

Gyakorló feladatok Szoftvertervezés és –fejlesztés I. tantárgy témaköréhez (2017. szeptember 15-i verzió)

A feladatokat Benkő Tiborné Visual Basic feladatai, programozás tankönyvek példái, valamint korábbi és mostani kollégáink (Erdélyi Krisztina, Sípos Marianna, Kotsis Domokos, Kurdi Zsombor, Miklós Árpád, Nagy Tibor, Szénási Sándor, Vámosy Zoltán) ötletei alapján állítottuk össze.

Készítse el az alábbi feladatok blokkdiagramját, struktogramját, majd pedig C# nyelvű programját.

1. FELADAT

Olvassunk be egy sugár értéket és számítsuk ki kör kerületét és területét, valamint a gömb felszínét és térfogatát.

2. FELADAT

Olvasson be három számot, majd írassa ki őket csökkenő sorrendben.

3. FELADAT

Készítsen programot, amely egy téglalap oldalhosszaiból kiszámolja a téglalap területét és kerületét!

4. FELADAT

Készítsen programot, amely egy háromszög oldalhosszaiból kiszámolja a háromszög kerületét és területét!

5. FELADAT

Írjon programot, mely egy háromszög oldalainak (a, b, c) hosszát olvassa be a billentyűzetről, majd megmondja, hogy a háromszög szerkeszthető-e! (A háromszög szerkeszthető, ha az $(a+b > c)$ és $(a+c > b)$ és $(b+c > a)$ feltétel teljesül.)

6. FELADAT

Olvasson be a billentyűzetről egy számot és mondjuk meg, hogy a szám negatív, vagy pozitív!

7. FELADAT

Kérjen be a billentyűzetről két számot, majd írja ki azok összegét, különbségét, szorzatát és hányadosát a képernyőre!

8. FELADAT

Készítsen programot, mely logikai true/false értékeket olvas be a billentyűzetről! True esetén a képernyőre az IGAZ szót írja ki a program!

9. FELADAT

Írjon programot, mely beolvasson egy számpárt a billentyűzetről, majd kiírja a két szám számtani közepét!

10. FELADAT

A téglalatest oldalainak ismeretében számítsa ki a lapátlókat, majd határozza meg a testátlót, a felszínt és a térfogatot!

11. FELADAT

A téglalatest oldalainak ismeretében határozza meg az oldalélek összegét, majd írassa ki a legrövidebb oldalél összeggel rendelkező test adatait, felszínét és térfogatát.

12. FELADAT

Deciliterben megadott súlyt bontsunk hektoliter, liter, deciliter egységekre!

13. FELADAT

Beolvasott egész számról döntsük el, hogy osztható-e kilenccel, a vizsgálatot a számjegyek összegének 9-cel való oszthatóságával végezzük, majd ellenőrizzük le maradékos osztással is.

14. FELADAT

Nem negatív egész számról határozza meg egy program, hogy hány jegyű!

15. FELADAT

Készítsen programot, amely egy adott szám adott hatványát számolja ki a szorzás műveletet felhasználva!

16. FELADAT

Készítsen programot, amely két szám szorzatát számolja ki az összeadás műveletet felhasználva!

17. FELADAT

Készítsen programot, amely egy másodfokú egyenlet együtthatóiból meghatározza az egyenlet gyökeit!

18. FELADAT

Készítsen programot, amely egy adott szám adott hatványát számolja ki az összeadás műveletet felhasználva!

19. FELADAT

Olvassa be egy hónap nevét, majd írja ki, hogy melyik évszakban van az adott hónap.

20. FELADAT

Kérje be egy cég éves bevételét és kiadását. Állapítsa meg, hogy nyereséges, veszteséges vagy nullszaldós volt-e a cég és az eredményt írja ki.

21. FELADAT

Kérdezzük meg a felhasználót, hogy beszél-e magyarul, angolul, spanyolul (vagy amilyen nyelven tudunk). Ha egyikre igennel felel, akkor köszönjük azon a nyelven, ha egyiket sem ismeri, írjuk ki neki egy ♥-et. (unicode: 2665)

22. FELADAT

Készítsen jelszóellenőrző programot: a program addig fut, amíg helyes jelszót nem kap, de próbálkozás után leáll (más üzenettel, mint amikor sikeres a belépés).

23. FELADAT

Olvassa be egy hónap sorszámát, majd írja ki, hogy melyik évszakban van az adott hónap.

24. FELADAT

Számítsa ki az éves középhőmérsékletet miután a felhasználó megadta a havi középhőmérsékleteket.

25. FELADAT

Készítsen programot, amely kávéautomatát szimulál: lehessen választani pohár nélkül, dupla cukorral és habos kávé közül. Az alapárhoz képest a dupla 20, a habos 50 Ft-tal drágább, a pohár nélküli 10-zel olcsóbb. A választás után az automata kérje el a megfelelő összeget.

26. FELADAT

Írjon címletező programot: a program megmondja, hogy mennyi kell egy adott összeg kifizetéséhez előre megadott (konstans) címletekből.

27. FELADAT

Készítsen pénzváltó programot: egy bekért pénznem és egy összeg alapján kiszámolja, hogy mennyi forint jár érte.

28. FELADAT

Kérje be egy számtani sorozat első elemét és differenciáját és írja ki a sorozat első 10 elemét.

29. FELADAT

Írja ki 10 első n hatványát, ahol $n < 15$ lehet és egész.

30. FELADAT

Tervezzünk egy alkalmazást, amellyel egy 1-100 között gondolt véletlen számot kitalálhatunk a következő módon: a felhasználó tippel, majd a program válaszként a következő lehetőségek közül egyet jelenít meg: sok – a szám kisebb; kevés – a szám nagyobb; eltalálta n lépés után.

31. FELADAT

Maximum n db. egész szám beolvasásakor döntsük el a negatívok összegét, a pozitívok átlagát és a zérusok darabszámát!

32. FELADAT

Beolvasott egész számról döntse el, hogy prím-e?

33. FELADAT

Beolvasott egész szám valós osztóit írja ki.

34. FELADAT

Készítsen programot, amely két szám legnagyobb közös osztóját adja meg!

35. FELADAT

Készítsen programot, amely két szám legkisebb közös többszörösét adja meg!

36. FELADAT

A beolvasott oldalhosszokból határozza meg egy sokszög területét a következő feltételekkel: (pozitív valós számokat gépel be a felhasználó); először a sokszög oldalszámát adja meg, majd az oldalak hosszát.

37. FELADAT

Készítsen programot, amely mindaddig olvas be egészeket, amíg 5 különbözőt nem adunk meg.

38. FELADAT

A beolvasott oldalhosszokból határozza meg egy sokszög területét a következő feltételekkel (pozitív valós számokat gépel be a felhasználó): nem adja meg előre az oldalak számát, hanem az oldalhosszak megadásának végét a 0 jelentse.

39. FELADAT

Kérjen be a billentyűzetről természetes számokat. Adja meg a billentyűzetről bekért számok átlagát. Az adatbevitel a -1 érték megadásával fejeződik be.

40. FELADAT

Egy vállalat alkalmazottai közül annak a nevére vagyunk kíváncsiak, aki a legtöbb fizetést kapja. Kérje be a billentyűzetről az alkalmazottak nevét és fizetését. Az adatbevitel akkor fejeződik be, ha a felhasználó alkalmazott neve helyett üres stringet ad meg.

41. FELADAT

Az előző feladatot egészítse ki úgy, hogy a fizetésnél a felhasználó csak pozitív egész számot adhasson meg. Ettől eltérő érték esetén írjon ki hibaüzenetet és addig ismétlje a fizetés bekérését, amíg a felhasználó helyes értéket nem ad meg.

42. FELADAT

Egy vállalat különböző osztályain dolgozó alkalmazottak neveit és fizetéseit kérjük be a felhasználótól (az osztályokat sorszámmal azonosítjuk: 0, 1, 2, ...). Írjuk ki annak az osztálynak a sorszámát, ahol a legmagasabb az átlagfizetés!

43. FELADAT

Egy vállalat különböző osztályain dolgozó alkalmazottak neveit és fizetéseit kérjük be a felhasználótól (az osztályokat sorszámmal azonosítjuk: 0, 1, 2, ...). Írjuk ki annak az osztálynak a sorszámát, ahol az osztály átlagfizetésének a minimálbérhez viszonyított eltérése a legkisebb (tegyük fel, hogy a minimálbér 70 000 Ft)!

44. FELADAT

Az előző két feladatot egészítse ki úgy, hogy a fizetésnél a felhasználó csak pozitív egész számot adhasson meg. Ettől eltérő érték esetén írjon ki hibaüzenetet és addig ismétlje a fizetés bekérését, amíg a felhasználó helyes értéket nem ad meg.

45. FELADAT

Számítsa ki egy szám négyzetgyökét Newton módszerrel. A beolvasott szám legyen A. $X_1=A/2$, $X_{n+1}=(X_n+A/X_n)/2$ ha $n \geq 1$. A számítást addig folytassa, amíg $|X_{n+1}-X_n| < 0,0001$. Az eredményt négy tizedes jegyre jelenítse meg!

46. FELADAT

Határozza meg két egész szám legnagyobb közös osztóját! Legyen N a kisebb, M a nagyobb szám. A közös osztó meghatározásának menete a következő:

1. R az M/N osztási maradéka
2. Ha $R=0$ vége, N a legnagyobb közös osztó
3. Ha $R < 0$, $M=N$ és $N=R$ és a 2. lépéstől folytatódik az eljárás.

47. FELADAT

Határozza meg az $F(x)=3x-2$ függvény gyökét. A számításhoz keressen egy olyan intervallumot, hogy az intervallum egyik végpontjában negatív, a másik végpontjában pozitív legyen a függvény helyettesítési értéke. Legyenek ezek az A és B pontok. A gyökszámítás menete:

1. $X=(A+B)/2$.
2. Ha $F(X)<0$, akkor $A=X$
3. Ha $F(X)>0$, akkor $B=X$
4. Az 1. ponttól folytatódik az eljárás, akkor van vége, ha $|A-B|<0,0001$, s ekkor X a keresett gyök.

48. FELADAT

Határozza meg az $x=\exp(-x)+x^3/3$ egyenlet gyökét az $x=0,7$ durva közelítés környezetében! A számítás menete:

1. $g(x)=\exp(-x)+x^3/3$
2. $X_e=0,7$
3. $X_{uj}=g(X_e)$
4. $X_e=X_{uj}$
5. Az eljárás a 3. lépéstől folytatódik

Az eljárást 50-szer hajtsa végre! Ezt követően X_{uj} lesz a keresett gyök.

49. FELADAT

Számítsa ki az 'e' szám értékét! Az érték meghatározásához az $e=(1+1/n)^n$ $n \geq 1$ sorozat értékeit kell kiszámítani. A számítást addig folytassa, míg két egymást követő kiszámított elem különbsége abszolút értékben kisebb, mint 0,00001. A hatvány kiszámításához használjon ciklust!

Feladatok tömbökkel

50. FELADAT

Töltsön fel egy mátrixot FOR ciklusok segítségével, majd állítsa elő a transzponáltját (a sorokat fel kell cserélni az oszlopokkal), és írassa ki mindkét mátrixot.

51. FELADAT

Készítsen keresztárfolyam táblázatot, melynek K_{ij} elemét úgy kapja meg, hogy az i -dik valuta árfolyamát elosztja a j -dik valuta árfolyamával. A valuták neveit (AUD, CHF, EUR, GBP, USD) és az árfolyamokat egydimenziós tömbben tárolja. Írassa ki a táblázatot a valuták nevének feltüntetésével.

52. FELADAT

Töltsön fel két mátrixot. Számítsa ki a két mátrix szorzatát. Ha az egyik mátrix elemei $a_{i,k}$, illetve $b_{k,j}$, akkor a szorzat mátrix egy elemét a következő módon kapja: $c_{i,j} = \sum_{k=1}^m a_{i,k} * b_{k,j}$, azaz az első mátrix i -dik sorának elemeit rendre meg kell szorozni a második mátrix j -edik oszlopának elemeivel, és a szorzatokat össze kell adni. Írassa ki mindhárom mátrixot.

53. FELADAT

Készítsen algoritmust pszeudokóddal és C# nyelvű programot, amely egy vállalat alkalmazottainak fizetését tartalmazó tömbből kiválasztja a legnagyobbat, a legkisebbet, az átlagtól legjobban, illetve legkevésbé eltérőt, kiírja a fizetések összegét és átlagát! Egészítse ki úgy, hogy a legnagyobb, legkisebb, átlagtól legjobban, legkevésbé eltérő fizetésű alkalmazott neve is íródjon ki!

54. FELADAT

Készítsen algoritmust pszeudokóddal és C# nyelvű programot, amely egy vállalat alkalmazottainak fizetését tartalmazó tömbben megszámolja, hogy hány olyan alkalmazott van, akinek a fizetése 100 000 Ft alatt van! Egészítse ki úgy, hogy a 100 000 Ft alatti fizetésű alkalmazottak nevei is íródjanak ki!

55. FELADAT

Készítsen algoritmust pszeudokóddal és C# nyelvű programot, amely egy vállalat alkalmazottainak fizetését tartalmazó tömbben minden alkalmazotról kiírja, hogy hány olyan alkalmazott van, aki nála többet, illetve kevesebbet keres!

56. FELADAT

Készítsen algoritmust pszeudokóddal és C# nyelvű programot, amely egy tömbben tároljuk minden hallgatónak minden tárgyból a félévközi jegyét. Adja meg annak a hallgatónak a sorszámát, akinek a tanulmányi átlaga a legjobb!

57. FELADAT

Készítsen algoritmust pszeudokóddal és C# nyelvű programot, amely egy tömbben tároljuk minden hallgatónak minden tárgyból a félévközi jegyét. Adja meg annak a tárgynak a sorszámát, amelyikből a legtöbb elégtelen született!

58. FELADAT

Készítsen algoritmust pszeudokóddal és C# nyelvű programot, amely az előző két feladatot egészítse ki úgy, hogy a hallgatók, illetve tantárgyak nevei is íródjanak ki!

59. FELADAT

Készítsen algoritmust pszeudokóddal és C# nyelvű programot, amely egy tömbben tároljuk minden hallgatónak minden tárgyból a félévközi jegyét. Kérje be a felhasználótól egy hallgató nevét. Írja ki azoknak a tárgyaknak a neveit, amelyekből a hallgató elégtelen szerzett!

60. FELADAT

Készítsen algoritmust pszeudokóddal és C# nyelvű programot, amely egy tömbben tároljuk minden hallgatónak minden tárgyból a félévközi jegyét. Kérje be a felhasználótól egy tantárgy nevét. Írja ki azoknak a hallgatónak a neveit, akiknek az adott tárgyból egy felhasználótól bekért értéknél jobb jegyük van!

61. FELADAT

Készítsen algoritmust pszeudokóddal és C# nyelvű programot, amely egy tömbben tároljuk minden hallgatóról, hogy az adott félévben mely tárgyakat vette fel (kétdimenziós tömbben, az adott hallgató sorának és az adott tárgy oszlopának találkozásánál lévő cellában 1-es szerepel, ha felvette, 0, ha nem vette fel a hallgató a tárgyat). Kérje be a felhasználótól, hogy mely tárgyakat kötelező a hallgatóknak felvenni. Írja ki azoknak a hallgatóknak a neveit, akik nem vették fel az összes kötelező tárgyat!

62. FELADAT

Készítsen algoritmust pszeudokóddal és C# nyelvű programot, amely egy tömbben tároljuk minden hallgatóról, hogy az adott félévben mely tárgyakat vette fel (egydimenziós tömbben, minden hallgatóhoz egy egész szám tartozik, amelynek minden egyes bitje egy tantárgyat reprezentál. Az adott tantárgyhoz tartozó bit 1-es, ha felvette, 0, ha nem vette fel a hallgató a tárgyat). Kérje be a felhasználótól, hogy mely tárgyakat kötelező a hallgatóknak felvenni. Írja ki azoknak a hallgatóknak a neveit, akik nem vették fel az összes kötelező tárgyat!

63. FELADAT

A műugró Európabajnokságon 10 bíró pontozza a bemutatott ugrásokat. Készítsen programot, amely kiszámolja a megadott 10 pontszámból az átlagpontot, ha

- A legnagyobbat és a legkisebbet nem számítjuk.
- Egyenlőség esetén egyik legnagyobbat és egyik legkisebbet sem számítjuk.

Feladatok függvényekkel

64. FELADAT

Készítsen függvényt, amely egy x szám y -adik hatványát számolja ki (y természetes szám)!

65. FELADAT

Készítsen függvényt, amely két – tömbben ábrázolt – mátrix szorzatát adja eredményül! Egészítse ki a feladatot úgy, hogy a függvény egy hibakódot is visszaadjon (pl.: -1-et), ha a mátrixok nem összeszorozhatók.

66. FELADAT

Készítsen függvényt, amely két szám szorzatát adja meg az összeadás műveletét felhasználva! Az eredmény az első paraméterben jelenjen meg!

67. FELADAT

Készítsen függvényt, amely két kétdimenziós tömbből előállít egy harmadikat úgy, hogy az eredménytömb minden egyes cellájában a bemenő tömbök azonos indexen lévő celláin található értékek közül a nagyobb szerepeljen! Egészítse ki a feladatot úgy, hogy a függvény egy hibakódot is visszaadjon (pl.: -1-et), ha a két tömb nem azonos méretű! Egészítse ki a feladatot úgy, hogy a függvény egy felsorolás típusú hibakódot is visszaadjon, ha a két tömb nem azonos méretű, illetve ha a két tömb minden megfelelő indexű eleme azonos értékű (nincs nagyobb elem az egyik, vagy másik tömbben)! A program írja ki, hogy melyik hiba következett be!

68. FELADAT

Készítsen két függvényt: az egyik a paraméterként megkapott tömbben lévő pozitív számok darabszámát, a másik a paraméterként kapott tömbben lévő pozitív számok összegét adja vissza! Készítsen egy harmadik függvényt, amely a paraméterként kapott tömbben lévő értékek közül visszaadja azt, amelyik a szintén paraméterként kapott értéktől a legkevésbé tér el. Az iménti három függvényt felhasználva egy hőmérsékleteket tartalmazó tömbből írja ki azt a hőmérsékletet, amely az átlaghőmérséklettől legkevésbé tér el!

Feladatok karaktersorozatokkal

69. FELADAT

Olvasson be egy mondatot, egy szót és azt, hogy a szót a mondat hányadik szava után kell beillesztve megjeleníteni a képernyőn. A szavakat elválaszthatja szóköz, vessző, pontosvessző, pont és kötőjel.

70. FELADAT

Olvasson be egy mondatot, és számolja össze, hogy a különböző magánhangzókból hány darab fordul elő a mondatban. (Deklaráljon egy tömböt a magánhangzók elhelyezésére, egy másikat a darabszámoknak és töltsze fel azokat.) A program a kis és nagybetűt is ismerje fel.

71. FELADAT

Kérje be egy ember teljes nevét!

- Írd ki, hány neve van. (A nevek között egy szóköz van.)
- Ha kettőnél több neve van, akkor csak az első kettőt hagyja meg, és azon köszöntse. (Tipp: addig vágja le a sztringet az utolsó szóköztől, amíg előlről és hátulról nézve a szóköz helyzete ugyanaz `IndexOf()`, `LastIndexOf()`)

72. FELADAT

Készítse el a következő programot: regisztráció céljából bekéri az illető email címét és születési évét. Ha nem email címet adott meg (nincs benne @), akkor erről értesíti, amúgy a cím @ előtti részével köszönti, és kiírja, hogy hány éves az illető.

73. FELADAT

Olvasson be egy mondatot, számolja meg, hogy hány szóból áll, majd írassa ki szavanként egymás alá! A szavakat elválaszthatja szóköz, vessző, pontosvessző, pont és kötőjel.

74. FELADAT

Olvasson be egy mondatot, majd írja ki a képernyőre úgy, hogy minden szó első betűjét nagybetűssé alakítja! A szavakat elválaszthatja szóköz, vessző, pontosvessző, pont és kötőjel.

75. FELADAT

Olvasson be egy mondatot, majd írja ki megfordítva.

76. FELADAT

Olvasson be egy mondatot, majd vizsgálja meg, hogy palindróm-e (tükrömondat, előről és hátulról olvasva ugyanaz). A vizsgálat eredményét jelenítse meg.
Palindróm például: Géza kék az ég.

77. FELADAT

Készítsen programot, amely egy karaktersorozatból az alábbi statisztikákat készíti el:

- magánhangzók száma,
- magánhangzók és egyéb karakterek aránya,
- szavak átlagos hossza.

78. FELADAT

Készítsen programot, mely inputként egy karaktersorozatot vár és eredményként a képernyőn megjeleníti a stringet, de az ékezetes betűk (áÁéÉóÓőŐúÚűŰ) helyére ékezet nélküli formát (aAeEoOoOuUuU-ra) ír ki.

79. FELADAT

Egy CD boltban egy héten át napi bontásban tároljuk az eladott CD-k árát (egyesével). Készítsen programot, amely megmondja, hogy melyik nap volt a legnagyobb a forgalom.

80. FELADAT

Készítse el 21-es játék egyszemélyes, gépi formáját:

Először kettőt „húz” a gép, majd kérésre újabbat „húz”. Amikor a játékos nem kér több lapot, akkor az addig húzott kártyák összértékének ezerszeresét kapja, de csak akkor, ha az nem haladja meg a 21-et.

81. FELADAT

Készítsen programot, amely egy fűrészfogas tömbben tárolja, hogy egy adott nap (hónap.nap) idén milyen napra esik/esett. Kérésre írassa ki a kért napot.

Készítsen objektum-orientált megoldást az alábbi feladatokra C# nyelven

82. FELADAT

Készítsünk „intelligens” tömböt, amely a következő jellemzőkkel bír:

- „int” vagy „float” típusú elemeket tárol (az elemek típusa legyen beállítható)
- Képes saját adatait kiírni a képernyőre
- Képes feltölteni saját magát véletlen számokkal (adott darab, adott intervallum használatával)
- Képes megadni az elemek összegét, átlagát, minimumát, maximumát
- Egy másik intelligens tömb elemeiből képes átvenni azokat az elemeket, amelyek még nem szerepelnek saját magában

Készítsünk továbbá egy másik osztályt, amelynek segítségével tesztelni lehet a fenti intelligens tömböt.

83. FELADAT

Készítsünk osztályt, amely a verem adatszerkezet megvalósítására szolgál. A verem adatszerkezet jellemzői a következők:

- „int” vagy „float” típusú elemeket tárol (az elemek típusa legyen beállítható)
- Egy művelettel egy új adatot tárolhatunk el a verem „tetején” („berakás”)
- Egy művelettel egy adatot vehetünk ki a veremből, és mindig csak a legutoljára berakott adatot („kivétel”)
- A verem maximális elemszáma legyen adott, ennél több adat tárolását ne engedje meg

Készítsünk továbbá egy másik osztályt is, amelynek segítségével tesztelni lehet a verem osztályt.

84. FELADAT

Készítsen osztályt, amely lakóhelyiségek alapterületét, falfelületét és a mennyezet méretét számolja ki a helyiség hossza, szélessége és belmagassága alapján.

Az osztály teszteléseként kérje be, hogy hány helyiségben szeretne az illető kövezni és festeni, majd kérje be a helyiségek adatait. Végül adja meg, hogy mennyi köre és falfestékre lesz szükség (az ajtók és ablakok méretét elhanyagoljuk).

85. FELADAT

Készítsünk egy számoló osztályt, amely két megadott valós számmal („float” típus) képes elvégezni a négy alapműveletet, és tárolja a műveletek eredményét is.

Készítsünk továbbá egy másik osztályt, amelynek segítségével tesztelni lehet a számoló osztályt.

86. FELADAT

Készítsünk egy „Háromszög” nevű osztályt, amely háromszögek kezelését valósítja meg. Az osztály a következő jellemzőkkel rendelkezzen:

- Legyen képes tárolni a 3 csúcs koordinátáit
- Legyen képes az adatok alapján eldönteni, hogy a háromszög egyenlőszárú, illetve szabályos-e
- Legyen képes kiszámítani a saját kerületét és területét

Készítsünk továbbá egy másik osztályt, amelynek segítségével tesztelni lehet a Háromszög osztályt.

87. FELADAT

Készítsünk egy „Kör” nevű osztályt, amely körök kezelését valósítja meg. Az osztály a következő jellemzőkkel rendelkezzen:

- Legyen képes tárolni a kör sugarát és középpontját
- Legyen képes a kört nagyítani és kicsinyíteni
- Legyen képes kiszámítani a saját kerületét és területét

Származtassunk a fenti Kör osztályból egy „SzínesKör” nevű osztályt, amely a továbbiakkal egészíti ki a Kör osztályt:

- Legyen képes tárolni, visszaadni és módosítani a kör színét

Készítsünk továbbá egy másik osztályt, amelynek segítségével tesztelni lehet a Kör és az OkosKör osztályokat.

88. FELADAT

Készítsük el egy SIM kártyakezelő egyszerűsített szimulációját. A kártyakezelő egyszerű és továbbfejlesztett SIM kártyákat kezel, amelynek sorszáma és PIN kódja van. Az egyszerű SIM kártya csak egyszer aktiválható, képes a PIN kód ellenőrzésére és többszöri hibás próbálkozás után képes érvényteleníteni saját magát. A továbbfejlesztett SIM kártya PUK kóddal is rendelkezik, melynek segítségével érvénytelenített állapotból újraaktiválható. A kártyakezelő legyen képes SIM kártyák létrehozására, tárolására és a helyes PIN (illetve PUK) kód megadását követően jelezze a felhasználónak, hogy az adott kártya aktív és használható. A kártyakezelő legyen képes továbbá az egyes SIM kártyákhoz tartozó egyenleg kezelésére.

89. FELADAT

Egy kártyalap a francia kártyában rendelkezik színnel (treff, káró, kör, pikk) és értékkel (2, 3, 4, 5, 6, 7, 8, 9, 10, J, D, K, A). Egy csomag kártya 52 lapból áll. Készítsen osztó programot, mely egy ilyen csomag kártyából adott számú (legfeljebb 4) játékosnak kioszt adott számú (legfeljebb 20) lapot.

(Megjegyzés: A kiosztott lapok számának és a játékosok számának szorzata természetesen nem lehet nagyobb mint 52, illetve osztáskor a lapok nem ismétlődhetnek.)

90. FELADAT

Készítsük el egy valós környezet szimulációját, melyben egy cég 5 db autóját kell működtetni az alábbi szabályok szerint:

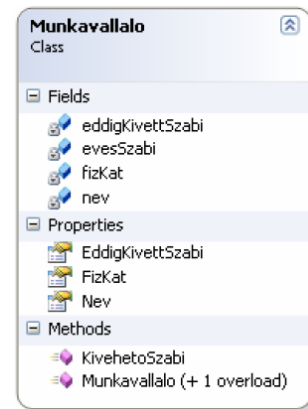
- Minden autónak van neve, a sofőrnél van valamennyi pénz, az autónak van fogyasztása, a benzintankjában bizonyos mennyiségű benzin, valamint megadható, hogy melyik autónak mekkora a benzintankja
- A cégnek van egy benzinkútja, ahol tankolni lehet. A benzinkútnak adott mennyiségű benzint tárol. A benzinnel ára van (literenként), és a benzinkútnak van kasszája is, ahova fizetni kell a tankolásért.
- A cég főnöke az autókat elküldi adott hosszúságú útra.
- A kiválasztott autó elindul, ha van elég benzinje, ellenkező esetben beáll tankolni.
- Az autó akkor tud tankolni, ha a sofőrnek van elég pénze és a benzinkútban van elég benzin.
- Ha az autó nem tud eljutni a kívánt távolságra, és nem tud eleget tankolni (nincs pénz, vagy nincs benzin a kútban), akkor az autó álljon le.
- A szimuláció akkor ér véget, ha minden autó leállt.

91. FELADAT

Készítsen egy munkavállaló osztályt. A mezői: név, fizetési kategória, éves szabadság és eddig kivett szabadság. Kétféle konstruktora legyen: az egyik paramétere a név és a fizetési kategória, a másiknak az előzőn kívül az éves szabadság is.

Készítsen metódust, amely a még szabadságként kivethető napok számát adja vissza.

A diagramnak megfelelően hozzon létre tulajdonságokat (a fizetési kategória 1-5 lehet). Tesztelje az osztályt



92. FELADAT

Készítsen olyan alkalmazást, amely két személy adatainak megadása után megmondja, hogy összeillenek-e.

a) Egy személyről a nevét, a magasságát és a korát tároljuk. A mezőkhöz tulajdonságok is tartozzanak. A személynek nem feltétlen kell minden adatát megadnia. Lehet, hogy csak a nevét, vagy a nevét és a magasságát, vagy minden adatot megad.

b) Hogy mikor illik össze két ember, találjon ki hozzá algoritmust.

A személy osztály példányosításához (az adatok bekéréséhez) és az összeállítás eldöntéséhez is készítsen egy osztályt.

93. FELADAT

Egy számítástechnikai eszközöket forgalmazó boltnak készítsen alkalmazást. Az eszközökről tároljuk az árukat, a készleten lévő mennyiséget, és bizonyos műszaki paramétereiket. Az eszköz képes kiírni a jellemzőit, és megadni, hogy kifogyóban van-e (10-nél kevesebb van készleten). Bizonyos gyorsan avuló eszközöket (de csak azokat) időnként leértékelik.

a) Gondolja végig, hogy az eszközökből alkotott osztályokat hogyan lehet hierarchiába foglalni.

b) Valósítsa meg, és tesztelje az alkalmazást.

c) Tárolja a készleten lévő eszközöket egy tömbben. A gyorsan avuló eszközöket értékelje le 10%-kal.

Készüljön egy „teljes konfiguráció” osztály is. Ez a többi eszköz osztályt egy tömbben foglalja magában. Ez hol helyezkedik el a hierarchiában? Készítse el hozzá a megfelelő metódusokat.

94. FELADAT

Készítsen osztályt, mely tárol egy szöveget, és a szöveget képes nagybetűsen, kisbetűsen kiírni. Tudjon szövegrészre rákeresni, ki tudja írni, hogy egy szövegrész hányszor fordul elő és egy-egy előfordulás hol kezdődik. Egy példány tartalmazza Móricz Zsigmond: „A török és a tehenek” című versét. Keressük meg benne hányszor szerepel benne a tehen szó és ragozott változatai. A tehen szót és ragozott változatait írjuk át a szövegbe csupa nagybetűvel!

95. FELADAT

Feladat statikus és öröklés bemutatására

Készítsünk egy Tag nevű osztályt, mely egy csoport tagjait tartalmazza. A tagnak van neve, felelős, kedveltsége (Ez lehet numerikus, vagy felsorolás típusú, pl semmiképp nem dolgoznék vele, ha kell dolgozok vele, ha úgy alakul jól dolgozok vele, szívesen dolgozok vele.) Az osztály tartalmaz egy TagokSzama statikus tulajdonságot, (mögötte egy azonos nevű statikus tagot).

Származtassunk egy AHétTagja nevű osztályt a Tag-ból, mely a felsoroltakon kívül egy sorszám mezővel is rendelkezik, hányadik héten lett a hét tagja.

A főprogramban lehessen közönségsvazatot adni a hétTagja címre javasolt összes tagra. (Sorszámozva kiírjuk a tagokat és a sorszám beírásával lehet szavazni. Végjel lehet:*)

Figyeljük meg, hogy az új a hétTagja objektum növeli a tagokSzama mező értékét! Döntsük el, hogy a tagokSzama a létrehozott különböző objektumok számát jelenti, mert akkor a program helyes értéket ad, vagy a csoport tagjainak számát ami itt értelmesebb feladat, akkor a hétTagja objektum létrehozásakor nem növekedhet a tagok száma!

96. FELADAT

Feladat értékadásra, típuskényszerítésre és elrejtésre (nem alapértelmezett őskonstruktor hívásra, protected láthatóságra (a típus esetén a tulajdonság, beágyazott osztálynál a felsorolástípus láthatósága is protected lehetne, de ha a konstruktorhoz kell, akkor public lesz.)

Készítsünk egy Árú osztályt. Tartalmazzon név, típus, készlet, maxDb, és nettóÁr mezőket az egységbezárás adatretjés elvét betartva. A típus legyen felsorolástípus, és a következő elemek közül választható: tejtermék, pékáru, édesség, mirelit húskészítmény és egyéb. A típus nem módosítható, de a létrehozáskor megadható. Alapértelmezésben egyéb. Készítsünk utódosztályokat: TejTermék utódosztály minden tagjának típusa tejtermék legyen és tartalmazzon egy zsirtartalom nevű értéket. A pékáru minden tagjának pékáru legyen a típusa és adjuk meg, hogy friss, vagy tartós termék. Az édességeknél határozzuk meg a tartalmaz-e cukrot mezőt. Minden osztálynak legyen Kiír tagfüggvénye, mely a konzolra kiírja az adott osztály tagjait és azok értékét.

Egy tömbben tároljuk az árúkat. Töltsük fel a tömböt, különböző típusú árúkkal!

Írjuk ki a tömb elemeit a konzolra!

Először típuskényszerítéssel (ha van idő különböző technikákkal) majd virtuális metódussal.

Számítsuk ki a raktárkészlet értékét!

Láthatóan TejTerméket létrehozhatunk úgy, hogy Árút hozunk létre, melynek típus mezője tejtermék, vagy úgy hogy TejTermék objektumot hozunk létre. Ez nem szerencsés, mert az Árú típusú tejterméknek nincs zsirtartalom mezője. Ne lehessen Árú típusú tejterméket létrehozni! Vagy úgy, hogy a konstruktorba ne lehessen a típust megadni, alapértelmezés egyéb, és a típushoz írt tulajdonság beállításakor csak a többi típust fogadja el, vagy legyen az Árú osztály absztrakt, és minden típushoz legyen alosztálya.

97. FELADAT

Feladat GyümölcsBolt (öröklés, ToString() felüldefiniálása, kivételkezelés bemutatására)

A boltban gyümölcsöt és üdítőt árulnak. Minden árunak van neve és ára. Az árúkat dl, l, g vagy kg mértékegységekkel mérjük. Hozzon létre egy felsorolástípust az egységek tárolására. Minden árú ismeri az egységét amiben mérjük, és a mennyiségét az adott egységben. Pl. 2 dl almalé, vagy 2l Cola, vagy 1 kg körte. Az italokat csak dl, l-ben mérhetjük, a gyümölcsöket csak g, kg-ban. Az italok tartalmazzák még a gyümölcsstartalmukat %-ban, a gyümölcsökről pedig tároljuk, hogy frissek-e.

Minden osztályhoz definiálja felül a ToString() metódust úgy, hogy az egy szépen kiírt sztringet adjon vissza az adatokról.

Az adatokat az egységbezárás elvét betartva mentse el.

Készítsen egy 10 elemű tömböt, mely a bolt árúit tartalmazza. Töltse fel 3 313mét kódból különböző típusú 3 objektummal. Írassa ki a tömb tartalmát. Ha a teljes tömböt kiírhatja kivételt dob a program, kezelje a kivételt. Olvasson be további elemeket a billentyűzetről addig, amíg a név mezőben * jelet nem adnak. Kezelje a hibás adatokat az eddig tanult módon vagy kivételkezeléssel.

Írassa ki csak a gyümölcsöket!

Írassa ki az üdítők gyümölcsstartalmát mennyiségben ne százalékban.

Összetett feladatok a félév végére

98. FELADAT

Készítsünk egy háttértároló osztályt, amely fájlok listáját kezeli az alábbi módon:

- A háttértárolónak van egy maximális tárolókapacitása, melyet csak konstruktorban lehet beállítani
- Van egy Format() metódusa, mely üríti a fájlok listáját
- Van egy MaximálisKapacitás tulajdonsága, amellyel le lehet kérdezni a maximális kapacitás értékét
- Van egy SzabadKapacitás tulajdonsága, amellyel le lehet kérdezni a meghajtó szabad kapacitását
- Van egy FoglaltKapacitás tulajdonsága, amellyel le lehet kérdezni a tárolt fájlok össz méretét
- Van egy Hozzáad() metódusa, amellyel új fájlt lehet hozzáadni, ha ugyanilyen nevű fájl nincs még a háttértárolón, illetve az új fájl elfér a háttértárolón
- Van egy Keres() metódusa, amely egy megadott fájlnevé alapján megkeresi és visszaadja a fájlt
- Van egy Töröl() metódusa, amely letörli a megadott fájlt, amennyiben létezik

A fájlok jellemzői a következők:

- Van nevük és méretük
- Van egy-egy CsakOlvasható, Rendszer és Rejtett attribútumuk

Fejlesszük tovább az alap háttértároló osztályt Floppy osztállyá az alábbi módosítások szerint:

- A floppy mérete 1440KB
- A floppy-nak van írásvédő tolokája, amely ha „írásvédett” állapotba kerül, akkor a floppyn sem a Format(), sem a Hozzáad(), sem a Törölés() nem működik

Fejlesszük tovább az alap háttértároló osztályt DVD osztállyá az alábbi módosítások szerint:

- A DVD mérete 4700MB
- A DVD alaphoz még írható, törölhető, de miután meghívjuk a Zárolás() metódusát (DVD felírása), akkor utána már nem használható rajta sem a Format(), sem a Hozzáad(), sem a Töröl() („egyszer írható DVD”)
- A zárolt DVD szabad kapacitása mindig 0 legyen

Fejlesszük tovább a DVD osztályt DVD-RW osztállyá az alábbi módosítások szerint:

- A DVD-RW többször is írható DVD, ezért van egy Megnyitás() metódusa is. Ekkor a DVD lemez visszaáll egy üres alapállapotba.

Fejlesszük tovább az alap Háttértároló osztályt HDD osztállyá a célnak megfelelő módosítások szerint.

Fejlesszünk ki egy Számítógép osztályt az alábbiak szerint:

- Több háttértárolója is lehet, melyek egy Felcsatol() metódus segítségével lehet csatlakoztatni a géphez
- Van egy Összkapacitás tulajdonsága, amely megadja az összes háttértároló teljes kapacitásának összegét
- Van egy SzabadKapacitás tulajdonsága, amely megadja az összes háttértároló teljes szabad kapacitásának összegét
- Van egy FoglaltKapacitás tulajdonsága, amely megadja az összes háttértároló teljes foglalt kapacitásának összegét
- Van egy Archivál() metódusa, mely a megadott fájlt megkeresi a háttértárolók valamelyikén, majd a megadott másik háttértárolóra a megfelelő módon archiválja a fájlt. Amennyiben nem adunk meg másik háttértárolót, a metódus keres egy másik használható háttértárolót és oda végzi el az archiválást.

99. FELADAT

Készítsünk egy személyeket reprezentáló osztályt. A személyekről tároljuk nevüket és lakcímüket (string típusú attribútumok)!

Tároljuk a személy születési dátumát is egy három egész számot tartalmazó tömbben.

Készítsünk konstruktorokat az osztályhoz!

- Csak nevet adunk meg.
- Nevet és címet adunk meg.
- Nevet és a születési dátumot tartalmazó tömböt adjuk meg.
- Nevet, címet és születési dátumot is megadunk.

Készítsünk 1 csak olvasható property-t a névhez és egy írható-olvashatót a lakcímhez!

Készítsünk egy string paraméterű indexelőt az osztályhoz, ami a születési dátum egyes elemeit (év/hónap/nap) adja vissza a paraméter értékétől függően!

Definiáljuk felül a ToString() metódusát értelemszerűen.

Készítsünk egy Kor() metódust.

Öröklésszünk egy alkalmazott nevű osztályt a Személy-ből! Kiegészítés: beosztás, fizetés.

Készítsünk konstruktorokat az Alkalmazott osztályhoz!

- Csak nevet adunk meg.
- Nevet, beosztást és fizetését adunk meg.

Készítsünk egy fizetésemelés metódust az Alkalmazott osztályhoz!

Készítsünk egy Név csak olvasható property-t az Alkalmazott osztályhoz, amely zárójelek között visszaadja az illető beosztását is!

Definiáljuk felül a Kor() metódust: akinek 300 000 Ft-nál magasabb a fizetése, annak a korából letagadunk 10 évet.

Teszteljük az osztályokat.

Próbáljuk ki a típuskényszerítést.

100. FELADAT

1. Papíron rajzolja le a következő osztályokat:

a. Hallgató

i. Adattagok:

1. név
2. neptunkód
3. ösztöndíj
4. felvett kurzusok

ii. Metódusok:

1. ToString(): adatok kiírása a konzolra
2. ElsoVezetekNev(): adja vissza a hallgató első vezetéknevét
3. NevekSzama(): adja vissza, hogy a hallgatónak hány neve van
4. JegyAtlag(): adja vissza a felvett kurzusok jegyeinek átlagát
5. KreditErtek(): adja vissza a felvett kurzusok teljesítésével megszerzett összkreditet
6. LegjobbAtlagu()

b. Tanár

i. Adattagok:

1. név
2. neptunkód
3. fizetés
4. tanított tantárgyak
5. heti összóraszám

ii. Metódusok:

1. ToString(): Adatok kiírása a konzolra
2. ElsoVezetekNev(): adja vissza a hallgató első vezetéknevét
3. NevekSzama() : adja vissza, hogy a hallgatónak hány neve van
4. Oradij(): a fizetés és a heti összóraszám alapján adja vissza, hogy a tanár egy órájáért átlagban hány forint fizetést kap
5. LegnagyobbFizetesu()

c. Tantárgy

i. Adattagok:

1. név
2. neptunkód
3. vezetőtanár neve
4. kreditérték
5. heti óraszám

ii. Metódusok:

d. Kurzus

i. Adattagok:

1. Tantárgy

2. Csoporkód

ii. Metódusok:

2. Öröklést használva küszöbölje ki az osztályok tagjai közti átfedéseket (értelemszerűen)!

3. Valósítsa meg az osztályokat C#-ban! A megvalósításnál tartsa be az egységbezárást elvét: legyen minden adattag privát láthatóságú, a kívülről elérni kívánt adatokhoz készítsen tulajdonságot!

4. Készítsen Program osztályt és benne egy Main() metódust!

5. Készítsen adatbevitelt a fenti osztályokhoz! Készítsen egy tömböt, amelyben tanárokat és hallgatókat egyaránt el lehessen helyezni. Kérje be a billentyűzetről, hogy a felhasználó hány hallgató és/vagy tanár adatait akarja bevinni, ezután minden egyes személyről kérdezze be az hozzá tartozó adatokat.

6. A bekérésnél minden esetben ellenőrizze, hogy a felhasználó helyes értéket adott-e meg. Ha nem, írjon ki hibaüzenetet és kérje be újból a hibásan megadott adatot.

7. Írja ki azoknak a tanároknak az adatait, akik által tanított tárgyak összkreditértéke a legnagyobb!

8. Írja ki annak a hallgatónak az adatait, aki a legkevesebb kreditet gyűjtötte a félév során!

9. Készítsen a Tanár osztályban egy metódust, amely a két paraméterben megkapott Tanár közül a nagyobb fizetésűt adja vissza. (A metódus használatához ne kelljen az osztályból példányt létrehozni!)

10. Készítsen olyan metódust is, amely az előző feladatban szereplő műveletet végzi, de tetszőleges számú tanárral.
11. A Main() metódusban a tömbbeli tanárok közül írja ki a legnagyobb fizetésűnek az adatait!
12. Készítsen a Hallgató osztályban egy metódust, amely két, paraméterben megkapott hallgató közül adja vissza a legjobb jegyátlagút! (A metódus használatához ne kelljen az osztályból példányt létrehozni!)
13. Készítsen olyan metódust is, amely az előző feladatban szereplő műveletet végzi, de tetszőleges számú hallgatóval.
14. A Main() metódusban a tömbbeli hallgatók közül írja ki a legjobb jegyátlagúnak az adatait!