

Felhasználói dokumentáció

Balog Ádám Márk (ELAO0E)

May 2020

1. Osztálykönyvtár célja

A felhasználó számára biztosít egy racionális számokat tároló, azokkal számoló, és egy lineáris egyenlőtlenségrendszer adott célfüggvény mellett maximalizáló/minimalizáló osztályt, nem negatív változók mellett. Lásd:

https://en.wikipedia.org/wiki/Linear_programming

2. Osztályok használata

A felhasználó az általa írt programban meghívja a osztálykönyvtárat a következő sorral a fejlécben:

```
#include "simplex.h"
```

Ezután lehetősége nyílik a következő osztályok, és azok tagfüggvényeinek használatára:

2.1. class Rac

2.1.1. private: (ezeket közvetlenül nem tudja hívni/elérni)

- **int** sz;
- **int** n;
- **void** simplify();

2.1.2. public:

- **Rac**(int sz, int n);
- **int** lnko(int x, int y);
- **int** getsz()const;
- **int** getn()const;

- **void** setsz(int a);
- **void** setn(int a);
- **void** print()const;
- **const Rac** operator+(const Rac& x)const;
- **const Rac&** operator+=(const Rac& x);
- **const bool** operator==(const Rac& x) const;
- **const Rac** operator-(const Rac& x)const;
- **const bool** operator!=(const Rac& x)const;
- **operator double**()const;
- **Rac** reciprok();
- **const Rac** operator*(const Rac& x)const;
- **const Rac** operator*=(const Rac& x);
- **const Rac** operator/(Rac& x)const;

2.2. class Simplex

2.2.1. private: (ezeket közvetlenül nem tudja hívni/elérni)

- **vector<vector<Rac>>** A;
- **vector<vector<Rac>>** PivotT;
- **vector<Rac>** b;
- **vector<Rac>** Z;
- **vector<int>** egyenlo;
- **vector<int>** nagyobb;
- **vector <int>** mmek;
- **bool** minn;
- **double** ertek;
- **void** korlell();
- **void** PivotTk();
- **void** printpivot();
- **const unsigned int** Pivotelem(const int x)const;
- **void** pivotalas(const unsigned int x,const unsigned int y);

2.2.2. public:

- Simplex();
- ~Simplex();
- void filebol(const std::string& hely);
- void insert_cond(const std::string& szov);
- void insert_Z(const std::string& szov);
- void Aprint();
- void bprint();
- void Zprint();
- void printmeg();
- void solve();
- const double geter();

2.3. Fontos tudnivalók:

Adott racionális szám létrehozása során alapértelmezetten a 0/1 számot tároljuk, ezt a "setn/sz" metódusokkal lehet állítani. A Simplex osztály példányostása során egy üres feladatot kapunk. A feladatot lehet soronként a programkódunkban bevinni, vagy txt fájlból beolvasni, melyeknek formai követelményei:

- Soronkénti bevitel: Csak az együtthatókat kell egymás után írni, ám a dimenzió egyezés érdekében a nullákat is ki kell írni. A tizedesvesszőt "." jelöli. Ezután következik a reláció, mely lehet "<=", ">=" vagy "=" (szigorú egyenlőtlenségeket nem vizsgálunk, zárt poliédereket követelünk meg). Ezután még vár egy számot, mely a b vektor eleme lesz. Célfüggvény-bevitel esetén az eljárás hasonló: max/min majd utána sorra a megfelelő előjelű együtthatók.

```
Simplex proba;  
proba.insert_cond("1 0.3 0.5 -1 3 0 <= 100");  
proba.insert_cond("5 3 2 0 0 2 = 650");  
proba.insert_cond("1 0.5 1 3 2 1 <= 150");  
proba.insert_Z("max 100 160 250 0 15 30");  
proba.solve();  
std::cout<<proba.geter();
```

1. ábra. Példa egy soronkénti feltétel megadásra

```

1 1 0.3 0.5 -1 3 0 = 100
2 5 3 2 0 0 2 = 650
3 1 0.5 1 3 2 1 = 150
4 min 100 160 250 0 15 30

```

2. ábra. Példa fájlból való feltétel megadásra

- Fájlból bevitel: Megadjuk a txt fájl helyét, ahonnan a függvény automatikusan kiolvassa a feltételeket, majd végül a célfüggvényt. Fontos itt is: Minden sorban ugyan annyi elem legyen, és a célfüggvény az utolsó sorba kerüljön. Ezután már nem fogja az adatokat beolvasni.

Miután felvittük a feltételeket ÉS a célfüggvényt, a `.solve()` metódussal lehet megoldatni a feladatot. Figyeljünk arra, hogy a felvitt feltételek jobb oldala ne legyen negatív (szorozzuk be -1-gyel, ha így járunk). Ha szükségünk van az optimum értékre (doubleben) a `geter()` tagfüggvény áll rendelkezésünkre. Közben bármikor kiíratjuk az együtthatómátrixot, eredményvektort és a célfüggvény vektort az osztályunk által kínált különböző `print()` függvények segítségével.