

Examenul național de bacalaureat 2024

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

Simulare

Filieră teoretică, profil real, specializare matematică-informatică / matematică-informatică intensiv informatică
Filieră vocațională, profil militar, specializare matematică-informatică

- 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.
- În grafurile din cerințe oricare arc/muchie are extremități distincte și oricare două arce/muchii diferă prin cel puțin una dintre extremități.

I T É T E L (20 pont)

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

1. Jelölje meg a mellékelt C/C++ kifejezés értékét. 6.3/20+24
- a. 0 b. 24 c. 24.315 d. 24.9
2. A mellékelt módon értelmezett az **f** alprogram. Adja meg, hogy mi kerül kiírásra az alábbi meghívás nyomán.
f(2024) ;
- ```
void f(int x)
{ if(x>=24) if(x%2==0)
 { cout<<2; | printf("2");
 f(x/2);
 }
 else
 { f(x/2);
 cout<<4; | printf("4");
 }
 else cout<<0; | printf("0");
}
```
- a. 04424222                      b. 22220444                      c. 22242440                      d. 44402222
3. A backtracking módszert használva előállítjuk az összes lehetséges változatot, ahogy négy személy kifizethet egy **24** lejes összefogyasztást, a következő feltételek mellett:
- mindenki fizet egy zérótól eltérő, **10**-nél szigorúan kisebb összeget;
  - az első kevesebbet fizet, mint a második, a második kevesebbet, mint a harmadik, és ez is kevesebbet, mint a negyedik.
- Egy megoldás négy értéket jelent, amelyek sorrendben a négy személy által fizetett összegeknek felelnek meg. Az első négy előállított megoldás, sorrendben: (1, 6, 8, 9), (2, 5, 8, 9), (2, 6, 7, 9), (3, 4, 8, 9). Adja meg közvetlenül a (3, 6, 7, 8) után következő megoldást.
- a. (3, 5, 7, 9)                      b. (4, 5, 6, 9)                      c. (4, 5, 7, 8)                      d. (5, 5, 6, 8)
4. Az alábbi értelmezésben a struktúra mezői egy videójátékkal kapcsolatos adatokat tárolnak: kategóriát, nehézségi szintek számát (legfeljebb öt) és a szintenként elérhető maximális pontszámot.
- Adja meg azt a C/C++ kifejezést, amelynek értéke akkor és csakis **1**, ha a **2**-essel jelölt szinten elérhető maximális pontszám szigorúan nagyobb **4**-nél.
- ```
struct joc
{ int categorie, nrNiveluri;
  int puncte[5];
}g;
```
- a. **joc(puncte[2])>4** b. **puncte.joc[2]>4** c. **puncte[2].g>4** d. **g.puncte[2]>4**
5. Egy városban **5** park van, jelölésük **A**, **B**, **C**, **D** és **E**, biciklissávok kötik össze őket, ezek egyirányúak a következőképpen: egy sáv az **A** parkból a **B** parkba, egy sáv a **B** parkból az **E** parkba, egy sáv a **C** parkból a **D** parkba, egy sáv a **D** parkból a **C** parkba, egy sáv az **E** parkból az **A** parkba, egy sáv az **E** parkból a **C** parkba és egy sáv az **E** parkból a **D** parkba. Adja meg a minimális számú sávot, amelyek esetében a forgalom irányát szükséges megfordítani ahhoz, hogy az így kapott irányított gráf, amelyeknek csúcsai a parkok és az élei a sávok, erősen összefüggő legyen.
- a. 1 b. 2 c. 4 d. 6

II TÉTEL

(40 punct)

1. Adott a mellékelt algoritmus pszeudokódban.

Az $a \div b$ az a természetes számnak a b nullától különböző természetes számmal való osztási maradékát és $[c]$ a c valós szám egész részét jelöli.

- Adja meg az algoritmus végrehajtása során kiírt értéket, ha a beolvasott szám 32625. (6p.)
- Írjon két számot a $[10, 100]$ intervallumból, amelyeket az n változóba beolvashatunk úgy, hogy mindkettőre az algoritmus végrehajtása után a 2 érték kerüljön kiírásra. (6p.)
- Írja le az algoritmusnak megfelelő C/C++ programot. (10p.)
- Írjon pszeudokódban a megadottal egyenértékű algoritmust, a **második** **amíg ... végezd el** struktúrát megfelelőképpen helyettesítve egy hátul tesztelő ismétlő struktúrával. (6p.)

```
beolvas n
(nem nulla természetes szám)
s ← 0; d ← 2
amíg d * d ≤ n végezd el
    p ← 0
    amíg n % d = 0 végezd el
        n ← [n/d]; p ← 1
    s ← s + d * p; d ← d + 1
ha n ≠ 1 akkor
    s ← s + n
kiír s
```

- Egy 10 csúcsú fát, 1-től 10-ig sorszámozott csúcsokkal, a (8, 1, 4, 5, 0, 9, 4, 5, 4, 3) ösvektor ad meg. Sorolja fel a fa minden csúcsát, amelynek egyetlen direkt leszármazottja (fia) van. (6p.)
- Az i és j változók egész típusúak, míg az a változó egy kétdimenziós tömböt tárol, 5 sorral és 5 oszloppal, amelyeket 0-tól számozunk, és eredetileg minden eleme egyenlő 1-gyel.

Írja le az alábbi C/C++ utasítássort, helyettesítve a kipontozott részt megfelelő utasításokkal, közöttük **legtöbb négy értékadással** úgy, hogy a kapott utasítássor végrehajtása után az a változó a mellékelt tömböt tartalmazza.

```
for (i=0; i<5; i++)
    for (j=0; j<5; j++)
```

```
0 0 0 0 0
0 2 2 2 2
0 2 4 4 4
0 2 4 6 6
0 2 4 6 8
```

(6p.)

III TÉTEL

(30 punct)

- Az **Impare** alprogram egy n paraméterrel rendelkezik, amelyen egy természetes értéket kap ($n \in [1, 10^4]$). Az alprogram visszatéríti a számot, amit az n páratlan számjegyeinek megduplázásával kap vagy -1-et, ha nincs egyetlen páratlan számjegy sem. Írja le a teljes alprogramot. (10p.)

Példa: ha $n=3361$, az alprogram által visszatérített érték 3333611.

- Egy **sablon** egy szöveg, amelyben a szavak egy-egy szóközzel vannak elválasztva és csak az angol ábécé kis- és nagybetűiből vagy csak * karakterekből állnak, ez utóbbiakat **generikus szavaknak** nevezzük. A szó hossza egyenlő az öt felépítő karakterek számával.

Egy számítógép egy ilyen sablon alapján állít elő egy mondatot, minden generikus szavat egy azonos hosszúságú szóval helyettesítve egy megadott listából.

Írjon egy C/C++ programot, amely a billentyűzetről beolvas egy n ($n \in [1, 100]$) természetes számot, és egy n szavas listát, amelyet egy megadott típusú sablon követ. A lista minden szava az angol ábécé maximum 10 kis- és nagybetűjéből áll és beolvasáskor külön sorba adjuk meg őket. A sablon maximum 100 karaktert tartalmaz. A program előállít a memóriában, majd kiír egyet a mondatokból, amit a sablon és a beolvasott lista alapján generálhatunk vagy az **imposibil** üzenetet, ha nem lehetséges ilyen mondatot előállítani.

Példa: ha $n=6$, és a szólista a mellékelt,

ha a sablon	Era o vreme	*****	si	*****	din belsug	*****
az előállított mondat	Era o vreme	placuta	si	soare	din belsug	soare
vagy	Era o vreme	calduta	si	soare	din belsug	acasa
stb., és ha a sablon	*** o vreme	*****	si	*****		

```
rece
placuta
acasa
calduta
innorata
soare
```

(10p.)

- Egy sort **api** típusúnak nevezünk, ha minden tag előfordulásának száma kisebb vagy egyenlő az adott tagnál, és a paritása is azonos a tagével.

A **bac.in** állomány legfeljebb 10^6 természetes számot tartalmaz a $[1, 10^3]$ intervallumból, egy-egy szóközzel elválasztva. A képernyőre írassa ki a **DA** üzenetet, ha a sor **api** típusú, vagy a **NU** üzenetet ellenkező esetben. Tervezzen egy a futási idő szempontjából hatékony algoritmust.

Példa: ha az állomány tartalma 6 27 2 6 27 6 6 14 14 2 27 a képernyőre kiíródik a **DA**.

(a páros 6 tag 4-szer jelenik meg, a 4 szintén páros és $4 \leq 6$, a 2 és 14 páros tagok 2-szer jelennek meg, a 2 is páros és $2 \leq 2$, illetve $2 \leq 14$, míg a páratlan 27 tag 3-szor fordul elő, a 3 szintén páratlan és $3 \leq 27$).

- Írja le saját szavaival a használt algoritmust, és indokolja annak hatékonyságát. (2p.)

- Írja meg a tervezett algoritmusnak megfelelő C/C++ programot. (8p.)