

# Rendszámfelismerő rendszer

## Fejlesztői dokumentáció

### Struktúrák:

Az „SKarakter” egy adott rendszámkarakter képét tartalmazza.

Az „SJarmu” egy adott járműhöz tartozó adatokat tartalmazza (*rendszám, forgalmi lejárati dátum, autó szín, tulajdonos név, tulajdonos címe felbontva helységre és címre*).

### Globális változók:

`map<char, SKarakter>` karakterek;

*A rendszámkarakterekhez tartozó képeket tárolja.*

`vector<string>` rendszamok;

*A képalkotó rendszertől kapott képekről leolvasott rendszámokat tárolja.*

`vector<SJarmu>` jarmuvek;

*Az autók adatait tárolja.*

`vector<string>` korozottek;

*Az eltűnés/lopás miatt körözött autók rendszámát tárolja.*

### Függvények:

`bool` karakterBeolvasas(`const string` &filename)

filename – bemeneti fájl neve

A rendszámkarakterekhez tartozó képeket olvassa be.

A bemeneti fájl formátuma megegyezik a feladathoz mellékelte alapbetu.txt formátumával.

`bool` kotojelBeolvas(`const string` &filename)

filename – bemeneti fájl neve

A kötőjel karakterhez tartozó kép beolvasása.

A függvényre a vesszo.txt eltérő formátuma miatt van szükség.

`bool` rendszamokBeolvasasa()

10.txt – 29.txt beolvasása, a rendszámok leolvasása és tárolása a rendszamoktombben.

`bool` korozottBeolvasas(`const string` &filename)

filename – bemeneti fájl neve

Körözött járművek rendszámainak beolvasása és tárolása a „korozottek” tárolóban.

`bool` jarmuBeolvasas(`const string` &filename)

filename – bemeneti fájl neve

A járművek adatainak beolvasása és tárolása a „jarmuvek” tárolóban.

`string` rendszamLeolvasas(`const vector<string>` &foto)

foto- A képalkotó rendszertől kapott szöveges rendszámfotó  
Rendszám leolvasása fotóról.  
Visszatérési érték: a rendszám stringként, illetve üres string, amennyiben a leolvasás sikertelen volt.

`SJarmu` \*jarmuKeres(`const string` &rendszam)

Jármű adatainak megkeresése a „jarmuvek” tárolóban rendszám alapján.  
Visszatérési érték: mutató a jármű adataira, illetve `nullptr`, amennyiben a rendszámhoz nem található jármű.

### Részfeladatok megoldására felhasznált függvények:

`void` osszesites(`const string` &filename)

filename- kimeneti fájl neve  
Az ellenőrzések eredményeinek fájlba íratása.

`void` razzia()

Azon helységnevek kiírása, amelyekhez a legtöbb körözött autó tartozik.

`void` biztosSzinek()

A körözött autók színeinek százalékos megoszlásának kiírása.

### Egyéb függvények:

`void` fejlec(`const string` &str)

Fejléc előállítás az `str` szövegből és `#` karakterekből.

`void` tovabb()

A program megállítása, amíg a felhasználó tovább nem akar lépni.

`int` elteltNapok(`string` datum1, `string` datum2)

datum1 és datum2- `eeee.hh.nn` formátumú dátumok  
Két dátum között eltelt napok számának kiszámítása

`string` maidatum()

Visszatérési érték: a mai dátum `eeee.hh.nn` formátumban.

`int` main()

Adatok beolvasása és feldolgozása a fenti függvényekkel. Az egyes feladatrészeket a „fejlec” és a „tovabb” függvény választja el.