

Autószerviz nagyházi félkész fejlesztői dokumentáció

Készítette: Anderlik Sándor

A program egy konzolos menüvel vezérelhető. Képes megjeleníteni az összes nyilvántartott autót, lehetőség van rendszám alapján keresni az autók között, és törölni is tudjuk őket. A program minden egyes **Kilépés** menüpont választáskor fájlba menti az adatokat. Az új autó felvétele is működik kisebb-nagyobb hibákkal, de még nem teljesen kész.

A menü kinézete:

```
(1) Auto felvetele
(2) Auto torlese
(3) Kereses rendszam alapjan
(4) Javitas felvetele
(5) Szerviztortenet
(6) Hamarosan lejaro muszaki vizsga
(7) Osszes auto listazasa
(x) Kilepes
Valasztott menupont szama:
```

Az adatokat láncolt lista segítségével tárolom, és az alábbi adatstruktúrákat használom egyelőre:

```
typedef struct Ugyfel{
    char* vezeteknev;
    char* keresztnév;
    char* telefonszam;
} Ugyfel;
typedef struct Auto{
    char* rendszám;
    char* marka;
    char* típus;
    int évjarat;
    char* muszaki;
    Ugyfel tulaj;
} Auto;

typedef struct Bejegyzes{
    Auto adat;
    struct Bejegyzes* kov;
} Bejegyzes;
```

Függvények:

`char*` beolvas(`char*` str);

A függvény dinamikusan foglalva olvas be szöveget a felhasználótól.

`Bejegyzes*` bejegyzes_felvetele(`Bejegyzes*` eleje, `Auto` uj_auto);

A függvény hozzáad egy Autó típusú elemet a láncolt lista végéhez.

`Bejegyzes*` bejegyzesTorlese(`Bejegyzes*` eleje);

A függvény bekér egy rendszámot, majd ha létezik elem ilyen rendszámmal akkor megpróbálja

törölni azt. Megerősítést kér a felhasználótól, ha megkapja akkor törlődik a láncolt lista adott eleme.

`void AutokListazasa(Bejegyzes* szerviz);`
Kiírja a láncolt listában tárolt összes autót soronként

`void SzervizMentes(Bejegyzes* szerviz);`
Egy fájlba menti a láncolt lista tartalmát soronként.

`Auto* kereses(Bejegyzes* szerviz);`
Bekér egy rendszámot, majd megkeresi az adott rendszámmal rendelkező autót. Ha megtalálja kiírja az autó adatait, különben pedig jelzi, hogy nincs ilyen autó a nyilvántartásban.

`void uj_auto_felvetele(Bejegyzes* szerviz);`
Bekéri az új autó adatait a felhasználótól és hozzáadja a láncolt listához. (Még nem hibátlan)

`void listaFelszabaditas(Bejegyzes* szerviz);`
Felszabadítja a láncolt lista dinamikusán foglalt memóriáit.

`void menuMegjelenites()`
Kiírja a menüpontokat a képernyőre.

`void menu(Bejegyzes* szerviz);`
Ez a függvény vezérli a menüt. Lekezeleli a felhasználótól érkező bemeneteket és felel azért, hogy a megfelelő menüpont akciója hajtsódjon végre. Képes kezelni 7 menüpontot, az érvénytelen bemenetet is jelzi, valamint a programot is betudja zárni (akkor történik az adatok fájlba mentése).