

**KAUNO TECHNOLOGIJOS UNIVERSITETAS**

**INFORMATIKOS FAKULTETAS**

**Kursinis darbas Cluster 2.0**

Grupės nariai:

Arnas Švenčionis

Gabrielė Sakalauskaitė

Kristis Sirijotavičius

Marija Rutkauskaitė

Benas Pociūnas

Kaunas, 2019

# **Turinys**

[**Turinys**](#_t5wi0em2dvgm) **2**

[**Įvadas**](#_tcxyqmp9imem) **4**

[Tikslas:](#_lsuuep5vsr2q) 4

[Darbo eiga:](#_2xqcw2flfyqt) 4

[**Tikslo funkcija**](#_ak0gtedoy4rz) **5**

[**Vartotojo vadovas**](#_ppdz6vxilvs0) **8**

[Apribojimai:](#_3qdvvqteakll) 9

[**Išvados**](#_yi6ftv9swg7v) **10**

# 

# 

# 

# 

# 

# 

# **Įvadas**

Kurinis darbas, skirtas susipažinti ir ištirti klusterio principą, kurio realizacija yra naudojama Europos šalyse. Projekte atliekama duomenų analizė bei klusterio principo realizacija kartu su Kauno Technologijos Universiteto ekonomikos fakulteto studentais. Su ekonomikos fakulteto studentų sudarytomis formulėmis stengsimės įvertinti ir parinkti optimaliausius sprendimus, ieškant sandėlių lokacijų, kurdami programą.

## Tikslas:

Surasti bei sudaryti funkcijas, parašyti programą ieškant trumpiausių kelių krovinių transportavimui tarp sandėlių, siekiant sumažinti CO2 emisiją bei taupant kaštus.

## Darbo eiga:

1. Išanalizavome pateiktus duomenis CLUSTER 2.0 modulio aplinkoje.
2. Išnagrinėjome iš EVF studentų gautas formules, skaičiavimo metodiką sandėlių išdėstymo gerumo įvertinimui nustatyti. Įvertinome ar ją galima realizuoti - suderinome skaičiavimo metodikos patikslinimus, papildomų duomenų užtikrinimą.
3. Suderinome funkcinius reikalavimus su EVF studentais, kad tiek realizuota projekto apimtis nebūtų per didelė, tiek užtikrintų funkcionalumą reikalingą bendro darbo įgyvendinimui.
4. Parašėme programą duomenų failų nuskaitymui ir pasirinkome tinkamą struktūrą svorinio grafo duomenims saugoti.
5. Realizavome funkcionalumą, kuris leistų įvertinti parinkto išdėstymo gerumą esant pateiktam sandėlių išdėstymui.

# Tikslo funkcija

Ekonomikos fakulteto studentų sudarytos optimalios funkcijos:

Tikslo funkcija :

SSK = PSK \* KSK \* KT

SSK - Sandėlio statybos kaštai

PSK - Pastovieji statybos kaštai;;

KSK - Kintamieji statybos kaštai;

KT - kiekis (tonomis);

m – sandeliu kiekis;

SVK - Sandėlio valdymo kaštai;

PSVK - Pastovieji sandėlio valdymo kaštai;

KSVK - Kintantys sandėlio valdymo kaštai;

T – tonos;

m – sandelių kiekis;

PPR – Paklausa, pasiula regionu;

TK – Transportavimo kaina(priklausomai nuo transporto);

D – atstumas;

CO – CO2 ismetimas;

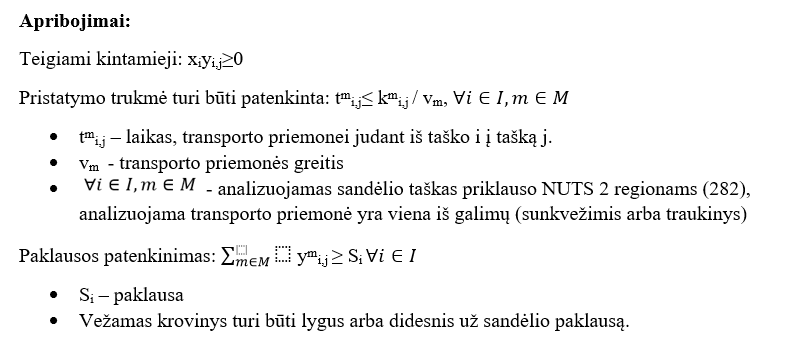
m – taškų kiekis;

PKK – transportavimo kainos, priklausomai nuo sandelių ir optimalesnių kelių;

SSK – sandelio statybos kaina;

SVK – sandelio išlaikymo kaina;

m – taškų kiekis;



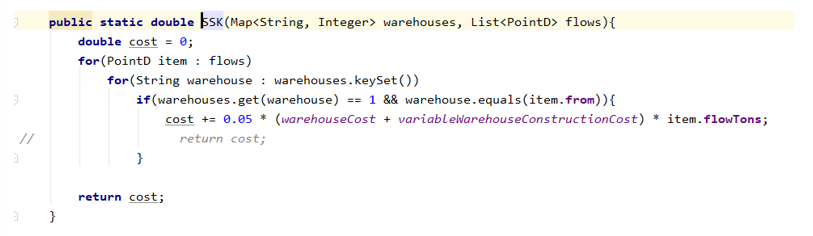
# Kaip programa veikia

Tam, kad programa veiktų, privalomi veiksmai yra duomenų nuskaitymas ir tikslo funkcijos lygties sudarymas. Maršruto duomenis laikėme PointD klasėje. Ji susidaro iš maršruto pradžios, pabaigos, atstumo ir paklausos bei pasiūlos sumos. Duomenis laikėme HashMap<K, V> struktūroje. Raktas - taško pavadinimas, o reikšmė - visų maršrutų objektų sąrašas.

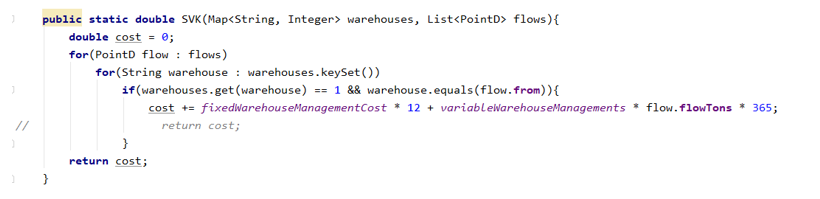
Nuskaičius duomenis, programa sudaro geriausių (turinčių aukščiausią eksporto ir importo sumą) taškų sąrašą. Šie taškai yra naudojami skaičiavimams. Kiekvienoje iteracijoje yra pridedamas kitas, geriausias sąraše esantis taškas. Su gautu nauju sandėlių išdėstymu, daromi veiksmai yra:

1. Skaičiuojamos sandėlių išlaidos, galimi optimalesni maršrutai iš kiekvieno taško į kitus. Tai yra daroma, analizuojant maršrutus. Tikrinama, ar abu taškai turi po sandėlį, jei taip, vežama traukiniu, neieškant optimalesnio maršruto. Jei tik vienas taškas turi sandėlį, yra ieškomas artimiausias sandėlis nuo taško, neturinčio sandėlio ir skaičiuojama tokio transportavimo kaina. Jeigu ji yra pigesnė, nei transportavimas sunkvežimiu, ji pasirenkama. Jei abu taškai neturi sandėlių, abiem taškam yra ieškomi artimiausi sandėliai, tikrinama ar vežimo kaina per sandėlius yra pigesnė, nei tiesiog vežimas sunkvežimiu. Jei taip, pasirenkamas toks alternatyvus kelias.
2. Apskaičiuojamos optimalių maršrutų transporto pristatymo ir emisijos kainos, dauginant reikiamą vežimo kiekį iš atstumo, naudojamų transporto priemonių emisijos lygio ir pervežimo kainos. Tai atliekama kiekvienam maršrutui.
3. Iteracijos pabaigoje yra tikrinama ar kaina yra mažesnė, nei buvusioje iteracijoje.
4. Jeigu gauta kaina yra mažesnė, ji yra išsaugoma, sandėlių išdėstymas yra įrašomas į atsakymų failą.

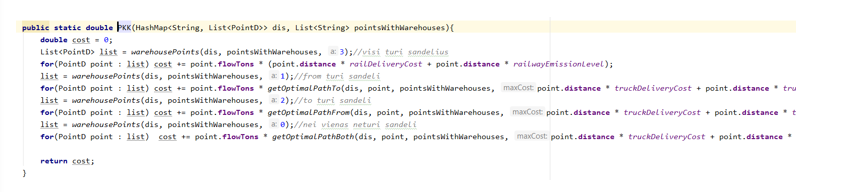
Pagrindinės programos funkcijos:



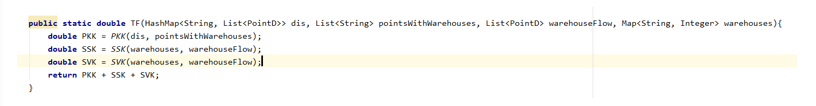
1. Apskaičiuojama visų sandėlių pastatymo kainų suma.



1. Apskaičiuojama visų sandėlių išlaikymo kainų suma.



1. Apskaičiuojami optimalūs maršrutai, jų kainos padaugintos iš paklausos ir pasiūlos.



1. Apskaičiuojama tikslo funkcija - galutinė kaina.

## Apribojimai:

1. Atstumai, paklausos ir pasiūlos, apskaičiuotos kainos negali būti neigiami skaičiai.
2. Du sandėliai negali turėti atstumo, lygaus nuliui.
3. Visos pasiūlos ir paklausos turi būti patenkintos.

# Išvados

Remiantis diskrečiųjų struktūrų paskaitų metu išdėstytomis teorinėmis žiniomis ir įgytais praktiniais įgūdžiais, parašyta programa, kuri suranda optimaliausius maršrutus, siekdama sumažinti CO2 emisiją bei kaštus, atsižvelgiant į tam tikras sąlygas bei paisant tam tikrus programos apribojimus. Remiantis parašyta programa, buvo surastos tikslingiausios ir optimaliausios sandėlių vietos bei nustatyta efektyvi tikslo funkcijos reikšmė. Iš jų galima daryti išvadas, kur statyti sandėlius būtų geriausia, siekiant mažinti CO2 emisiją bei taupant logistikos infrastruktūros išlaidas.