Áll az alku - C#

# Áll az alku konzolos játék

A program célja, hogy jól ismert televíziós vetélkedő játékot megvalósítsa egy számítógépen is játszható alkalmazás formájában.

A játékban 23 egyforma, de különböző azonosítóval (1..23) ellátott táska vesz részt, melyekben különböző összegek foglalnak helyet. Ezeket az összegeket a játékos a játék előtt nem ismeri.

A különböző nyeremények növekvő sorrendben:

1; 5; 10; 100; 500; 1000; 5000; 10000; 25000; 50000; 80000; 150000; 300000; 500000; 800000; 1000000; 2500000; 5000000; 7000000; 10000000; 15000000; 20000000; 50000000

A játék úgy indul, hogy a játékos választ egy neki szimpatikus táskát, amelyet nem nyithat ki a játék végéig.

Ezek után a játék körökre lesz osztva, első körbe 5, majd háromszor 3, aztán újból háromszor 2, táskát kell nyitnia (körönként), majd a maradék táskákat egyesével.

Minden kör végén a Bank igazgatója tesz egy ajánlatot a még játékban lévő táskákban lévő összegek alapján. A játékos elfogadhatja ezt az ajánlatot, ilyenkor ezt az összeget nyeri meg. Ha játékos nem fogadja el az ajánlatot, akkor újabb körben nyithat táskákat, amelynek végén ismét a bank ajánlata következik.

Ha játékos egyszer sem fogadja el a bank ajánlatát és már minden táskát kinyitott (a sajátját kivéve), akkor utolsóként a saját táskáját nyitja ki, ezzel megnyerve a benne lévő összeget.

# 1, Adatmodell – mit és hogyan tárol el a program

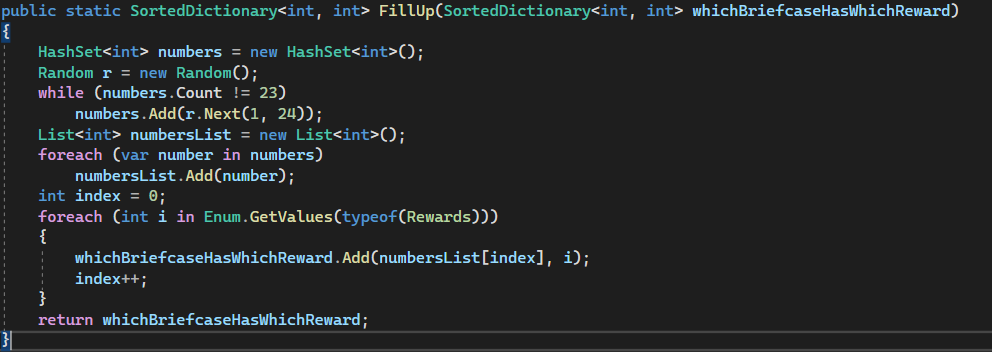
A program futáskor létrehoz egy SortedDictionary-t, ami azt tárolja sorrendben, hogy melyik azonosítójú táskában mennyi nyeremény van, és ezt aztán feltölti

A képen szöveg látható

Automatikusan generált leírás

Enum, itt tárolódnak a nyeremények

Táskák feltöltése



HashSet, hogy az enumbol kiválasztott generált random számok ne ismétlődjenek (ne legyen mondjuk 5 ft-ból 3 darab, és emiatt 2 nyeremény kimarad), majd a HashSet feltöltése random számokkal, ezután a HashSet listává alakítása, hogy lehessen az elemeket indexelni

ForEach ciklushoz indexelő

ForEach ciklus végig megy az Enumon, majd a SortedDictionary-ba berakja Key-ként a táska azonosítóját, Value-ként pedig egy véletlenszerűen generált nyereményt, majd az indexelő növelése

Majd visszaadja az elkészült SortedDictionaryt

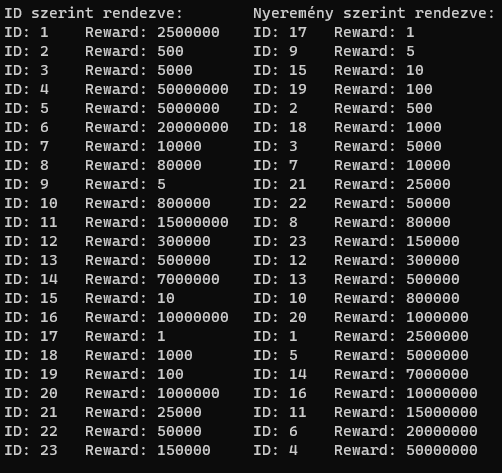
(Próbáltam sima Dictionary-vel, de azzal az a problémák keletkeztek, hogy például a 3-as azonosítójú táska a 7. helyen volt, ezt a SortedDictionary megjavította)

# 2, Vezérlési folyamat - a program indulásától kezdve egészen a végéig mi fog történni? (folyamatábra)

SortedDictionary<int, int> briefcases = new SortedDictionary<int, int>();

Game.FillUp(briefcases);

A táskákba belenézhetünk a Game.Inspect(briefcases); segítségével



A felhasználót megkérjük, válasszon egy táskát

Console.Write("Válassz egy táskát: ");

int selectedBriefcase = int.Parse(Console.ReadLine())