenet jelenjen meg.  
Írja meg pszeudokódban a feladat megoldásának algoritmusát.  
**Példa:** ha **n=683420** a kiírt érték **15**. (a kért tulajdonsággal rendelkező számjegyek: **0, 4, 3** és **8**), ha pedig **n=325** akkor megjelenik a "**nu există**" üzenet. **(10p.)**
2. Írjon egy C/C++ programot, amelyik a billentyűzetről beolvas három természetes számot, **n**, **p1** és **p2** ( $n \in [3, 20]$ ,  $p1, p2 \in [2, n]$ ,  $p1 \leq p2$ ), és felépít a memóriában egy **n** elemű egydimenziós tömböt a következő tulajdonságokkal:
- a **p1** és **p2** pozíciók közötti (beleértve a **p1** és **p2** pozíciókat) értékek egyenlőek **0**-val.
  - a tömb az első pozíciótól kezdve a **(p1-1)**-edik pozícióig (balról jobbra haladva) az első **(p1-1)** egymást követő páros, nullától különböző számok szigorúan növekvő sorozatát, a **(p2+1)**-edik pozíciótól kezdve pedig az utolsó pozícióig (balról jobbra haladva) az első **(n-p2)** egymást követő páratlan szám szigorúan növekvő sorozatát tartalmazza.
- A program a létrehozott tömb elemeit egy szóközzel elválasztva jelenítse meg a képernyőn.  
**Példa:** ha **n=10**, **p1=4** és **p2=6** akkor a kapott tömb **(2, 4, 6, 0, 0, 0, 1, 3, 5, 9)**. **(10p.)**
3. A **BAC.TXT** állomány legalább kettő és legtöbb **10000** természetes számot tartalmaz növekvő sorrendben. A számokat egymástól egy szóköz választja el és mindegyik legtöbb **9** számjegyből áll. Az állományban legalább az egyik szám páratlan.
- a) Írjon egy C/C++ programot, amelyik beolvassa az állományból az összes számot majd, egy futási idő és memóriahasználat szempontjából is hatékony algoritmus használatával határozza meg és írja ki a képernyőre szigorúan növekvő sorrendben, egymástól egy szóközzel elválasztva az összes páratlan számot az állományból. Minden szám csak egyszer jelenjen meg. **(8p.)**  
**Példa:** ha az állomány tartalma a következő **1 1 2 2 2 7 10 10 19 19 19 19 20 20 25** akkor a képernyőn megjelenő számok, ebben a sorrendben: **1 7 19 25**.
- b) Írja le saját szavaival az a) alpontnál használt algoritmust és indokolja annak hatékonyságát. **(2p.)**