

Gráf tárolása

A programban egyidejűleg egy gráfot kezelünk. Ezt szomszédsági listával tároljuk el, tehát minden csúcsához a gráfnak eltároljuk, melyik csúccsal szomszédos, és milyen súllyal. A csúcsokat is egy listában tároljuk. Betölthetünk a programba egy gráfot fájlból, melynek az első sora tartalmazza szóközzel elválasztva a csúcsok és az élek számát. A további sorokban egy-egy él adatai vannak, <csúcs1> <csúcs2> <súly> formátumban. A gráf-építő segítségével módosíthatjuk az aktuálisan betöltött gráfot.

Adatszerkezet

Graf - fő adatszerkezet, csúcsok listáját és számát tartalmazza

```
typedef struct Graf {  
    CsucsLista *csucsok;  
    int csucsok_szama;  
} Graf;
```

CsucsLista - duplán láncolt lista, csúcsokat tartalmazza

```
typedef struct CsucsLista {  
    struct CsucsListaElem *első, *utolsó;  
} CsucsLista;  
typedef struct CsucsListaElem {  
    Csucs *csucs;  
    struct CsucsListaElem *előzo, *kov;  
} CsucsListaElem;
```

Csucs - egy csúcs sorszámát és szomszédsági listáját tartalmazza

```
typedef struct Csucs {  
    int id;  
    SzomszedsagiLista *szomszedok;  
} Csucs;
```

SzomszedsagiLista - adott csúcs szomszédjainak listája

```
typedef struct SzomszedsagiLista {  
    struct SzomszedsagiListaElem *első, *utolsó;  
} SzomszedsagiLista;  
typedef struct SzomszedsagiListaElem {  
    int csucs;  
    int suly;  
    struct SzomszedsagiListaElem *előzo, *kov;  
} SzomszedsagiListaElem;
```

main.c

```
void graf_epito(Graf *g)
```

Gráf-építő működéséért felel, g gráfot módosítja.

megjelenites.c

```
void menu_kiiras()
```

Kiírja a menüt.

listak.c

```
SzomszedsagiLista *beszur_szomszedsagi(SzomszedsagiLista *l, int csucs, int suly)
```

Paraméterként kapott listába beszúr egy új elemet, visszatér a listára mutató pointerrel.

```
SzomszedsagiListaElem *keres_szomszedsagi(SzomszedsagiLista *l, int csucs)
```

Paraméterként kapott listában keres elemet, visszatér az elemre mutató pointerrel.

```
void torol_szomszedsagi(SzomszedsagiLista *l, int csucs)
```

Paraméterként kapott listában törli a megadott elemet.

```
void felszabadit_szomszedsagi(SzomszedsagiLista *l)
```

Paraméterként kapott dinamikusan foglalt listát felszabadítja.

```
CsucsLista *beszur_csucs(CsucsLista *l, Csucs *csucs)
```

Paraméterként kapott listába beszúr egy új elemet, visszatér a listára mutató pointerrel.

```
CsucsListaElem *keres_csucs(CsucsLista *l, int csucs)
```

Paraméterként kapott listában keres elemet, visszatér az elemre mutató pointerrel.

```
void torol_csucs(CsucsLista *l, int csucs)
```

Paraméterként kapott listában törli a megadott elemet.

```
void felszabadit_csucs(CsucsLista *l)
```

Paraméterként kapott dinamikusan foglalt listát felszabadítja.

grafkezeles.c

```
void csucs_hozzaad(Graf *g, int id)
```

Paraméterként kapott gráfban létrehoz egy új csúcsot.

```
void csucs_torol(Graf *g, int id)
```

TODO: Paraméterként kapott gráfban törli a megadott csúcsot és az összes hozzátartozó élt.

```
void el_torol(Graf *g, int c1, int c2)
```

Paraméterként kapott gráfban törli a megadott élt (mindkét csúcs szomszedsági listájából kiveszi az adott csúcs sorszámát).

```
void el_modosit(Graf *g, int c1, int c2, int s)
```

Paraméterként kapott gráfban módosítja az adott él súlyát.

```
void el_letrehoz(Graf *g, int c1, int c2, int s)
```

Paraméterként kapott gráfban létrehoz a megadott csúcsok között egy élt, a megfelelő súllyal.

```
void szelessegi(Graf *g, int id)
```

TODO: Szélességi bejárással végigmegy a gráfon a kiindulóponttól, az útvonalat fájlba menti.

```
void melysegi(Graf *g, int id)
```

TODO: Mélységi bejárással végigmegy a gráfon a kiindulóponttól, az útvonalat fájlba menti.

```
void dijkstra(Graf *g, int c1, int c2)
```

TODO: Dijkstra algoritmusával megkeresi a gráfban a két pont közötti legrövidebb útvonalat, majd fájlba menti az eredményt.

fajlkezeles.c

```
void betolt_graf(Graf *g, char* fajlnev)
```

Beolvassa a paraméterként kapott gráfba a megadott fájl által meghatározott gráfot.

```
void ment_graf(Graf *g, char* fajlnev)
```

TODO: Megadott fájlba menti a paraméterként kapott gráfot.