

VILNIAUS GEDIMINO TECHNIKOS UNIVERSITETAS

Fundamentinių mokslų fakultetas

InFOrmacinių technologijų katedra

**Prekybos bazės uždavinys**

Laboratorinis darbas nr. 1

Atliko: DGTfm-18 gr. st. Gabrielė Gvaizdikaitė

Priėmė: lekt. dr. A.Igumenov

Vilnius, 2019

**Uždavinio sąlyga:**

Prekybos bazė įsipareigojo saugoti prekes ir išduoti jas vartotojui kiekvieną dieną po 𝑏 tonų. Ji gali gauti prekes periodiškai kas 𝑡 parų vienodo dydžio partijomis po 𝑞 tonų. Vienos prekių partijos priėmimas ir pakrovimas į sandėlį kainuoja 𝑐2 Lt. Vienos tonos vidutinės išdavimo išlaidos 𝑐1 = 𝑐∗ ∙ 𝑡 Lt. Naujos prekės vežamos paskutinę anksčiau atvežtų prekių išdavimo dieną. Reikia nustatyti optimalią vienos prekių partijos apimtį 𝑞 ir jos vežimo periodą 𝑡, kad bazės išlaidos per parą būtų mažiausios.

*b = + 2,5 = + 2,5 = 3,5*

*c2 = ( mod 30) +150 = (20145120 mod 30) + 150 = 150*

*c\* = (*

1. **Uždavinio matematinis modelis**

b - tonų skaičius, išduodamas vartotojui per vieną dieną;

t – parų skaičius;

c2 – vienos prekių partijos pakrovimo ir priemimo į sandėlį kaina;

c1 = 𝑐∗ ∙ 𝑡 – vienos tonos vidutinės išdavimo išlaidos;

Vienkartinės (vienos dienos) prekių išdavimo išlaidos – c1 ∙ b Lt;

Vienos dienos prekių priemimo ir pakrovimo į sandėlį išlaidos - ;  
Prekybos bazės išlaidos per parą aprašomos kintamojo 𝑡 funkcija:

Be to, žinome, kad t = , todėl q = t ∙ b;

Taigi, uždavinio matematinis modelis:

*∙ 𝑐∗ ∙ 𝑡 ∙ b;*

1. **Tikslo funkcija**

*∙ 𝑐∗ ∙ 𝑡 ∙ b);*

**Matlab programos kodai:**

1. **Tikslo funkcijos kodas**

function [tf] = tikslo\_funkcija(t)

%reikšmės nustatomos iš duotos sąlygos

b = mod(20145120, 7)/2 + 2.5;

c2 = mod(20145120, 30) + 150;

c = mod(20145120, 5) + 3;

%matematinis modelis

tf = c2./t+b\*c\*t;

end

1. **Intervalo dalijimo pusiau metodas**

function [x\_min\_vid, f\_min, nauja\_pradzia, nauja\_pabaiga, iteracija, naujas\_ilgis, x, x\_reiksme] = metod(pradzia, pabaiga)

iteracija = 1; %pirmoji iteracija skaiciavimui

x\_vid = (pradzia + pabaiga)/2; %skaiciuojamo intervalo vidurio taškas

fx\_vid = tikslo\_funkcija(x\_vid); %esamo matematinio modelio reikšme intervalo vidurio taške

x\_min\_vid = x\_vid; %esamas minimalus intervalo vidurio taškas

fx\_vid\_1 = fx\_vid; %nauja funkcijos reikšme

fx\_vid\_2 = fx\_vid; %nauja funkcijos reikšme

x(iteracija) = x\_vid; %išsisaugome vidurio taške intervale

x\_reiksme(iteracija) = fx\_vid\_1; %išsaugoma f-jos reikšme x\_vid taške

nauja\_pradzia = pradzia; %nauja intervalo pradžia

nauja\_pabaiga = pabaiga; %nauja intervalo pabaiga

ilgis = pabaiga - pradzia; %turimas intervalo ilgis

naujas\_ilgis = ilgis; %išsaugomas intervalo ilgis

while fx\_vid\_2 >= fx\_vid\_1%ciklas vyks tol, kol fx\_vid\_2 >= fx\_vid\_1

iteracija = iteracija + 1;

x1 = pradzia + ilgis/4; %naujai pradžiai nustatyti pridedame ketvirti intervalo ilgio

x2 = pabaiga - ilgis/4; %nauja intervalo pabaiga

fx1 = tikslo\_funkcija(x1); %funkcijos reikšme naujoje intervalo pradžioje

fx2 = tikslo\_funkcija(x2); %funkcijos reikšme naujoje intervalo pabaigoje

%jeigu funkcijos reikšme naujoje intervalo pradžioje yra mažesne už matematinio modelio reikšme buvusiame vidurio taške, tuomet

if(fx1 < fx\_vid)

pabaiga = x\_vid; %intervalo pabaiga yra buves vidurio taškas

x\_vid = x1; %buves vidurio taškas tampa nauja intervalo pradžia

else

%kitu atveju, jei funkcijos reikšme naujoje intervalo pabaigoje yra mažesne už matematinio modelio reikšme buvusiame vidurio taške, tuomet

if(fx2 < fx\_vid)

pradzia = x\_vid; %intervalo pradžia yra buves vidurio taškas

x\_vid = x2; %buves vidurio taškas tampa nauja intervalo pabaiga

else %kitu atveju

pradzia = x1; %intervalo pradžia pasistumia

pabaiga = x2; %intervalo pabaiga atsitraukia

end

end

fx\_vid\_1 = tikslo\_funkcija(x\_vid); %matematinio modelio reikšme naujame vidurio taške

x(iteracija) = x\_vid; %išsisaugome taške

x\_reiksme(iteracija) = fx\_vid\_1; %išsaugome reikšme

ilgis = pabaiga - pradzia; %išsaugomas naujas intervalo ilgis

if(fx\_vid\_2 > fx\_vid\_1) %jeigu tenkina šia salyga tuomet yra perrašomos reikšmes

naujas\_ilgis = ilgis;

nauja\_pradzia = pradzia;

nauja\_pabaiga = pabaiga;

x\_min\_vid = x\_vid;

fx\_vid\_2 = fx\_vid\_1;

end

end

f\_min = fx\_vid\_2; %rasta matematinio modelio minimali reikšme

end

1. **Išlaidų apskaičiavimas ir grafiko braižymas**

function [q, t, x\_min\_vid, f\_min, islaidos, iteracija, ilgis] = min\_islaidos(pradzia, pabaiga)

[x\_min\_vid, f\_min, nauja\_pradzia, nauja\_pabaiga, iteracija, ilgis, x, x\_reiksme] = pirmas(pradzia, pabaiga);

vid\_taskas = (nauja\_pradzia+nauja\_pabaiga)/2;%randame intervalo vid. taska

vid\_taskas\_apacia = floor(vid\_taskas); %randame intervalo vid. tasko apacia

f\_apacia = tikslo\_funkcija(vid\_taskas\_apacia); %randame modelio reiksme vd\_taskas\_apacia taske

vid\_taskas\_virsus = ceil(vid\_taskas); %randame intervalo vid. tasko virsus

f\_virsus = tikslo\_funkcija(vid\_taskas\_virsus); %randame modelio reiksme vd\_taskas\_virsus taske

if(f\_apacia > f\_virsus) %jeigu virsutine modelio reiksme yra didesne nei apataine, tada

islaidos = f\_virsus; %isaugome virsutine reiksme

t = vid\_taskas\_virsus; %issaugome paros reiksme

else %kitu atveju

islaidos = f\_apacia; %isaugome apatine reiksme

t = vid\_taskas\_apacia; %issaugome paros reiksme

end

a = pradzia:1:pabaiga; %optimalus prekiu vezimo periodas

b = tikslo\_funkcija(a); %islaidos kiekviename periode

%auksinio pjuvio metodas

optimizavimas = optimset('Display', 'iter','MaxFunEvals', 1000,'PlotFcns', @optimplotfval , 'TolX', ilgis);

[xfminbnd, yfminbnd] = fminbnd(@tikslo\_funkcija, pradzia, pabaiga, optimizavimas);

figure %breziamas grafikas

plot(a, b, '.-', x, x\_reiksme, '.-'), legend('legendaa', 'legendab');

title('Prekybos bazes islaidu kitimas pagal vezimo perioda t');

xlabel('Prekiu vezimo periodas (t)');

ylabel('Prekybos bazes islaidos per viena diena');

b = mod(20145120, 7)/2 + 2.5;

f\_min

x\_min\_vid

t

q = t \* b;

q

islaidos

iteracija

xfminbnd

yfminbnd

end

**4 Rezultatai**

* Tikslo funkcijos minimumas **80.9949Lt**;
* Tikslo funkcijos minimumo taškas **4.6250**;
* Optimalus prekių vėžimo periodas yra 4 paros, o partijos apimtis – **14t.**
* Tuomet išlaidos yra **79.5Lt**;
* Naudojant fminbnd, reikia atlikti 4 iteracijas ir gauname:
  + Tikslo funkcijos minimumas **83.6009Lt**;
  + Tikslo funkcijos minimumo taškas **5.2310**;



