

12.bc programozás beadandó

Információk

A beadandó három részből fog állni. Programozás, dokumentálás, tesztelés. Mindegyik feladatrészre egy-egy jegyet fognak kapni, a megfelelő tantárgynál.

Határidő: 2024. január 10. Szerda 23:59:59

Programozás

1. Minden feladat grafikus (FORM) felületen legyen megoldva! (Szükséges a rajz vagy grafikon vagy táblázat stb. használata. Amit a feladat kér/ajánl.) A grafikus felületnek esztétikusnak és az eseményeket tekintve jól működőnek kell lennie. Nem működő funkciót ne építsen be! Pontlevonás jár érte!
2. A program megírása során, osztályokat, metódusokat kell használnia. Minden metódus mélysége legfeljebb 2 lehet! Legyen minden kiszervezve külön metódusba. (A programozás során tartsa szem előtt a tiszta kód elveit)
3. Mindenhol függvénnyel dolgozzon, csak kiírások során használhat eljárást!
4. Figyeljen a tiszta kód elveire:
 - a. Egy metódus hossza (zárójelek nélkül) 8-12 sor lehet.
 - b. Minden legyen kiszervezve metódusokba.
 - c. Használjon osztályokat és az osztály példányaira jellemző metódusokat az osztályban írja meg!

Tesztelés:

1. minden metódus rendelkezzen Summary-val (///)
2. minden metódushoz rendelkezzen tesztfüggvénnyel, a tesztfüggvény közvetlenül a metódushoz legyen megírva,
3. minden property is rendelkezzen tesztfüggvénnyel,
4. nem kell minden függvényhez kimerítő tesztet írni, de elvárás, hogy az összes ekvivalenciaosztály legyen lefedve,
5. szerepeltesse az összes tanult attribútumot,
6. használjon legalább 1 olyan attribútumot is, amiről tanórán nem esett szó.

Dokumentáció:

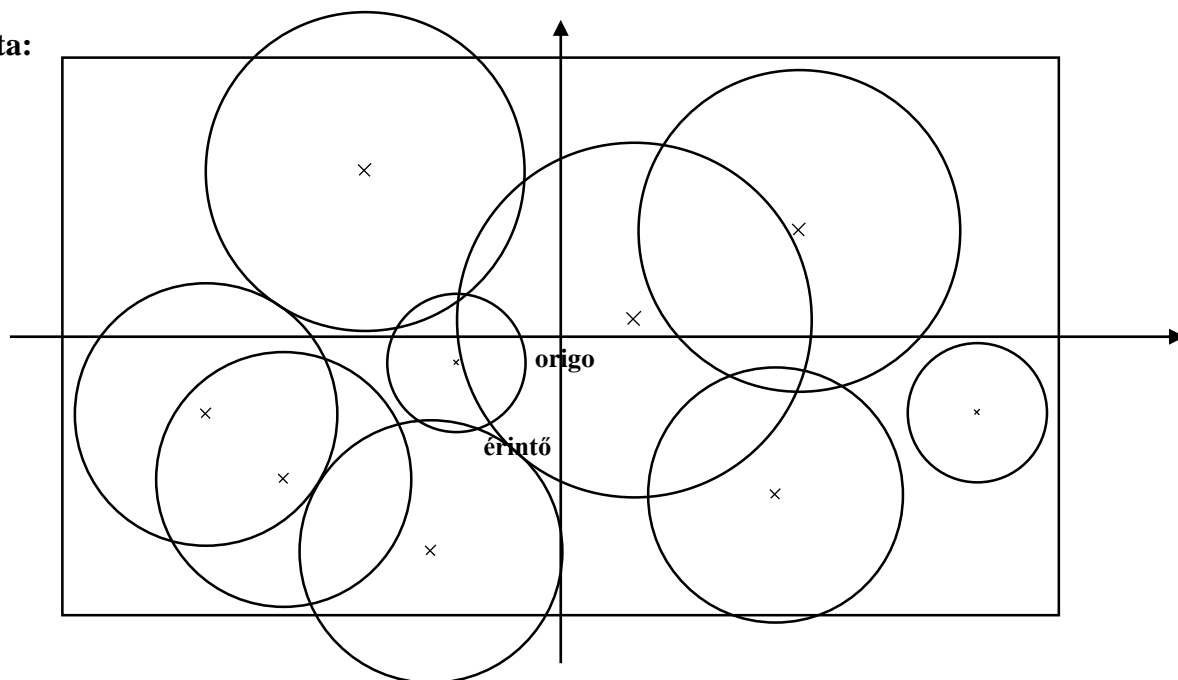
1. Készítsen egy több oldalas dokumentációt, amin
 - a. bemutatja a program működését, képekkel illusztrálva, pontos feladat leírással!
 - b. bemutatja a főbb funkciókat. Kifejti a program megírása közben létrehozott főbb osztályok lényegét, folyamatát.
 - c. végül bemutat legalább 10 különálló tesztesetet, ami különböző nézőpontból nézi meg az adott metódust.

Körök

1. **Feladat:** Írjon programot, amely listában tárolt körökkel statisztikákat és számításokat végez!
 - a. Adott egy vagy több szöveges állomány, amit Önnek kell létrehoznia a feladat leírás alapján! Az adatok beolvasása során a beolvasott adatokra adja meg a következő választ!
 - b. **Kör**
 - Adott a középpontjának az „x” és „y” koordinátái, a harmadik szám a kör sugara „r”.
 - Lekérhető legyen a kör Területe, Kerülete és az origótól mért Távolsága (középpontra nézve)
 - Lehessen a kör adatait egyben lekérni.
 - c. **Statisztika**
 - Tárolja el egy egységes adatszerkezetben a körök adatait!
 - Ha nincs beolvasott fájl, ne álljon le hibával, ajánljon fel egy generálási lehetőséget a felhasználónak, ami a lehetőségekhez mérten generálja ki a körök adatait. A felhasználó adhassa meg a darabszámot, ha annyit nem tud kigenerálni, akkor figyelmeztessen a program és ajánljon egy darabszámot! Figyeljen rá, hogy megjelenítés során legfeljebb 20%-os mértékben csúszhatnak össze a körök!
 - Válaszoljon a következő kérdésekre:
 1. Lehessen kiírni az összes kör adatát!
 2. Lehessen lekérni, hogy hány darab kör tartalmazza az origót.
 3. A körök mekkora területet foglalnak el a megjelenítő felületen. Ha a körök összecsúsznak, akkor a metszetet ne számolja kétszer!
 4. Lehessen lekérni, hogy melyik kör van a legtávolabb!
 5. Lehessen lekérni két kör távolságát!
 6. Lehessen megkeresni azokat a köröket, amelyek csak érintik egymást!
 - d. Készítsen a feladathoz egy grafikus megjelenítő alkalmazást! Jelenítse meg a koordináta rendszer tengelyeit és körök középpontját és magukat a köröket! A megoldás során arányosan alakítsa ki grafikus felület nagyságát, hogy az összes kör arányosan beleférjen és látható legyen!

- e. Hozzon létre egy grafikus felületet, amin ábrázolj a programot! Legyenek tengelyek, és legyen jelölve a kör középpontja is! Ne legyenek a körök kitöltve!

Minta:



Autószerelő műhely munkabeosztása

2. **Feladat:** Írjon programot, amely egy autószerelő műhely szerelőinek napi munkabeosztási tábláját tervezi meg és jeleníti meg a bejövő munka-megrendeléseknek megfelelően! Adott egy Szerviz, ahol szerelők dolgoznak. Mindegyik szerelőnek megvan a maga munkája. Egy beosztást több szerelő is végezhet, de mindegyik szerelőnek egy beosztása lehet csak! Lehessen több napra (akár dátum alapján) foglalásokat indítani!
- a. Adott egy vagy több szöveges állomány, amit Önnek kell létrehoznia a feladat leírás alapján! Az adatok beolvasása során a beolvasott adatokra adja meg a következő választ!
- b. A Szerelők:
- munkaköre/beosztás: f – futómű; o – olajcsere; e – elektronika; k - kerékcseré
 - nev: a szerelő neve
 - minden szerelő munkába áll és nincs szabadságon 10 órás munkaideje van.
 - Lehessen lekérni a szerelők elfoglaltságát!
 - Lehessen lekérni, hogy hány és melyik munkaórái vannak lefoglalva.
- c. Munkatábla:
- Egy összetett adatszerkezetben tárolja a szerelők adatait!
 - Lehessen lekérni egy beosztás alapján, hogy kik végzik el.
 - Lehessen lekérni egy munkakör alapján, hogy kinél melyik órában van szabad hely!
 - Lehessen rendezni az adatokat név és foglaltság alapján!
 - Lehessen foglalni időpontot. A foglalás kezdő és befejező óra megadásával lehessen. *Pl.: e 2 3, azaz elektronikát szeretnék javítani a 2. órától az 4. óráig.* Foglalásnál vegye figyelembe, hogy csak szabad helyre lehessen foglalni.
 - Lehessen lekérni egyénileg is és összesen is a szabad órákat!
- d. Hozzon létre egy szerelők.txt fájlt a fentebb megadott formátumban úgy, hogy legalább 10 szerelő szerepeljen benne, és legyen benne olyan is akinek több keresztnéve is van.
- e. Készítsen a feladathoz egy grafikus megjelenítő alkalmazást!
- Jelenítse meg a dolgozókat és az órákat!
 - Lehessen napokhoz kötni a foglalást. Vagy dátum alapján vagy legalább 10 napos intervallumra.

- A megjelenítés során figyeljen a színezésre, illetve a szövegnél a minta alapján a név mellette jelenjen meg a munkaköre.
- Kijelöléssel tudjon foglalni az üres helyekre. A pirosat, azaz a már foglalt helyeket ne tudja felülírni.
- Lehesen lekérni, hogy melyik szerelőnek melyik napon van szabad ideje. Ezt is jelenítse meg valamilyen formában!

f. Extra, az Ön által kitalált funkciókat is építhet bele, ha a fentiek elkészültek!

Példa:

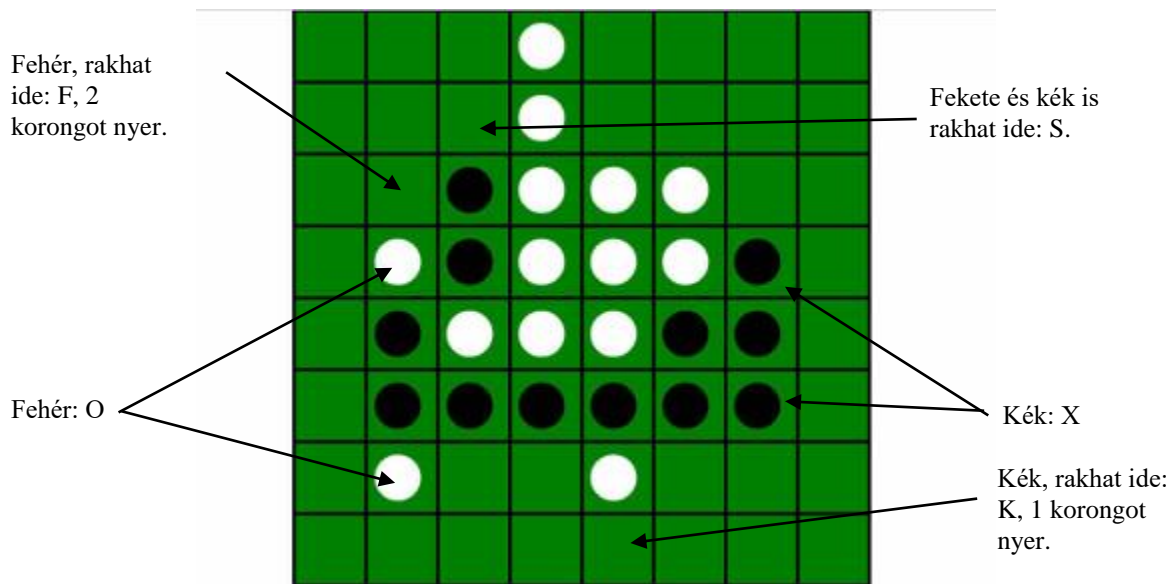
	1	2	3	4	5	6	7	8	9	10
Kis Pista (Olajcsere)										
Banános Joe (Olajcsere)										
Bevíz Elek (Elektronika)										

Olajcsere, foglal 2,2 ➔ Kis Pista

Elektronika, foglal 3,3 ➔ nincs időpont

Reversi

3. **Feladat:** Írjon programot, amely a Reversi játék egy állapotát elemzi ki.
- Hozzon létre egy grafikus felületet, amin ábrázolj a programot!
 - Lehessen látni az aktuális állást!
 - A végén jelezze ki nyert!
 - Lehessen több kört is játszani!
 - Lehessen gép ellen is játszani!
 - Két játékos esetén lehessen menteni az adott állást és visszakérni a mentés eredményét!



4. **Feladat (Forma 1):** Írjon programot, amely a Forma 1 verseny adatait vizsgálja.

- a. Adott egy vagy több szöveges állomány, amit Önnek kell létrehoznia a feladat leírás alapján! Az adatok beolvasása során a beolvasott adatokra adja meg a következő választ!
- b. Dolgozzon a következő adatokkal:
 - Versenyző: neve, kora, magassága, autóinak listája, adott pályán elért maximum sebessége, adott pályán az átlag sebessége, adott pályán elért helyezése, adott pályán használt autó(k), befejezte-e a versenyt
 - Autók: azonosító, lóerő, csúcsebesség, ürtartalom, javítás alatt van.
 - Csapat: versenyzőinek listája, autók listája
- c. Lehesse egy adott szezon pályái közül választani és lehesse lekérni a végeredmény adatait! Bármelyik pályát kiválasztva lehesse lekérni bármelyik versenyző, csapat vagy autó adatait!
- d. Lehesse rendezetten megjeleníteni az adatokat. A felhasználó tudjon több mező alapján rendezni.
- e. Hozzon létre egy grafikus felületet, amin ábrázolja a bemeneti adatokat.
 - Tudjon a program szűrni legalább két feltétel alapján! (csapat, kor)
 - A fent említett funkciókat jelenítse meg az Ön által választott események alapján (kattintásra, egér ráhúzásra, stb)

Minta:

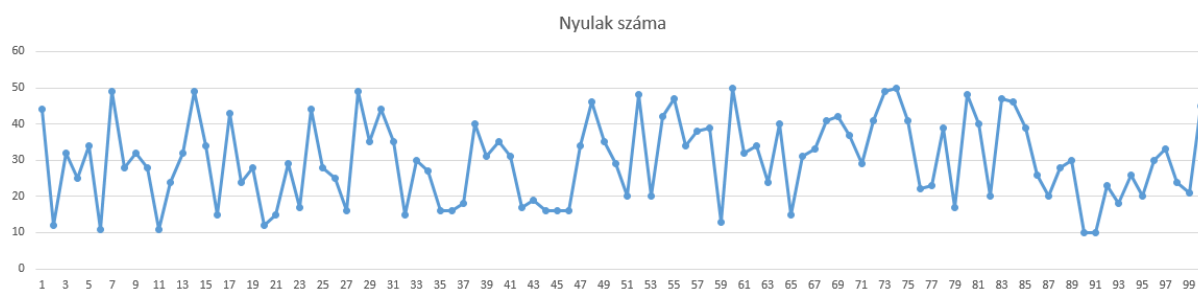
[illegible]

5. **Feladat (Nyulak a mezők):** Írjon programot, amely a mezőn lévő nyulak életét szimulálja és vizsgálja.
- a. Adott egy vagy több szöveges állomány, amit Önnek kell létrehoznia a feladat leírás alapján! Az adatok beolvasása során a beolvasott adatokra adja meg a következő választ!
 - b. Írjon osztályokkal, metódusokkal programot, amit elvégzi a következő feladatok:
 - Lehesse lekérni a nyuszi aktuális helyzetét, korát!
 - Egy mezőre legyen feltöltve nyuszikkal. Ezt lehessen egy fájlból bekérni, de lehessen véletlenszerűen kigenerálni. A megjelenítő felület legyen egy kezdő állapot!
 - A szimuláció aktuális állapotát lehessen kimenteni külön fájlba, olyan szerkezettel, amit később felhasználhatók beolvasás során!
 - Minden beolvasás vagy kiírás során legyen egy kommunikáló ablak a felhasználónak!
 - A szimulálás egy-egy lépés, amit minden nyuszin végre kell hajtani először.
 - Lépések:
 1. Születés: az üres mezőket végig járva, ha a környezetében pontosan három nyuszi van, akkor az adott területen új nyuszi születik, ezek a következő körben kerüljenek bele az adott adatszerkezetbe.
 2. Meghalás: ami bejárva a már meglévő nyuszikat, ha két vagy három nyuszi áll a szomszédságában akkor túléli és bekerül a következő listába, más különben a nyuszi a túlnépesedés miatt elpusztul.
 - A következő állapot listával írjuk felül a nyuszi listát.
 - *Figyeljünk oda, hogy a mező körkörös, azaz ha a nyúl bal szélén van akkor a bal szomszédja jobb szélén lévő érték.*
 - c. A szimulációhoz készítsen egy grafikus felületet. Készítsen a számok ábrázolásához grafikont is! Legyen lehetőségünk egy szimulált környezetet lépésről lépésre vizsgálni, illetve automatikusan is lehessen végig nézni egy szimulációt. Minden a két esetben egy grafikonon jelezze, hogy az adott állapotban épp mennyi a nyuszi születésének száma.

Minta kezdő állapotra 10x10 –es területen:

X									
		X			X	X	X		
		X				X			
		X		X	X				X
		X					X		
	X				X			X	
			X	X		X			X
X				X			X		
X				X		X		X	
					X				

Minta grafikonra 100. lépés után



6. **Feladat (Moziműsor):** Írjon programot, amely a ceglédi Uránia mozi műsorait elemezi ki egy hét adatai alapján.

- a. Adott egy vagy több szöveges állomány, amit Önnek kell létrehoznia a feladat leírás alapján! Az adatok beolvasása során a beolvasott adatokra adja meg a következő választ!
- b. Lehesse lekérni a Film magyar és angol címét, kategória listáját, hosszát, és a kiadott évét!
- c. Lehesse lekérni a filmeket kategória alapján!
- d. Kiíratásnál jelenítse meg a perc mellett, hogy az hány óra és hány perc.
- e. Lehesse több (legalább két) terem közül választani! Minden teremnél legyen fix férőhely megadva! Legyen megadva a terem hossza és az adott sorban hány szék van!
- f. A termekhez legyenek hozzárendelve filmek egymás után (szünet nélkül)
- g. Lehesse időpontot foglalni a hétre. (csütörtök, péntek, ... kedd, szerda). A foglalásokat addig lehessen kezelni, amíg van szabad hely.
- h. Lehesse lekérni egy adott nap filmjeit!
- i. Lehesse rendezni a filmeket időpont, hossz, név stb. alapján!
- j. Lehesse lekérni, hogy egy adott időpontban van-e éppen futó film, elkezdődött-e már a film.
- k. Lehesse rákeresni megadott filmre, hogy játszó-e a héten!
- l. Az adatokat fájlból olvassa be és fájlba írassa ki!
- m. Hozzon létre egy grafikus felületet a feladat bemutatásához. Használjon táblázatot a megjelenítéshez. A megjelenítésben szabad kezdet kap az ábrázoláshoz! Törekedjen arra, hogy minél több funkcióra minél több információ lekérésére és megjelenítésére lehessen használni!