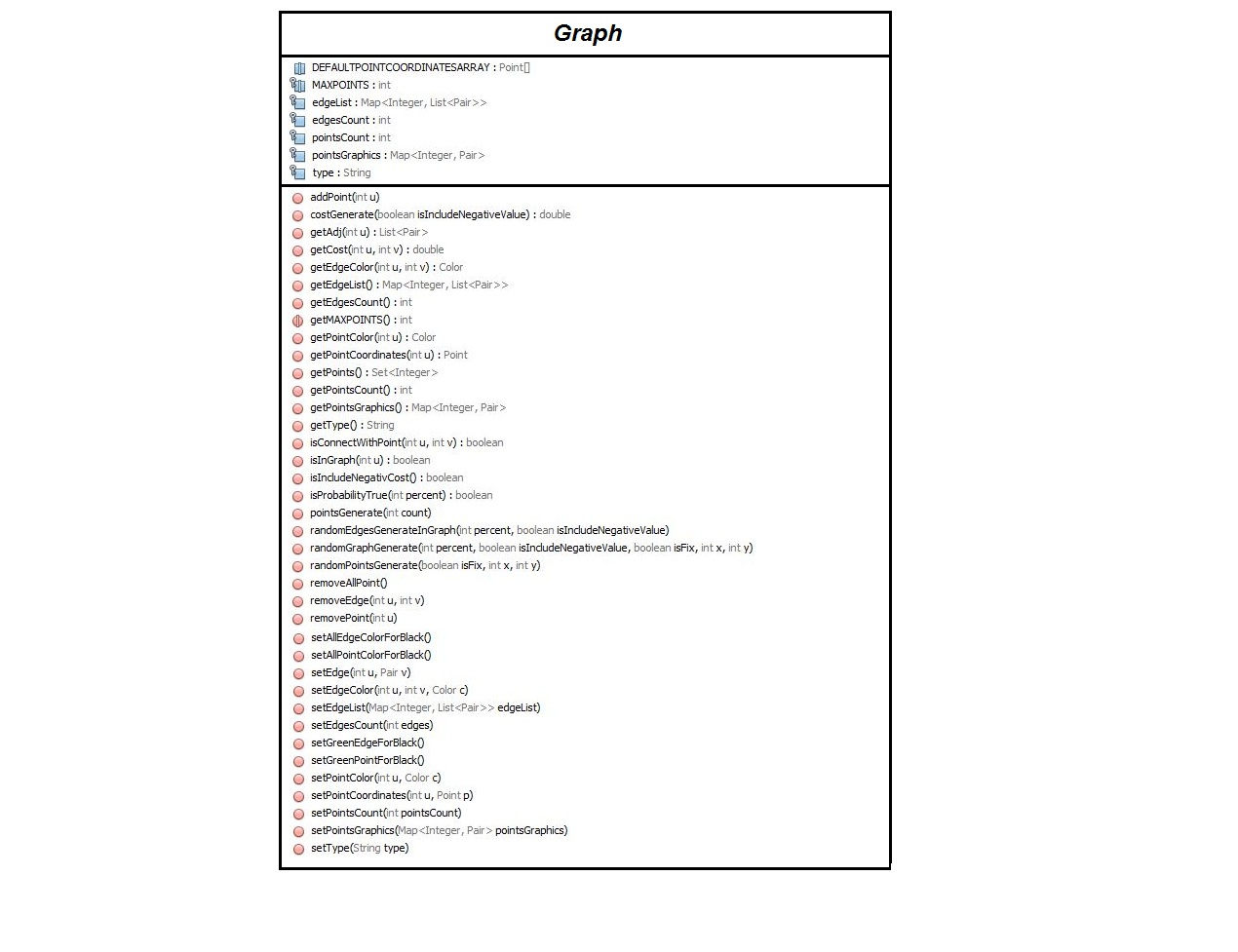
### Gráfosztályok

#### *Graph.java*

Gráfot megvalósító absztrakt osztály, melyből a speciális gráfok származnak, a mely megvalósítja a Serializable interfészt, hogy a gráfot fájlban tárolhassuk.



Változók

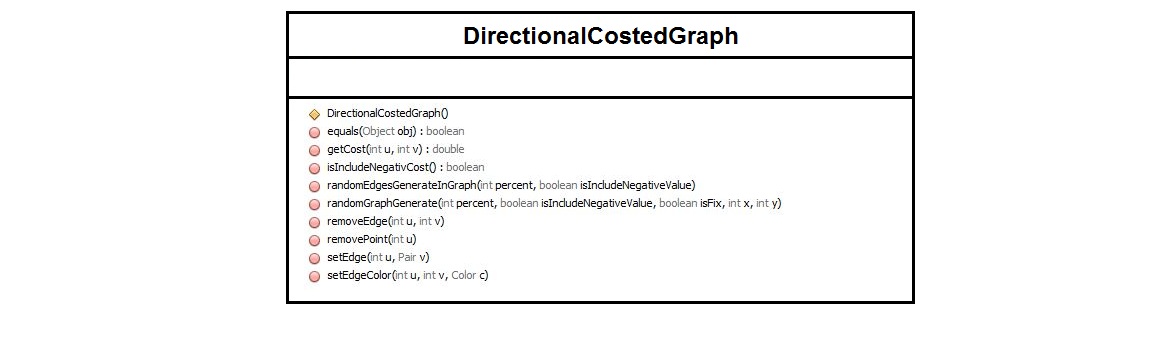
|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Funkció** |
| DEFAULTPOINTCOORDINATESARRAY | java.awt.Point[]  (final static) | A gráf csúcsainak alapértelmezett koordinátái |
| MAXPOINTS | int (final static) | Maximális csúcsok számát tartalmazó változó |
| edgeList | java.util.Map<Integer, java.util.List<Pair>> | Gráf éllistáját tartalmazó változó, amelyet szótár típusú gyűjteményben tárol az osztály, kulcsértéke a kezdőcsúcs sorszáma, értéke a kezdőcsúcsból kiinduló élek (végcsúcs, ha súlyozott a gráf, akkor az él súlyát is tartalmazza, valamint az él színét) |
| edgesCount | int | Gráfban levő élek számát tartalmazza |
| pointsCount | int | Gráfban levő csúcsok számát tartalmazza |
| pointsGraphic | java.util.Map<Integer, Pair> | Gráf grafikai adatait tartalmazó változó, amelyet szótárt típusú gyűjteményben tárol az osztály, kulcsértéke a gráf csúcsának sorszáma, értéke a csúcshoz tartozó rekord, mely tartalmazza a csúcs koordinátáját és a színét |
| type | String | Gráf típusát tartalmazó változó |

Függvények

|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Funkció** |
| addPoint(int u) | void | A paraméterben megadott sorszámmal egy új csúcsot ad a függvény a gráfhoz, növelve 1-gyel a csúcsok számát tartalmazó változót és bejegyezve az éllistába valamint a csúcsok grafikai adatait tartalmazó változóba, már ha nincs benne a megadott sorszámú csúcs a gráfban. |
| costGenerate(boolean isIncludeNegativeValue) | double | Súlygeneráló függvény, amely egy valós számot generál, a megadott paraméter szerint, ha igaz, akkor ez a valós szám lehet negatív [-5,20) intervallumból választ ekkor, különben nem lehet negatív, [1..15] intervallumból választ ekkor. |
| getAdj(int u) | java.util.List<Pair> | A paraméterben megadott sorszámú csúcs szomszéd csúcsainak a listáját visszaadó függvény. |
| *getCost(int u, int v)* | double | A paraméterben megadott él (kezdőcsúcs és végcsúcs) súlyát visszaadó absztrakt függvény, már ha súlyozott a gráf, ezért más-más módon valósítják meg a különböző gráfok. |
| getEdgeColor(int u, int v) | java.awt.Color | A paraméterben megadott él (kezdőcsúccsal és végcsúccsal) színét visszaadó függvény, már ha létezik az él. |
| getEdgeList() | java.util.Map<Integer, List<Pair>> | A gráf éllistáját visszaadó függvény. |
| getEdgesCount() | int | A gráf éleinek a számát visszaadó függvény. |
| getMAXPOINTS() | int | A gráf maximális csúcsainak a számával visszatérő függvény. |
| getPointColor(int u) | java.awt.Color | A paraméterben megadott csúcs színét visszaadó függvény, már ha a csúcs benne van a gráfban. |
| getPointCoordinates(int u) | java.awt.Point | A paraméterben megadott csúcs koordinátáit visszaadó függvény. |
| getPoints() | java.util.Set<Integer> | A gráf csúcsainak a sorszámát egy egész szám típusú halmazzal visszaadó függvény. |
| getPointsCount() | int | A gráf csúcsainak a számát visszaadó függvény. |
| getPointsGraphics() | java.util.Map<Integer, Pair> | A gráf grafikai adatait tartalmazó változót visszaadó függvény. |
| getType() | String | A gráf típusát visszaadó függvény. |
| isConnectWithPoint(int u, int v) | boolean | Igaz vagy hamissal visszatérő függvény, aszerint, hogy a paraméterben megadott két csúcs között fut-e él vagy sem, másképpen benne van-e a gráfban a paraméterben megadott él. |
| isInGraph(int u) | boolean | Igaz vagy hamissal visszatérő függvény, aszerint, hogy a paraméterben megadott csúcs benne van-e a gráfban vagy sem |
| *isIncludeNegativeCost()* | boolean | Az absztrakt függvény igazzal vagy hamissal tér vissza aszerint, hogy tartalmaz-e a gráf negatív költséget vagy sem, már ha súlyozott a gráf, ezért másképp valósítják meg a különböző gráf típusok. |
| isProbabilityTrue(int percent) | boolean | A függvény generál egy számot 0 és 100 között véletlenszerűen, ha 0 és a paraméterben megadott szám közé esik, vagy egyenlő a generált érték, akkor igazzal különben hamissal tér vissza a függvény, másképpen a paraméterben megadott valószínűséggel generál a függvény igazat vagy hamisat. |
| pointsGenerate(int count) | void | A paraméterben megadott számú csúcsot ad hozzá a gráfhoz a függvény. |
| randomEdgesGenerateInGraph(int percent, boolean isIncludeNegativeCost) | void | Az absztrakt függvény más-más megvalósítással a gráf típusa szerint generál véletlenszerűen éleket a megadott paraméterek alapján. |
| *randomGraphGenerate(int percent, boolean isIncludeNegativeCost, boolean isFix, int x, int y)* | void | Az absztrakt függvény más-más megvalósítással a gráf típusa szerint generál véletlenszerűen egy gráfot a megadott paraméterek alapján. |
| randomPointsGenerate(boolean isFix, int x, int y) | void | Csúcsok generálása, aszerint, hogy az első paraméter igaz vagy hamis. Ha igaz, akkor a függvény fixen annyi pontot generál a gráfnak, amennyi a harmadik paraméter. Ha hamis akkor a függvény második és harmadik paraméterben megadott egész számok közötti intervallumból választ egy számot véletlenszerűen és generál annyi csúcsot a gráfnak, amennyi a választott szám volt. |
| removeAllPoint() | void | Az összes pont törlését elvégző függvény. |
| *removeEdge(int u, int v)* | void | A paraméterben megadott él (kezdőcsúcs és végcsúcs) törlését elvégző absztrakt függvény más-más megvalósítással, aszerint, hogy milyen típusú a gráf. |
| removePoint(int u) | void | A paraméterben megadott csúcs törlését elvégző absztrakt függvény más-más megvalósítással, aszerint, hogy milyen típusú a gráf. |
| setAllEdgeColorForBlack() | void | Az összes él színét feketére állító függvény, az éllistába is bejegyezve. |
| setAllPointColorForBlack() | void | Az összes csúcs színét feketére állító függvény a grafikai adatokat tartalmazó gyűjteményben rögzítve |
| *setEdge(int u, Pair v)* | void | Két csúcs közötti él beállítását elvégző absztrakt függvény, a paraméterben megadott kezdőcsúccsal és végcsúccsal, valamint súllyal, ha súlyozott a gráf, más-más megvalósítással, a szerint, amit az adott típusú gráf megkövetel. |
| *setEdgeColor(int u, int v, java.awt.Color c)* | void | A paraméterben megadott él színének beállítása a paraméterben megadott színnel a gráfok típusai szerint más-más megvalósítással elvégző absztrakt függvény |
| setEdgesCount(int edges) | void | Élek számát beállító függvény a paraméterben megadott számmal |
| setGreenEdgeForBlack() | void | A zöld élek színének feketére állítását elvégző függvény |
| setGreenPointForBlack() | void | A zöld csúcsok színének feketére állítását elvégző függvény |
| setPointColor(int u, java.awt.Color c) | void | A paraméterben megadott csúcs színét a megadott színnel beállító függvény, a grafikai adatokat tartalmazó gyűjteményben rögzítve |
| setPointCoordinates(int u, java.awt.Point p) | void | A gráf egy megadott csúcsának koordinátáit a megadott koordinátákkal (ponttal) beállító függvény |
| setPointsCount(int pointsCount) | void | A gráf csúcsinak a számát beállító függvény, a paraméterben megadott számmal |
| setPointsGraphics(java.util.Map<Integer, Pair>) | void | A gráf grafikai adatait tartalmazó változót beállító függvény, a paraméterben megadott grafikai adattal |
| setType(String type) | void | A gráf típusát beállító függvény, a paraméterben megadott függvény típussal |

#### *DirectionalCostedGraph.java*

Irányított súlyozott gráf osztály, amely az absztrakt gráf osztályból származik.



25. ábra

Változók

A változók a Graph absztrakt osztályból származnak.

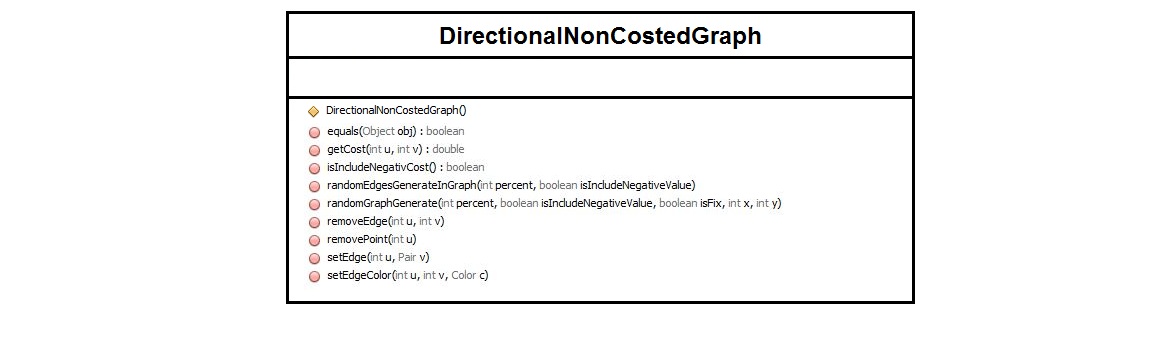
Függvények

Függvények, amelyek a *Graph* osztályban vannak, megörökli. A *Graph* osztály absztrakt függvényei kötelezően megvalósítottak.

|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Funkció** |
| *DirectionalCostedGraph()* | Konstruktor | Irányított súlyozott gráf osztály konstruktora, mely üres éllistát, üres grafikai adatokat tartalmazó gyűjteményt hoz létre és beállítja a gráf típusát irányított súlyozott gráf típusra. |
| equals(Object obj) | boolean | Két irányított súlyozott gráf egyenlőségét vizsgáló függvény, a két gráf egyenlő, ha típusuk megegyezik, ugyanazokat a csúcsokat, ugyanazokat az éleket tartalmazza és az élek súlya is megegyezik. (megjegyzés: az élek és csúcsok számának is egyeznie kell). |
| getCost(int u, int v) | double | A *Graph* osztály egy absztrakt függvényét valósítja meg, úgy hogy a paraméterben megadott él (kezdőcsúcs és végcsúcs) súlyát adja vissza, már ha létezik az él. Ha nem létezik, akkor végtelent ad vissza. |
| isIncludeNegativeCost() | boolean | *Graph* osztály egy absztrakt függvényét valósítja meg, amely, igazat vagy hamisat ad vissza, aszerint, hogy tartalmaz-e negatív súlyú élt a gráf vagy sem. |
| randomEdgesGenerateInGraph(int percent, boolean isIncludeNegativeCost) | void | Él generáló függvény, amely mindegyik lehetséges két ponthoz (oda-vissza esetben is) próbál élt beállítani, a szerint, hogy a paraméterben megadott valószínűséggel az *isProbability()* függvény igazat ad vagy sem, ha igazat ad, beállítja az élt az aktuális két csúcshoz és úgy rendel hozzá véletlenszerűen súlyt, hogy ha a 2. megadott paraméter igaz, akkor a generált súly lehet negatív is. |
| randomGraphGenerate(int percent, boolean isIncludeNegativeCost, boolean isFix, int x, int y) | void | Véletlenszerűen gráfot generáló függvény. Létrehoz fixszámú csúcsot a gráf, annyi csúcsot létrehozva, amennyi az utolsó paraméter. Ha a 3. paraméter igaz, különben a 4. és 5. paraméterben megadott intervallumból választ véletlenszerűen csúcsszámot és létrehoz ennyi csúcsot. Ezután generál éleket, az első paraméterben, százalékban megadott valószínűség szerint, ami lehet negatív súlyú, amennyiben igaz a 2. paraméter. |
| removeEdge(int u, int v) | void | A paraméterben megadott élt törli két csúcs között, már ha van köztük él és mindkét csúcs a gráfban van. A művelet után 1-gyel csökkenti a függvény a gráf él számát. |
| removePoint(int u) | void | A paraméterben megadott csúcs törlését elvégző függvény. A megadott csúcsból kiinduló éleket is törli a függvény, a végén csökkenti 1-gyel a csúcsszámot és csökkenti az él számot annyival, ahány él kivezetett a törölt csúcsból, már ha a csúcsot tartalmazta a gráf |
| setEdge(int u, Pair v) | void | *Graph* osztály egy absztrakt függvényét valósítja meg, amely él beállító függvény. Az első paraméterben megadott kezdőcsúcs listájához hozzáadja a végcsúcsot és súlyt tartalmazó rekordot az éllistában, már ha a kezdőcsúcs és végcsúcs sorszáma nem egyezik meg, (csak egyszerű gráfok lehetnek) és a megadott csúcsok szerepelnek a gráfban. Az élt éllistába jegyzése után a gráf él számát tartalmazó változót növeljük 1-gyel |
| setEdgeColor(int u, int v, java.awt.Color c) | void | A megadott él színét a megadott színnel beállító függvény. |

#### *DirectionalNonCostedGraph.java*

Irányított súlyozatlan gráf osztály, amely az absztrakt gráf osztályból származik.



26. ábra

Változók

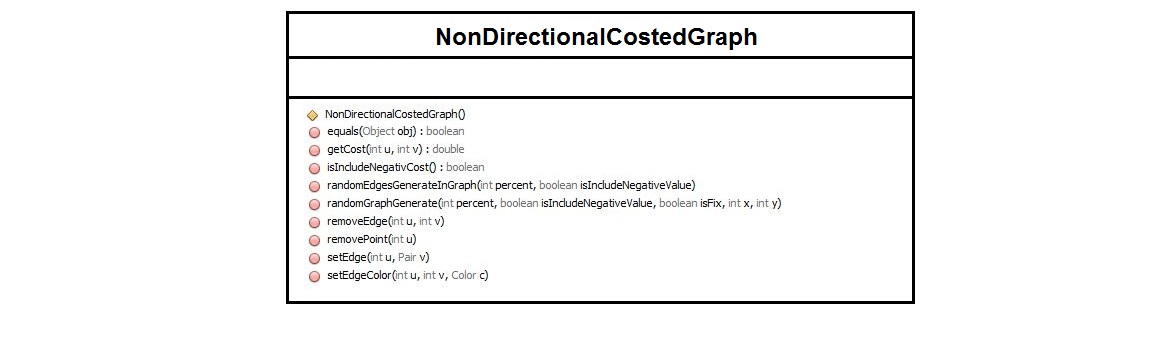
A változók a Graph absztrakt osztályból származnak.

Függvények

Függvények, amelyek a *Graph* osztályban vannak, megörökli. A *Graph* osztály absztrakt függvényei kötelezően megvalósítottak.

|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Funkció** |
| *DirectionalNonCostedGraph()* | Konstruktor | Irányítatott súlyozatlan gráf osztály konstruktora, mely üres éllistát, üres grafikai adatokat tartalmazó gyűjteményt hoz létre és beállítja a gráf típusát irányított súlyozatlan gráf típusra. |
| equals(Object obj) | boolean | Két irányított súlyozatlan gráf egyenlőségét vizsgáló függvény, a két gráf egyenlő, ha típusuk megegyezik, ugyanazokat a csúcsokat, ugyanazokat az éleket tartalmazza. (megjegyzés: az élek és csúcsok számának is egyeznie kell). |
| getCost(int u, int v) | double | Adott él súlyát visszaadó függvény. Mivel súlyozatlan gráfról van szó, ezért ez él súlyát visszaadó függvény értelmetlen, emiatt végtelen értékkel tér vissza. |
| isIncludeNegativeCost() | boolean | Igazat vagy hamisat ad vissza a függvény, aszerint, hogy a gráf tartalmaz-e negatív él súlyt vagy sem. Mivel súlyozatlan gráfról van szó, ezért mindig hamissal tér vissza, mivel súlya sem lehet, nemhogy negatív súlya. |
| randomEdgesGenerateInGraph(int percent, boolean isIncludeNegativeCost) | void | Él generáló függvény, amely mindegyik lehetséges két ponthoz (oda-vissza esetben is) próbál élt beállítani, a szerint, hogy a paraméterben megadott valószínűséggel az *isProbability()* függvény igazat ad vagy sem, ha igazat ad, beállítja az élt az aktuális két csúcshoz. |
| randomGraphGenerate(int percent, boolean isIncludeNegativeCost, boolean isFix, int x, int y) | void | Véletlenszerűen gráfot generáló függvény. Létrehoz fixszámú csúcsot a gráf, annyi csúcsot létrehozva, amennyi az utolsó paraméter. Ha a 3. paraméter igaz, különben a 4. és 5. paraméterben megadott intervallumból választ véletlenszerűen csúcsszámot és létrehoz ennyi csúcsot. Ezután generál éleket, az első paraméterben, százalékban megadott valószínűség szerint. |
| removeEdge(int u, int v) | void | A paraméterben megadott élt törli két csúcs között, már ha van köztük él és mindkét csúcs a gráfban van. A művelet után 1-gyel csökkenti a függvény a gráf él számát. |
| removePoint(int u) | void | A paraméterben megadott csúcs törlését elvégző függvény. A megadott csúcsból kiinduló éleket is törli a függvény, a végén csökkenti 1-gyel a csúcsszámot és csökkenti az él számot annyival, ahány él kivezetett a törölt csúcsból, már ha a csúcsot tartalmazta a gráf |
| setEdge(int u, Pair v) | void | *Graph* osztály egy absztrakt függvényét valósítja meg, amely él beállító függvény. Az első paraméterben megadott kezdőcsúcs listájához hozzáadja a végcsúcsot tartalmazó rekordot az éllistában, már ha a kezdőcsúcs és végcsúcs nem egyezik meg, (csak egyszerű gráfok lehetnek). Végül, ha a megadott csúcsok szerepelnek a gráfban, az élt éllistába jegyzése után, a gráf él számát tartalmazó változót növeljük 1-gyel |
| setEdgeColor(int u, int v, java.awt.Color c) | void | A megadott él színét a megadott színnel beállító függvény. |

#### *NonDirectionalCostedGraph.java*



27. ábra

Változók

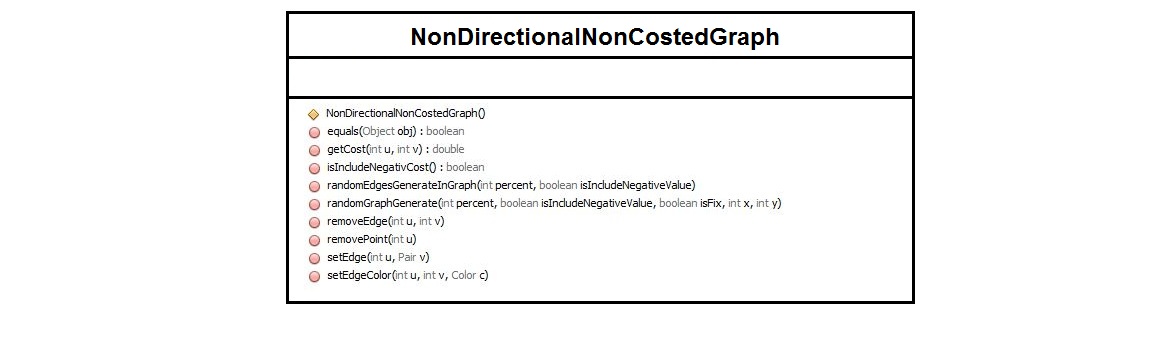
A változók a Graph absztrakt osztályból származnak.

Függvények

Függvények, amelyek a *Graph* osztályban vannak, megörökli. A *Graph* osztály absztrakt függvényei kötelezően megvalósítottak.

|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Funkció** |
| *NonDirectionalCostedGraph()* | Konstruktor | Irányítatlan súlyozott gráf osztály konstruktora, mely üres éllistát, üres grafikai adatokat tartalmazó gyűjteményt hoz létre és beállítja a gráf típusát irányítatlan súlyozatlan gráf típusra. |
| equals(Object obj) | boolean | Két irányítatlan súlyozott gráf egyenlőségét vizsgáló függvény, a két gráf egyenlő, ha típusuk megegyezik, ugyanazokat a csúcsokat, ugyanazokat az éleket tartalmazza, valamint az élek súlya is egyezik. (megjegyzés: az élek és csúcsok számának is egyeznie kell). |
| getCost(int u, int v) | double | A *Graph* osztály egy absztrakt függvényét valósítja meg, úgy hogy a paraméterben megadott él (kezdőcsúcs és végcsúcs) súlyát adja vissza, már ha létezik az él. Ha nem létezik, akkor végtelent ad vissza. |
| isIncludeNegativeCost() | boolean | *Graph* osztály egy absztrakt függvényét valósítja meg, amely, igazat vagy hamisat ad vissza, aszerint, hogy tartalmaz-e negatív súlyú élt a gráf vagy sem. |
| randomEdgesGenerateInGraph(int percent, boolean isIncludeNegativeCost) | void | Él generáló függvény, amely mindegyik lehetséges két ponthoz (oda-vissza esetet nem vizsgálja, ha egyszer megvizsgált két csúcsot, többször nem fogja) próbál élt beállítani, aszerint, hogy a paraméterben megadott valószínűséggel az *isProbability()* függvény igazat ad vagy sem. Ha igazat ad, beállítja az élt az aktuális két csúcshoz, két helyen az éllistában, mert irányítatlan gráfról van szó és úgy rendel hozzá véletlenszerűen súlyt az élhez, hogy ha a 2. megadott paraméter igaz, akkor a generált súly lehet negatív is. |
| randomGraphGenerate(int percent, boolean isIncludeNegativeCost, boolean isFix, int x, int y) | void | Véletlenszerűen gráfot generáló függvény. Létrehoz fixszámú csúcsot a gráf, annyi csúcsot létrehozva, amennyi az utolsó paraméter. Ha a 3. paraméter igaz, különben a 4. és 5. paraméterben megadott intervallumból választ véletlenszerűen csúcsszámot és létrehoz ennyi csúcsot. Ezután generál éleket, az első paraméterben, százalékban megadott valószínűség szerint, ami lehet negatív súlyú, amennyiben igaz a 2. paraméter. |
| removeEdge(int u, int v) | void | A paraméterben megadott élt törli két csúcs között, már ha van köztük él és mindkét csúcs a gráfban van. A művelet után 2-vel csökkenti a függvény a gráf él számát, mivel két helyen kell a bejegyzést törölni, amit a függvény meg is tesz az éllistában mert irányítatlan gráfról van szó. |
| removePoint(int u) | void | A paraméterben megadott csúcs törlését elvégző függvény. A megadott csúcsból kiinduló és beérkező éleket is törli a függvény, a végén csökkenti 1-gyel a csúcsszámot és csökkenti az él számot annyival, ahány él kivezetett és bevezetett a törölt csúcsból/csúcsba, már ha a csúcsot tartalmazta a gráf. |
| setEdge(int u, Pair v) | void | *Graph* osztály egy absztrakt függvényét valósítja meg, amely él beállító függvény. Az első paraméterben megadott kezdőcsúcs listájához hozzáadja a végcsúcsot és súlyt tartalmazó rekordot az éllistában és fordítva, így kifejezve, hogy irányítatlan gráfról van szó, már ha a kezdőcsúcs és végcsúcs nem egyezik meg, (csak egyszerű gráfok lehetnek) és a megadott csúcsok szerepelnek a gráfban, az élt éllistába jegyzése után a gráf élszámát tartalmazó változót növeljük 2-vel, mivel két helyen végzi el a függvény az élbeállítást. |
| setEdgeColor(int u, int v, java.awt.Color c) | void | A megadott él színét a megadott színnel beállító függvény, az éllistában két helyen állítja be, mert irányítatlan gráfról van szó. |

#### *NonDirectionalNonCostedGraph.java*



28. ábra

Változók

A változók a Graph absztrakt osztályból származnak.

Függvények

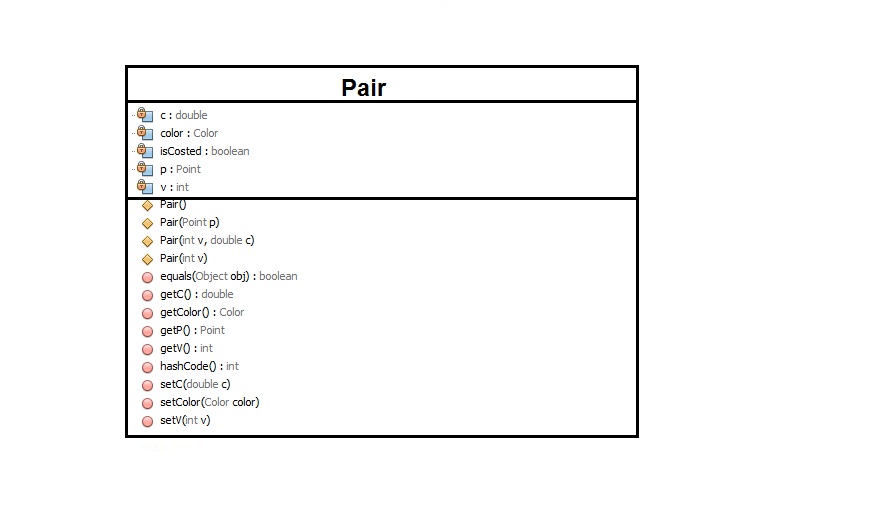
Függvények, amelyek a *Graph* osztályban vannak, megörökli. A *Graph* osztály absztrakt függvényei kötelezően megvalósítottak.

|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Funkció** |
| *NonDirectionalNonCostedGraph()* | Konstruktor | Irányítatlan súlyozatlan gráf osztály konstruktora, mely üres éllistát, üres grafikai adatokat tartalmazó gyűjteményt hoz létre és beállítja a gráf típusát irányítatlan súlyozatlan gráf típusra. |
| equals(Object obj) | boolean | Két irányítatlan súlyozatlan gráf egyenlőségét vizsgáló függvény, a két gráf egyenlő, ha típusuk megegyezik, ugyanazokat a csúcsokat, ugyanazokat az éleket tartalmazza és az élek súlya is megegyezik. (megjegyzés: az élek és csúcsok számának is egyeznie kell). |
| getCost(int u, int v) | double | Adott él súlyát visszaadó függvény. Mivel súlyozatlan gráfról van szó, ezért ez él súlyát visszaadó függvény értelmetlen, emiatt végtelen értékkel tér vissza. |
| isIncludeNegativeCost() | boolean | Igazat vagy hamisat ad vissza a függvény, aszerint, hogy a gráf tartalmaz-e negatív él súlyt vagy sem. Mivel súlyozatlan gráfról van szó, ezért mindig hamissal tér vissza, mivel súlya sem lehet, nemhogy negatív súlya. |
| randomEdgesGenerateInGraph(int percent, boolean isIncludeNegativeCost) | void | Élgeneráló függvény, amely mindegyik lehetséges két ponthoz (oda-vissza esetet nem vizsgálja, ha egyszer megvizsgált két csúcsot, többször nem fogja) próbál élt beállítani, aszerint, hogy a paraméterben megadott valószínűséggel az *isProbability()* függvény igazat ad vagy sem. Ha igazat ad, beállítja az élt az aktuális két csúcshoz, két helyen az éllistában, mert irányítatlan gráfról van szó. |
| randomGraphGenerate(int percent, boolean isIncludeNegativeCost, boolean isFix, int x, int y) | void | Véletlenszerűen gráfot generáló függvény. Létrehoz fixszámú csúcsot a gráf, annyi csúcsot létrehozva, amennyi az utolsó paraméter. Ha a 3. paraméter igaz, különben a 4. és 5. paraméterben megadott intervallumból választ véletlenszerűen csúcsszámot és létrehoz ennyi csúcsot. Ezután generál éleket, az első paraméterben, százalékban megadott valószínűség szerint. |
| removeEdge(int u, int v) | void | A paraméterben megadott élt törli két csúcs között, már ha van köztük él és mindkét csúcs a gráfban van. A művelet után 2-vel csökkenti a függvény a gráf él számát, mivel két helyen kell a bejegyzést törölni, amit a függvény meg is tesz az éllistában mert irányítatlan gráfról van szó. |
| removePoint(int u) | void | A paraméterben megadott csúcs törlését elvégző függvény. A megadott csúcsból kiinduló és beérkező éleket is törli a függvény, a végén csökkenti 1-gyel a csúcsszámot és csökkenti az él számot annyival, ahány él kivezetett és bevezetett a törölt csúcsból/csúcsba, már ha a csúcsot tartalmazta a gráf. |
| setEdge(int u, Pair v) | void | *Graph* osztály egy absztrakt függvényét valósítja meg, amely él beállító függvény. Az első paraméterben megadott kezdőcsúcs listájához hozzáadja a végcsúcsot tartalmazó rekordot az éllistában és fordítva, így kifejezve, hogy irányítatlan gráfról van szó, már ha a kezdőcsúcs és végcsúcs nem egyezik meg, (csak egyszerű gráfok lehetnek). Ha a megadott csúcsok szerepelnek a gráfban, az élt éllistába jegyzése után a gráf él számát tartalmazó változót növeljük 2-vel, mivel két helyen végzi el a függvény az él beállítást |
| setEdgeColor(int u, int v, java.awt.Color c) | void | A megadott él színét a megadott színnel beállító függvény, az éllistában két helyen állítja be, mert irányítatlan gráfról van szó. |

### Segéd struktúrák

#### Pair.java

Pair osztály/rekord, olyan egyedi osztály, amely a gráf működés szempontjából többféleképpen is viselkedhet. Ilyen típusú adatokból áll a gráf éllistája és a gráf grafikus adatait tartalmazó gyűjtemény, de más-más funkcióval. Az éllistát mentésnél használjuk, ezért ez az osztály megvalósítja a *java.io.Serializable* interfészt.



35. ábra

Változók

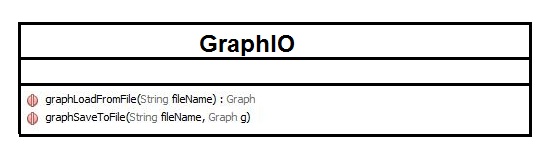
|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Funkció** |
| c | double | Valós típusú változó, gráf élének súlya. |
| color | java.awt.Color | Szín típusú változó, él/csúcs színe, kezdetben fekete. |
| isCosted | boolean | Logikai típusú változó, súlyozott vagy súlyozatlan élről van-e szó. |
| p | java.awt.Point | Koordinátákat tartalmazó változó. |
| v | int | Egész típusú változó, gráf végcsúcsa. |

Függvények

|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Funkció** |
| Pair() | Konstruktor | Üres konstruktor. |
| Pair(Point p) | Konstruktor | Olyan konstruktor, amelynek használata esetén koordinátákkal jön létre az osztály. Ekkor egy csúcs színének és koordinátáinak a tárolására használatos (gráfok grafikai adatait tartalmazó értékek használják). |
| Pair(int v) | Konstruktor | Olyan konstruktor, amelynek használata esetén egy csúccsal és egy súllyal jön létre az osztály (súlyozott gráfok ezt használják az éllistában). Ekkor egy csúcsból kiinduló él súlyát és végcsúcsát tárolja az osztály. A súlyozott-e logikai változó igazra állítódik. |
| Pair(int v, double c) | Konstruktor | Olyan konstruktor, amelynek használata esetén egy csúccsal jön létre az osztály (súlyozatlan gráfok ezt használják az éllistában). Ekkor egy csúcsból kiinduló él végcsúcsát tárolja az osztály. A súlyozott-e logikai változó hamisra állítódik. |
| equals(Object obj) | boolean | Él funkcionalításnál használatos egyenlőség vizsgáló függvény. Egyenlőkét rekord, ha súlyozatlan esetben a végcsúcsok megegyeznek. Súlyozott esetben ugyanez, de még a súlyoknak is egyeznie kell ebben az esetben. |
| getC() | double | Ha súlyozott él funkcionalítású az osztály, akkor a függvény a rekord súlyát adja vissza. Ha nem súlyozott, akkor végtelent. |
| getColor() | java.awt.Color | A rekord színét visszaadó függvény. |
| getP() | java.awt.Point | Ha koordináta funkcionalítású az osztály, akkor a csúcs koordinátáit adja vissza a függvény. |
| getV() | int | Ha él funkcionalítású a rekord, akkor az él végcsúcsát adja vissza a függvény. |
| setC(double cost) | void | Ha súlyozott él funkcionalítású a rekord, akko a paraméterben megadott értékkel beállítódik a rekord *c* vagy súlyváltozója. |
| setColor(java.awt.Color) | void | A rekord színét megváltoztató függvény a paraméterben megadott színnel. |
| setV(int v) | void | Ha él funkcionalítású az osztály, akkor az él végcsúcsát beállítódik a paraméterben megadott értékkel. |

#### GraphIO.java

Gráfok fájlkezelési műveleteit kezelő osztály.



37. ábra

Függvényei

|  |  |  |
| --- | --- | --- |
| **Név** | **Típus** | **Funkció** |
| graphLoadFromFile(String fileName) | Graph | A paraméterben megadott fájlnevű fájlból, gráf betöltő statikus függvény. A betöltött gráfot adja vissza. Először beolvassa a gráf típusát és létrehoz egy ilyen típusú üres gráfot. Majd beállítja a létrehozott gráf éllistáját, a fájlban találtható éllistával, ezután a csúcsszámot, az élszámot és a gráf grafikai adatait tartalmazó gyűjteményt. Végül minden él, csúcs színét beállítja feketére és visszaadja a gráfot. Hiba esetén, például nem létező fájl vagy a gráf hibás adatokat olvas be, akkor a függvény ezt jelzi és null értéket ad vissza. |
| graphSaveToFile(String fileName, Graph g) | void | A paraméterben megadott gráfot a paraméterben adott nevű fájlba mentő statikus függvény. Az adott fájlt megnyitja/létrehozza, először a gráf típusát írja bele a fájlba, majd a gráf éllistáját, csúcsok és élek számát, valamint gráf grfikai adatait tartalmazó gyűjteményt. Ha hibás a megnyitott fájl, akkor nem csinál semmit a függvény. |