

Examenul național de bacalaureat 2021
Proba E. d)
INFORMATICĂ
Limbajul Pascal

Testul 2

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)

Îrja a vizsgalapra az 1 és 5 közötti feladatokra adott helyes válaszoknak megfelelő betűt! Minden helyes válasz 4 pontot ér.

- Az **x** és **y** egész típusú változók nullától különböző természetes számokat tárolnak. Határozza meg azt a Pascal kifejezést, amelynek az értéke **true** akkor és csak akkor, ha az **x** változóban tárolt szám egyeseinek számjegye 2, és többszöröse az **y** változóban tárolt számnak!
a. **(x div 10 mod 10=2) and (x mod y=0)** b. **(x div 10 mod 10=2) or (y mod x=0)**
c. **(x div 2 mod 10=10) and (x mod y=0)** d. **(x div 2 mod 10=10) or (y mod x=0)**
- Annak ellenőrzésére, hogy egy egydimenziós tömbben megtalálható-e az **x=16** érték, a bináris keresés módszerét alkalmazzuk, és a módszer alkalmazása során sorban a **14, 24, 16** elemekkel hasonlítjuk össze az **x** értékét. A tömb lehetséges elemei:
a. **(7, 8, 9, 14, 15, 16, 24)** b. **(14, 14, 24, 24, 16, 16)**
c. **(48, 16, 12, 24, 7, 14, 2)** d. **(48, 24, 16, 14, 9, 8, 4)**
- Az **A** egydimenziós tömb elemei: **A=(2, 20, 27, 36, 50)**, és a **B** egydimenziós tömbbel való növekvő sorrend szerinti összefésülése során a **(2, 3, 5, 8, 20, 27, 36, 45, 50, 63)** elemeket tartalmazó egydimenziós tömböt kapjuk. Határozza meg a **B** tömb elemeit, az előfordulásuk sorrendjében!
a. **(-47, -31, -19, 25, 51)** b. **(-19, -15, 1, 9, 13)**
c. **(63, 45, 8, 5, 3)** d. **(63, 45, 27, 3)**
- Egy Pascal kifejezés, amelynek értéke **false**:
a. **trunc(20.19)=20** b. **trunc(-20.19)=-20**
c. **trunc(19.20)=20** d. **trunc(-19.20)=-19**
- Az **x**, **y**, **z** és **w** változók egy-egy valós számot tárolnak úgy, hogy **(x>w) and (z>w) and (y>x)** a mellékelt C/C++ kifejezés értéke 1. Az a változó, amelynek az értéke a fent említett számok közül a legkisebb:
a. **x** b. **y** c. **z** d. **w**

SUBIECTUL AL II-LEA

(40 de puncte)

- Adott a mellékelt pszeudokód algoritmus.**
Az $a \% b$ jelölés az a természetes szám osztási maradékát jelenti a b nem nulla természetes számmal és a $[c]$ jelölés a c valós szám egész részét jelenti.

a. Írja le, mit ír ki az algoritmus, ha a következő szám lesz beolvasva 2186310! **(6p.)**

b. Írjon le két számot a $[10^4, 10^5]$ intervallumból, amelyeket beolvasva az algoritmus végrehajtása után mindkét esetben ugyanaz az érték íródik ki! **(6p.)**

c. Írjon az algoritmusnak megfelelő C/C++ programot! **(10p.)**

d. Írjon a megadott algoritmussal egyenértékű algoritmust, ahol az **amíg...végezd el** szerkezetet egy másik ismétlődő szerkezetre cseréli! **(6p.)**
- Reciprok polinomnak** nevezünk egy olyan polinomot, amelynek a széleitől ugyanolyan távolságra levő együtthatói egyenlők egymással (az első egyenlő az utolsóval, a második az utolsó előttivel stb.). Az a, b, c, d és e egy negyedfokú polinom együtthatóit tárolják, amelynek alakja $x^4 + a \cdot x^3 + b \cdot x^2 + c \cdot x + e$. Írjon egy C/C++ utasítás sorozatot, amelyet végrehajtva kiíródik a képernyőre a **RECIPROC** szó ha a polinom reciprok, vagy ellenkező esetben az **OARECARE** szó.

Példa: ha $a=3, b=4, c=1, d=4$ és $e=3$, kiíródik, hogy **RECIPROC**
de ha $a=3, b=4, c=1, d=3$ és $e=4$, kiíródik, hogy **OARECARE** **(6p.)**
- A c változó **char** típusú és az i és k változók egész típusúak.
Írja le kiegészítve a kipontozott részeknél az alábbi utasítás sorozatot úgy, hogy a végén a k értéke 1 legyen, ha be lett olvasva legalább egy **A** betű, ellenkező esetben pedig 0!

Például: ha a következő betűk olvasódnak be **P R E a f r U** $k := \dots \dots \dots$
M o A S A akkor a k értéke 1 legyen **for i:=1 to 12 do**
de ha a következő betűk olvasódnak be **P R E a f r U M o** **begin read(c);**
a S a akkor a k értéke 0 legyen **.....**
(6p.) **end;**

SUBIECTUL AL III-lea

(30 de puncte)

- Beolvassuk az n és m változókat, és ki kell íratni azoknak a prímszámoknak a számát, amelyek ugyanazon a hatványon jelennek meg az n prímtényezőkre bontásában, mint az m prímtényezőkre bontásában.
Írja le pszeudokódban a feladatot megoldó algoritmust!
Például: ha $n=16500$ és $m=10780$, akkor kiíródik a 2 ($16500=2^2 \cdot 3 \cdot 5^3 \cdot 11$, $10780=2^2 \cdot 5 \cdot 7^2 \cdot 11$). **(10p.)**
- Írjon egy C/C++ programot, amely beolvassa a billentyűzetről az n és k változókat ($n \in [2, 20]$, $k \in [1, n]$) és felépít a memóriában egy olyan n elemű egydimenziós tömböt, amely a következő jellemzőkkel rendelkezik:

 - al k -adik elem értéke n ;
 - a k -adik elemtől balra és a k -adik elemtől jobbra is úgyszintén az elemek egymásutáni értékek, és szigorúan csökkenő sorrendben vannak!

A program írja ki a képernyőre a tömb elemeit egy-egy szóközzel elválasztva!
Példa: ha $n=10$ és $k=3$ a képernyőre ki kell íratni, hogy 8 9 10 9 8 7 6 5 4 3 **(10p.)**
- A **bac.in** egy legfeljebb 10^6 egész, szóközzel elválasztott számokból álló sorozatot tartalmaz, amelynek elemei a $[-10^9, 10^9]$ intervallumban találhatók. Legalább két elem pozitív.
A program írja ki a képernyőre annak a leghosszabb részsornak a hosszát, amely egy pozitív számmal kezdődik és egy pozitív számmal végződik! Egy részsor olyan elemek alkotnak, amelyek egymásutáni pozíciókban találhatók és a hossza a részsor alkotó elemek száma. Tervezzon hatékony algoritmust a felhasznált memória és a végrehajtási idő szempontjából!

Példa: ha a fájlban levő sorozat a következő: -15 -7 4 -7 21 -5 -200 -26 52 -24 -7 -9 -20
A képernyőre kiíródik 7 (a 4 -7 21 -5 -200 -26 52 részsornak megfelelően).

a. Írja le saját szavaival a használt algoritmust, indokolva annak hatékonyságát! **(2p.)**
b. Írja meg a megtervezett algoritmusnak megfelelő C/C++ programot! **(8p.)**