



Magasszintű programozási nyelvek I.

2023/2024 őszi félév
1. évfolyamdolgozat (A csoport)

Általános információk

Készítsen egy **mappát** a „*C : \temp*” mappába, melynek neve "**A_**" és az **Ön Neptunkódja!** Ebben oldja meg a kijelölt feladatokat! Az egyes projektek nevei a feladatok sorszámait legyenek (01, 02, ...)! Miután a feladatokkal végezett, a teljes mappát **tömörítse be (.zip, .rar, .7z) azonos névvel!**

Pl.: A_IKSI78.zip

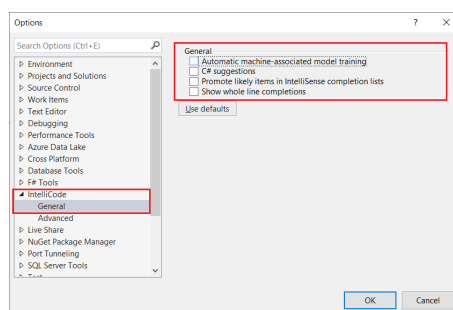
Ezt a fájlt kell feltöltenie a megadott helyre!

A dolgozat írása közben az órai anyagok felhasználása nem megengedett!

A dolgozat írása során nem használhat tömb alapú listát! Nem hívhat meg olyan beépített metódusokat, melyek az alap algoritmusok kiváltására valók, mint pl. *.Sum()*, *.Min()*, *.Max()*, *.Average()*, *.Contains()*, ... Ezen kívül nem használhat olyan metódusokat sem, melyek lambda kifejezésekkel operálnak!

A dolgozat írása közben, amennyiben **Visual Studio 2022** (vagy újabb) fejlesztői környezetet használ, úgy ki kell kapcsolnia az **IntelliCode** összes funkcióját!

Az erre vonatkozó beállításokat a **Tools > Options...** menüpontban találja.



Amennyiben a feladat azt kéri, hogy egy értéket ellenőrzöttén kérjen be, úgy a feladata az, hogy amennyiben a felhasználó hibás értéket ír be, akkor azt kérje be újra addig, amíg az érték megfelelő nem lesz! Ahol ezt a feladat nem kéri, ott feltételezheti, hogy érvényes adat érkezik.

A dolgozat írása közben nem használhat internetes segédanyagokat, és nem kommunikálhat senkivel (beleértve a nagy nyelvi modelleket is)! Bármilyen idegenkezűség, vagy nem megengedett segédanyag használata következtében a dolgozat érdemjegye automatikusan elégtelen. Ilyen esetben a dolgozat újbóli megírására természetesen nincsen lehetőség!

1. Feladat. A nyári időszakban Eger egyik leglátogatottabb helyszíne az Egri Termálfürdő, vagy ahogyan mindenki ismeri, az Egri strand. Az Ön feladata egy olyan alkalmazás implementálása, mely segítségével megállapítható az egyén vagy család napi belépőjének ára. A belépők ára az alábbi táblázatok szerint alakul.

| Egyéni árak | |
|------------------------------------|---------|
| Felnőtt belépő kedvezmények nélkül | 3800 Ft |
| Kedvezményes jegyek | |
| Esti belépő (du. 4 óra után) | 1800 Ft |
| Eger városkártya | 1500 Ft |
| Diák/nyugdíjas | 1000 Ft |

| Családi árak | |
|---|----------|
| 2 felnőtt + 1 gyermek | 9700 Ft |
| 2 felnőtt + 2 gyermek | 12100 Ft |
| Minden további gyermek esetén fizetendő kedvezményes ár | 2400 Ft |

A program működését az alábbiak szerint implementálja!

- a) Kérdezze meg a felhasználótól, milyen típusú jegyet szeretne vásárolni! ("egyéni" vagy "családi")
- b) Egyéni jegy esetén kérdezze meg, van-e valamilyen kedvezményre jogosultsága a felhasználónak! ("igen" vagy "nem")
 - Ha van, kérdezze meg mi a kedvezmény típusa és ez alapján számolja ki a jegy árát! ("városkártya" vagy "diák" vagy "nyugdíjas")
 - Amennyiben nincs, úgy kérdezze meg hány óra van! Oldja meg ellenőrzött adatbekéréssel, hogy ez egy 9 és 19 közötti egész szám legyen, majd ennek tekintetében számolja ki a belépő árát!
- c) Családi jegy esetén kérdezze meg a gyermekek számát! Oldja meg ellenőrzött adatbekéréssel, hogy ez egy 1 és 69 közötti egész szám legyen, majd ennek tekintetében számolja ki a belépő árát!

Példa: Ha a családnak 5 gyermeke van, akkor a fizetendő összeg

$$12100 + 3 \cdot 2400 = 19300 \text{ Ft}$$

- d) Jelenítse meg a képernyőn a fizetendő összeget!

2. Feladat. Egy bonbonokat forgalmazó kimérős bolt számára fejlesztünk alkalmazást. A bonbonokat a felhasználó válogatja össze díszdobozba, melyek mindegyikébe 120 gramm termék helyezhető el. Minden egyes díszdoboz ára 100 Ft. A megvásárolható bonbonokat az alábbi táblázatban foglaltuk össze.

| Megnevezések | Tömeg | Ár |
|---|----------|--------|
| ét, tej, fehér | 10 gramm | 150 Ft |
| mogyorós, diós | 20 gramm | 200 Ft |
| szilvás, pisztáciás, nugátos, ananászos | 30 gramm | 300 Ft |

Extra feladat: Az egyes terméktípusokat tárolja tömbökben a program könnyű bővíthetőségének érdekében, és az egyes feladatokat ezek segítségével oldja meg!

A dobozokat nem csak teljesen megtöltve lehet megvásárolni, viszont egy dobozt mindig teljesen fel kell tölteni, mielőtt új dobozt nyitnánk! Ha egy bonbont beraktunk a dobozba, már nem vehetjük ki!

Készítsen alkalmazást, mely a bonbonok összeválogatását és a végső ár kiszámítását végzi az alábbiak szerint!

- Folyamatosan kérje be a felhasználótól, hogy milyen bonbonokat szeretne megvásárolni!
- A termékek bekérését és a fizetési szándékot a "végeztem" kifejezés beírásával jelzi a felhasználó.
- Jelenítse meg a képernyőn, hogy a felhasználó hány dobozt kezdett meg, és mennyi a fizetendő végösszeg!

3. Feladat. Egy kisebb kínai egyetem kis kampusza számára fejlesztünk alkalmazást, mely egy adott épület termeinek kihasználtságáról tárol adatokat és készít statisztikákat. Az épületben minden emeletén 12 terem található, melyek befogadóképessége 40 fő.

- a) Kérje be a felhasználótól az épület emeleteinek számát! Oldja meg ellenőrzött adatbekéréssel, hogy ez egy 10 és 25 közötti egész szám legyen!
- b) Az egyetemen 15 fő alatt nem indítanak kurzust. Hozzon létre egy egydimenziós tömböt, mely képes eltárolni minden emelet minden termének létszámadatait! Töltse fel ezt a tömböt 15 és 40 közötti egész értékekkel!
- c) Számítsa ki és jelenítse meg a képernyőn százalékos formában, hogy mekkora az épület kihasználtsága a férőhelyek tekintetében!
- d) Állapítsa meg és jelenítse meg a képernyőn, van-e olyan terem, mely minimális kihasználtsággal működik (15 fő)!
- e) A további feladatokat az emeletek szerinti csoportosítással kell megoldania!
 - i) Mekkora az egyes szinteken a legmagasabb és legalacsonyabb kihasználtságú termek közötti különbség?
 - ii) Van-e olyan emelet, mely maximális kihasználtságú?