

A feladat megoldására és benyújtására 120 perc áll rendelkezésre. A létrehozott projekt megnevezése tartalmazza az Ön nevét, Neptun kódját és a feladatsor csoportját (pl. `AliceBob_ABC123_X`). A megoldását tartalmazó mappát (a teljes solution-t) tömörítve, a `http://zh.nik.lan` címen elérhető felületen keresztül nyújtsa be.

Ügyeljen a fordítási hibától mentes kódra, ellenkező esetben a megoldás nem értékelhető. A megoldáshoz ne használjon LINQ függvényeket.

Készítsen egy alkalmazást a Weyland–Yutani társaság számára, amely segítségével a vállalat lekérdezheti az űrhajóinak aktuális adatait.

1 Hozza létre az alábbi felsorolt típusokat (enumokat). (2 pont)

1. `ShipClassType`: az űrhajók lehetséges osztályai (kategóriái).
`Cargo, Military, Research, Mining, Colonial, Rescue`
2. `CrewStatusType`: a legénység állapotának lehetséges értékei a megadott konstansokkal.
`Active=1, InCryosleep=2, MissingInAction=3, Deceased=4`

2 `Spaceship` osztály: egy űrhajót reprezentál, és az alábbi tagokkal rendelkezik.

- Tárolja az űrhajó nevét (`string`), osztályát (`ShipClassType`), a legénység létszámát (`int`) és állapotát (`CrewStatusType`), a hajó rakományának tömegét tonnában (`int`), valamint a hajótól érkező legutóbbi üzenet dátumát (`DateTime`) egy-egy privát mezőben. (3 pont)
- Minden mezőhöz készítsen egy-egy csak lekérdezhető tulajdonságot. (3 pont)
- Az osztály rendelkezzen egy konstruktorral, amely egyetlen paraméterként az alábbival egyező formátumú karakterláncot vár.

```
USCSS Nostromo;Cargo;7;InCryosleep;20000;2122-06-29
```

A karakterlánc az űrhajó adatait tartalmazza pontosvesszővel elválasztva abban a sorrendben, ahogyan az előző részfeladatban szerepelt. A konstruktor dolgozza fel a karakterláncot, és végezze el a mezők értékadását. (5 pont)

- Rendelkezzen egy `int DaysSinceLastMessage(DateTime date)` szignatúrájú privát metódussal, amely a paraméterben megadott dátumhoz viszonyítva az utolsó üzenet óta eltelt napok számával tér vissza. (2 pont)
- Rendelkezzen egy `bool NeedsRescue(DateTime date)` szignatúrájú publikus metódussal, amely meghatározza, hogy szükséges-e mentőhajót indítani a rakományért. Erre akkor van szükség, ha a legénység állapota `MissingInAction` vagy `Deceased`, vagy amennyiben `Active` állapot esetén 30 napnál, `InCryosleep` állapot esetén 3650 napnál (10 évnél) régebben érkezett utoljára üzenet a hajótól. (5 pont)
- Írja felül a `ToString` metódust úgy, hogy az a hajó nevét és osztályát adja vissza az alábbi formátumban. (2 pont)

```
USCSS Nostromo (Cargo)
```

- Rendelkezzen egy `string GetStatusReport(DateTime date)` szignatúrájú publikus metódussal, amely az űrhajó adatait adja vissza az alábbi formátum szerint.

```
=== USSS Nostromo (Cargo) ===  
Crew: 7 (InCryosleep)  
Last message: 23085 days  
Rescue needed: Yes
```

A szöveg az űrhajó nevét és osztályát, a legénység létszámát és állapotát, illetve az utolsó üzenet óta eltelt napok számát, valamint azt tartalmazza, hogy szükséges-e mentőhajót indítani a rakományért. (5 pont)

3 FleetHandler osztály: a vállalat űrhajóinak tárolását és kezelését végző osztály.

- Tárolja az űrhajókat egy tömb vagy lista típusú privát mezőben. (1 pont)
- Tárolja az aktuális¹ dátumot egy privát mezőben. (1 pont)
- Az osztály konstruktora egy fájl nevét várja paraméterként. A fájl egyes sorai (az elsőt kivéve, amely az aktuális dátum) az előző részfeladatban megadott formátumú sorokat tartalmaznak, mindegyik sor egy-egy űrhajó adatait. Töltse be és dolgozza fel a fájlt, a benne lévő adatok alapján állítsa be az aktuális dátumot, és készítse el az űrhajókat reprezentáló példányokat, amelyeket helyezzen el az előbbi tömbben vagy listában. (5 pont)
- Készítsen egy `int TotalShipCount` csak lekérdezhető publikus tulajdonságot, amely visszaadja a flottát alkotó hajók darabszámát. (1 pont)
- Készítsen egy `bool HasAnyDeceasedCrew` csak lekérdezhető publikus tulajdonságot, amely visszaadja, hogy van-e olyan hajó, amelyen a legénység elhunyt (Deceased). (4 pont)
- Készítsen egy `double AverageCargo(ShipClassType shipClass)` szignatúrájú publikus metódust, amely paraméterként egy hajóosztályt vár, majd visszaadja az ilyen űrhajók rakományának átlagos tömegét. (6 pont)
- Készítsen egy `Spaceship[] GetShipsGroupByRisk()` szignatúrájú publikus metódust, amely az összes űrhajót egy tömbben adja vissza úgy, hogy a tömb elejére válogatja a magas kockázatú hajókat, a végére pedig a többi hajót. Magas kockázatú a hajó, ha az Cargo vagy Research osztályú és kimentést igényel. (8 pont)
- Készítsen egy `string GetLargestCrewData(CrewStatusType status)` szignatúrájú publikus metódust, amely visszaadja annak a hajónak az adatait (lásd `GetStatusReport`), amelyen a megadott állapotú legénységek közül maximális nagyságú teljesít szolgálatot². Ha nincs a megadott állapotnak megfelelő legénység, térjen vissza egy üres karakterlánccal. (8 pont)

¹A szempontunkból ez egy jövőbeli dátum.

²Ha több maximális méretű legénységgel rendelkező űrhajó van a flottában, válassza a legelső ilyen.

- Készítsen egy `void GenerateReport()` szignatúrájú publikus metódust, amely az alábbi adatokat jeleníti meg a képernyőn: *(9 pont)*
 - aktuális dátum,
 - a flotta mérete (`TotalShipCount`),
 - átlagos rakomány hajóosztályonként (`AverageCargo`),
 - az összes hajó adata kockázati szintek alapján csoportosítva (`GetShipsGroupByRisk`).

Példa a metódus által előállított kimenet formátumára (részlet).

```
Current date is 2185. 09. 11. 0:00:00.
```

```
Total ship count is 17.
```

```
Average cargo by ship types:
```

```
  Cargo: 13340000
```

```
  Military: 3500
```

```
  Research: 6000
```

```
  Mining: 27666666,666666668
```

```
  Colonial: 11333,333333333334
```

```
  Rescue: 750
```

```
Detailed info sorted by risk:
```

```
=== USSCSS Nostromo (Cargo) ===
```

```
  Crew: 7 (InCryosleep)
```

```
  Last message: 23085 days
```

```
  Rescue needed: Yes
```

```
=== USSCSS Prometheus (Research) ===
```

```
  Crew: 17 (Deceased)
```

```
  Last message: 33497 days
```

```
  Rescue needed: Yes
```

```
...
```