Examenul de bacalaureat naţional 2020
Proba E. d)
Informatică
Limbajul C/C++

Testul 1

Filieră teoretică, profil real, specializare științe ale naturii

- **Toate subiectele sunt obligatorii. Se acordă 10 puncte din oficiu.**
- **Timpul de lucru efectiv este de 3 ore.**
- **Identificatorii utilizaţi în rezolvări trebuie să respecte precizările din enunţ (bold), iar în lipsa unor precizări explicite, notaţiile trebuie să corespundă cu semnificaţiile asociate acestora (eventual în formă prescurtată). Datele de intrare se consideră corecte, validarea lor nefiind necesară.**

I. TÉTEL (20 pont)

Minden item esetében 1-től 5-ig írja a vizsgalapra a helyes válasznak megfelelő betűt. Minden helyes válasz 4 pontot ér.

1. Adja meg azt a **C/C++** kifejezést, amelynek az értéke **1** akkor és csakis akkor, ha az **n** egész változó értéke osztható **2**-vel és **5**-el.

a. `!(n%2==1 || n%5!=0)` **b.** `!(n/2==1 && n/5!=0)`

c. `n%2==0 || !(n%5==0)` **d.** `n/2==0 && !(n/5==0)`

2. Annak ellenőrzésére, hogy az **x=18** érték megtalálható-e a **(4,7,9,15,16,20,27)** elemeket tartalmazó egydimenziós tömbben a bináris keresés módszerét alkalmazzuk. A módszer alkalmazása során az **x**-el összehasonlított elemek rendre:

a. `4,16,20` **b.** `15,20,16` **c.** `15,16,20` **d.** `4,9,16,27`

3. Adottak az **A** és **B** egydimenziós tömbök értékei rendre **A=(20,16,10,7,2)** és **B=(56,15,14,11,1)**. A két tömb csökkenő sorrendben való összefésülésének eredménye:

a. `(20,15,10,7,1)` **b.** `(56,20,16,15,14,10,11,7,2,1)`

c. `(56,15,14,11,1,20,16,10,7,2)` **d.** `(56,20,16,15,14,11,10,7,2,1)`

4. Az a **C/C++** kifejezés, amelynek értéke 1:

a. `ceil(20.20)-1==floor(19.20)` **b.** `ceil(20.19)==floor(20.20)`

c. `ceil(20.20)==ceil(19.20)` **d.** `floor(19.20)==floor(20.20)`

5. A mellékelt utasítás sorozatban miden változó egész típusú. Az a kifejezés, amelyet a pontozott helyre beírva, az utasítássorozat elvégzése után az **m** változó értéke egyenlő lesz a **2020** és **1020** legnagyobb közös osztójával:

```
m=2020; n=1020;
while (...)
{ x=m%n; m=n; n=x;}
```

a. `m!=0` **b.** `m%n!=0` **c.** `n!=0` **d.** `n!=m`

II. TÉTEL (40 pont)

1. **Adott a mellékelt algoritmus pszeudokódban.**
Az **a%b** az **a** természetes számnak, **b** nullától különböző természetes számmal való osztási maradékát jelöli, valamint **[c]** a **c** valós szám egész részét.

a. Írja le az algoritmus által kiírt értéket, ha a beolvasott számok **12345, 780, 921, 4013, 75, 100214** a megadott sorrendben. **(6p.)**

b. Ha az **n** változóba beolvasott szám **49**, írjon két olyan adatsort, amelyeket, ha a továbbiakban beolvasunk mindkét esetben az algoritmus elvégzése után a kiírt érék **49** legyen. **(6p.)**

c. Írja meg az adott algoritmusnak megfelelő **C/C++** programot. **(10p.)**

d. Írjon az algoritmussal egyenértékű pszeudokód algoritmust, amely csak egy ismétlő szerkezetet tartalmaz. **(6p.)**

```
olvas n (természetes szám)
p←1; m←0; k←0;
┌amíg n≠0 végezd el
│ olvas x (természetes szám)
│ ┌minden i←1,k végezd el
│ │ x←[x/10]
│ └■
│ ┌ha x≠0 akkor c←x%10
│ │ellenben c←n%10
│ └■
│ m←c*p+m
│ n←[n/10]
│ p←p*10; k←k+1
└■
kiír m
```

2. Az alábbi változók egy háromszög valós koordinátáit tárolják az **xOy** koordináta rendszerben. A **tAx** és **tAy** az **A** pont, a **tBx** és **tBy** az **B** pont, a **tCx** és **tCy** az **C** pont, abszcisszája illetve ordinátája. Deklarálja megfelelő módon a változókat, majd írjon egy utasítássorozatot, amelynek eredményeként a képernyőre kiíródik a háromszög **AB** oldalhosszának a négyzete. **(6p.)**

3. A mellékelt utasítássorozatban az **a** karakter típusú, míg az **i** és **k** egész típusú változók. Írja a vizsgalapra a **k** értékét az utasítássorozat végrehajtása után, tudva azt, hogy a beolvasott karakterek rendre ebben a sorrendben a **V I c T o r I E** betűk. **(6p.)**

```
k=0;
for(i=1;i<=8;i++)
{ cin>>a; | scanf("%c",&a);
  if(a>='A' && a<='Z') k=k+1;
  else k=k-1;
}
```

III. TÉTEL (30 pont)

1. Beolvasunk egy **n** természetes számot, a feladat az, hogy írassa ki a **d** és **p** számokat ebben a sorrendben, amely megadja azt a **d** prímtényezőt, amely a legnagyobb **p** hatványkitevőjű az **n** szám prímtényezőre bontásában. Ha több ilyen legnagyobb hatvánnyal rendelkező tényező van, akkor a legnagyobb tényezőt írassa ki.
Írja le pszeudokódban a kijelentett feladat megoldási algoritmusát.
Példa: ha **n=10780**, akkor a kiírt értékek **7 2** (**$10780=2^2.5.7^2.11$**). **(10p.)**

2. Írjon egy **C/C++** programot, amely a képernyőről beolvassa az **n** és **k** természetes számokat a **[2,20]** intervallumból. A memóriában felépít egy **n.k** elemű egydimenziós tömböt, amelynek elemei növekvő sorrendben vannak, a tömb első eleme **k** és minden eleme pontosan **k**-szor szerepel egymás mellett. A tömb két egymás melletti eleme vagy egyforma, vagy egymást követő számok. A program írja ki a képernyőre egy-egy szóközzel elválasztva a felépített tömböt.
Példa: ha **n=4 és k=3** a képernyőn megjelenik: **(10p.)** **3 3 3 4 4 4 5 5 5 6 6 6**

3. Adott a következő sorozat **1, 1, 2, 5, 14, 41, 122, 365, 1094**……
a következő módon meghatározva **$f_1=f_2=1$, $f_n=3.f_{n-1}-1$** (ahol n≥3 természetes szám)
Beolvasunk a billentyűzetről egy **x (x≤10^9)**, természetes számot a sorozat egy elemét. Írassuk csökkenő sorrendbe a sorozat elemeit egy-egy szóközzel elválasztva a **bac.txt** állományba, amelyek egyenlők vagy kisebbek az adott **x**-nél.
Tervezzen hatékony algoritmust a futási idő és a felhasznált memória szempontjából.
Példa: ha a beolvasott szám **365**
akkor a **bac.txt** állomány tartalma **365 122 41 14 5 2 1 1**
a) Írja le saját szavaival a használt algoritmust és indokolja annak hatékonyságát. **(2p.)**
b) Írja meg az előbb leírt algoritmusnak megfelelő **C/C++** programot. **(8p.)**