

Examenul național de bacalaureat 2025

Proba E. d)  
INFORMATICĂ  
Limbajul C/C++

Simulare

Filieră 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 ítemek esetén, írja a vizsgalapra a helyes válasznak megfelelő betűt. Minden helyes válasz 4 pontot ér.

1. Adja meg a mellékelt C/C++ kifejezés értékét.  $20.25/2.5*100$
- a. 0                                      b. 0.081                                      c. 810                                      d. 1000
2. Adottak egy egydimenziós tömb elemei, ebben a sorrendben: (2, 3, 5, 20, 25, 26, 45). Annak ellenőrzésére, hogy a tömbben létezik az  $x=25$  értékű elem, a bináris keresés módszerét használjuk. Jelölje meg azt az értéksort, amelynek értékei az  $x$  értékéhez hasonlítódnak az említett módszer alkalmazása során.
- a. 45, 26, 25                                      b. 20, 26, 25                                      c. 3, 5, 25                                      d. 2, 5, 25
3. Adja meg a mellékelt C/C++ kifejezés értékét.  $\text{floor}(-20.25)$
- a. -21                                      b. -20                                      c. 20.25                                      d. 25
4. A mellékelt C/C++ kódrészletben minden változó egész. Adjon meg egy kifejezést, amely a pontozott részt helyettesítheti úgy, hogy kapott kód végrehajtása nyomán az  $x$  változó értéke 25 legyen, ami a 1000 és 2025 számok legnagyobb közös osztója.
- ```
x=1000; y=2025;
while (y!=0)
{ r=...;
  x=y; y=r;
}
```
- a.  $x\%y$                                       b.  $(x+y)\%2$                                       c.  $x\%2025$                                       d.  $1000\%y$
5. Az  $x$  és  $y$  változók egészek. Adjon meg egy C/C++ utasításort, amely végrehajtása nyomán, az  $x$  változóba a  $2025$  érték kerül, az  $5 \cdot (5 \cdot (5 \cdot 5 + 55) + 5)$  számítások eredményként.
- a. `x=1; y=11; while(y>0) { x=5*x+y; y=y/2; }`                                      b. `x=1; y=55; while(y>0) { x=5*x+y; y=y/10; }`                                      c. `x=5; y=11; while(y>0) { y=y/2; x=5*x+y; }`                                      d. `x=5; y=555; while(y>0) { y=y/10; x=5*x+y; }`

II. TÉTEL (40 pont)

1. Adott a mellékelt algoritmus pszeudokódban.
- $a\%b$  jelöli az  $a$  természetes számnak a  $b$  nem nulla természetes számmal való osztási maradékát
- a. Írja le, hogy mit ír ki az algoritmus a végrehajtása során, ha beolvassuk, ebben a sorrendben, a 20 és 25 számokat. (6p.)
- b. Ha az  $m$ -be olvassuk az 5 számot, írjon le két számot az  $[1, 50]$  intervallumból, amelyeket az  $n$ -be olvashatunk úgy, hogy az algoritmus végrehajtása során, mindkét esetben az utolsó kiírt számérték 2 legyen. (6p.)
- c. Írja meg az algoritmusnak megfelelő C/C++ programot. (10p.)
- d. Írjon az adott algoritmussal egyenértékű pszeudokód algoritmust, amelyben a **minden...végezd el** szerkezetet megfelelőképpen helyettesíti egy **amíg...végezd el** ismétlődő szerkezettel. (6p.)
- ```
beolvas m,n
      (természetes számok)
ha m<n akkor
  n←n+m
  m←n-m
  n←n-m
■
k←m
minden i←m,n,-1 végezd el
  kiír k, ' '
  ha i%2=0 akkor
    k←k-1
    kiír '*'
  ■
  k←k-1
■
```

2. Az **A** és **B** egydimenziós tömbök elemei **A**=(3,20,25,26,2025) és **B**=(**x**,**y**,**z**). Az elemek meghatározásához a tömbök növekvő sorrendbe való összefésülésekor, az **x** az **A** két értékével, az **y** az **A** egy értékével, míg a **z** az **A** négy értékével hasonlítódik össze. Írjon le a **B** tömbre egy lehetséges értéksort, abban a sorrendben, amelyben a tömbben szerepelnek. (6p.)
3. Egy cég két IT szakemberének ismert a születési éve és az alkalmazás éve.  
Az **sn1** és **sa1** egész változók az első szakember születési és alkalmazási évét tárolják, míg az **sn2** és **sa2** egész változók, a második szakember születési és alkalmazási évét tárolják. Írjon egy C/C++ utasítássort, amelynek futtatásakor a képernyőre kerül az alkalmazás éve, ha mindkét szakembert ugyanabban az évben alkalmazták, ellenkező esetben a korábban alkalmazott szakember adatai: születési és alkalmazási év, egy szóközzel elválasztva.  
**Példa:** Az **sn1**=1990-re és az **sa1**=2020-re, ha **sn2**=2000 és **sa2**=2020 a kiírt érték 2020 és ha **sn2**=2000 és **sa2**=2025 a kiírt értékek 2000 2025 (6p.)

### III. TÉTEL

(30 pont)

1. Egy „örömet hozó” szám - **harsad** (vagy Niven szám), egy a saját számjegyeinek összegével osztható egész szám.  
Beolvasunk egy **k** ( $k \geq 1$ ) természetes számot, és írassuk ki a legnagyobb természetes harsad számot, amely kisebb vagy egyenlő a **k**-val. Írja le pszeudokódban, az adott feladat megoldási algoritmusát.  
**Példa:** ha **k**=2027, a kiírt érték 2025 ( $2+0+2+5=9$ , és a 2025 osztható 9-cel). (10p.)
2. Két nemnulla természetes számot **asemenea**-nak nevezzük, ha különbözőek és ha az 5 ugyanazon a hatványon jelenik meg mindkettő törzstényezőre bontásában.  
Írjon egy C/C++ programot, amely a billentyűzetről beolvas egy **n** ( $n \in [1, 10^2]$ ) természetes számot, majd egy **n** tagú, természetes számokból álló számsort az  $[1, 10^6]$  intervallumból, egy egydimenziós tömb elemeit. A program írjon ki a képernyőre, egy-egy szóközzel elválasztva, minden olyan számot a számsorból, amelyek **asemenea**-k az utolsó beolvasott számmal, vagy a **nu exista** üzenetet, ha nem létezik ilyen szám.  
**Példa:** ha **n**=9, a tömb (20, 875, 100, 250, 12347, 5, 500, 625, 250), a képernyőre került számok, nem feltétlenül ebben a sorrendben: 875 500 (10p.)
3. Egy divatbemutatóra ékszerkészleteket készítenek elő, egy készlet tartalmaz egy fülbevalót és egy nyakláncot, legalább két drága- vagy féldrágakővel. Kilencféle követ használnak, amelyeket 1-től 9-ig számoznak és minden ékszernek van egy címkéje, egy természetes szám, amelyben minden számjegy egy beépített kőnek felel meg, a modellen belüli fontosságának csökkenő sorrendjében. Egy **ékszerkészlet potrivit (találó)**, ha a készlet ékszereinek legfontosabb köve azonos típusú.  
A **bijuterii.in** állomány természetes számokat tartalmaz a  $[10, 999]$  intervallumból: az első sorban két számot, az **nc**-t és **np**-t, amelyek a rendelkezésre álló fülbevalók, illetve a nyakláncok számát jelentik, a második sorban egy **nc** elemszámú számsort, a fülbevalók címkéit, és a harmadik sorban egy **np** elemszámú számsort, a nyakláncok címkéit. Az azonos sorban levő számok egy-egy szóközzel vannak elválasztva. A képernyőre a **DA** üzenet kerüljön, ha egy potrivit tulajdonságú ékszerkészlet kialakítható vagy a **NU** üzenetet, ellenkező esetben. Tervezzon a végrehajtási idő szempontjából hatékony algoritmust.  
**Példa:** ha az állomány a mellékelt számokat tartalmazza a **DA** üzenet lesz kiírva.  

10	11
497 125 521 497 513 258 491 55 551 16	
27 259 943 77 945 57 52 552 16 17 71	

  
(a 18 potrivit tulajdonságú készletből, kettőt a 258 címkéjű fülbevalóból képezhetünk, a 259 és 27 címkéjű nyakláncokkal szerre, mindezeknek az ékszereknek a legfontosabb köve a 2-es típusú).  
a. Írja le saját szavaival a használt algoritmust és indokolja annak hatékonyságát. (2p.)  
b. Írja meg a leírt algoritmusnak megfelelő C/C++ programot. (8p.)