



KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS

4 Laboratorinis darbas
Nr. 17

Atliko:
IFE-8 gr. studentas
Kemežys Martynas

Priėmė:
lekt. Andrius Kriščiūnas

KAUNAS, 2020

TURINYS

1. UŽDUOTIS.....	3
2. Sprendinys.....	3
2.1 Diferencialinė lygtis.....	3
2.2 Gauti sprendinio grafikai.....	4
2.2.1 Eulerio ir Rungės ir Kutos grafikai(žingsnis 0.5)	4
2.2.2 Eulerio ir Rungės ir Kutos grafikai(žingsnis 1)	5
2.2.3 Eulerio ir Rungės ir Kutos grafikai(žingsnis 2)	6
2.2.4 Eulerio ir Rungės ir Kutos grafikai(žingsnis 5)	7
2.3 Atsakymai į pateiktus klausimus uždavinyje	9
2.3.1 Kada kūno temperatūra pasiekia aplinkos temperatūrą?	9
2.3.2 Išspręskite tą pačią lygtį, jeigu pradinė kūno temperatūra lygi T_2	9
2.3.3 Kaip skiriasi su skirtingomis pradinėmis kūno temperatūromis gauti sprendiniai ir jų savybės?	10
3. Kodo fragmentas.....	11
3.1 Eulerio kodo fragmentas	11
3.2 IV eilės Rungės ir Kutos kodo fragmentas	12
4. IŠVADOS	13

1. UŽDUOTIS

5 – užduotis (16-20 variantams)

T_1 temperatūros kūnas patalpinamas į aplinką, kurios temperatūra T_{A1} . Tariaama, kad aplinkos temperatūra yra palaikoma išorinių šaltinių ir kūno temperatūra neturi įtakos aplinkos temperatūrai. Praėjus laikui t_s aplinkos temperatūra pradeda kisti pagal nurodytą dėsnį $T(t)$ ir pakinta iki T_{A2} , kuri yra palaikoma likusį laiką. Žinoma, kad Niutono temperatūros kaitos dėsnyje taikomas proporcingumo koeficientas priklauso nuo kūno temperatūros pagal dėsnį $k(T)$. Raskite, kaip kinta kūno temperatūra nuo pradinio laiko momento iki t_{max} . Kada kūno temperatūra pasiekia aplinkos temperatūrą?

Išspręskite tą pačią lygtį, jeigu pradinė kūno temperatūra lygi T_2 . Kaip skiriasi su skirtingomis pradinėmis kūno temperatūromis gauti sprendiniai ir jų savybės (stabilumo ir tikslumo žingsniai)?

Uždavinyje naudojami dydžiai:

Varianto numeris	T_1 , K	T_{A1} , K	t_s , s	T_{A2} , K	t_{max} , s	T_2 , K
17	400	320	30	460	80	270

Lentelė. 1

Uždavinyje naudojami dėsniai:

Varianto numeris	$T_A(t)$, K	$k(T)$
17	$T_A(t) = T_{A1} + \frac{(T_{A2} - T_{A1})}{2} \left(1 - \cos \left(\frac{\pi}{20} (t - t_s) \right) \right)$	$k(T) = -0,01 - 0,16 \left(\frac{T - 273}{100} \right) - 0,04 \left(\frac{T - 273}{100} \right)^2$

Lentelė. 2

2. Sprendinys

2.1 Diferencialinė lygtis

Niutono temperatūros kitimo dėsnis:

$$dT/dt = k \cdot (T - T_A)$$

T = kūno temperatūra laiko momentu

T_A = aplinkos temperatūra

k = temperatūros kitimos greitis

Eulerio dėsniu lygtis:

$$T \, dt = T + z \cdot dt$$

T = kūno temperatūra

z = žingsnis

dt = išvestinė

IV eilės Rungės ir Kutos etapų lygtys:

1 etapas:(žingsnis)

$$dt_1 = k \cdot (T - T_A)$$

$$T_{11} = T + z/2 \cdot dt_1$$

2 etapas(atgalinis žingsnis)

$$dt_2 = k \cdot (T_{11} - T_A)$$

$$T_2 = T + z/2 \cdot dt_2$$

3 etapas(vidurinio taško žingsnis)

$$dt_3 = k \cdot (T_2 - T_A)$$

$$T_3 = T + z \cdot dt_3$$

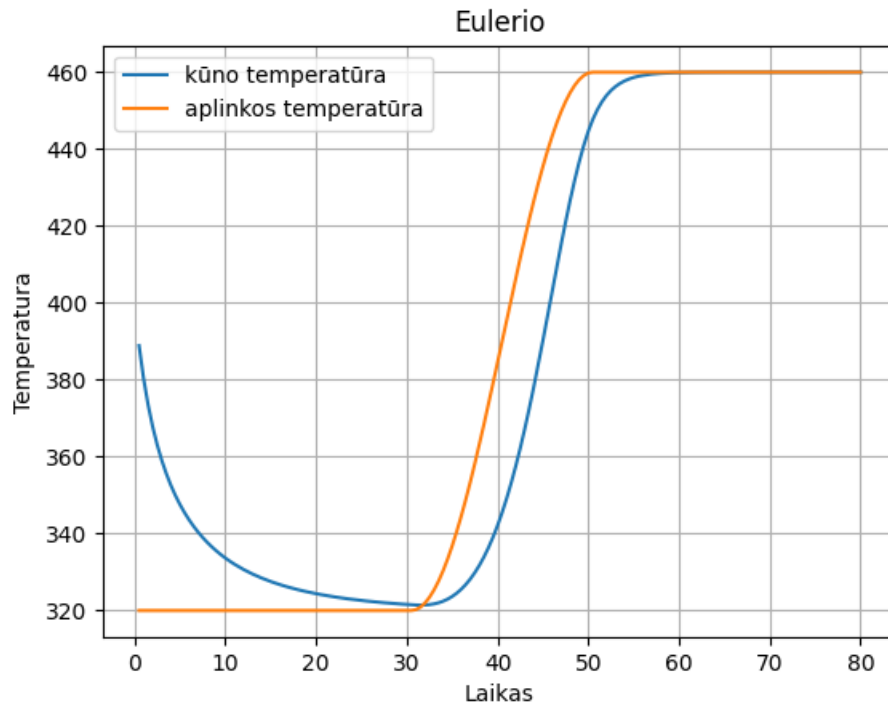
4 etapas(simpsono žingsnis)

$$dt4 = k * (T3 - TA)$$

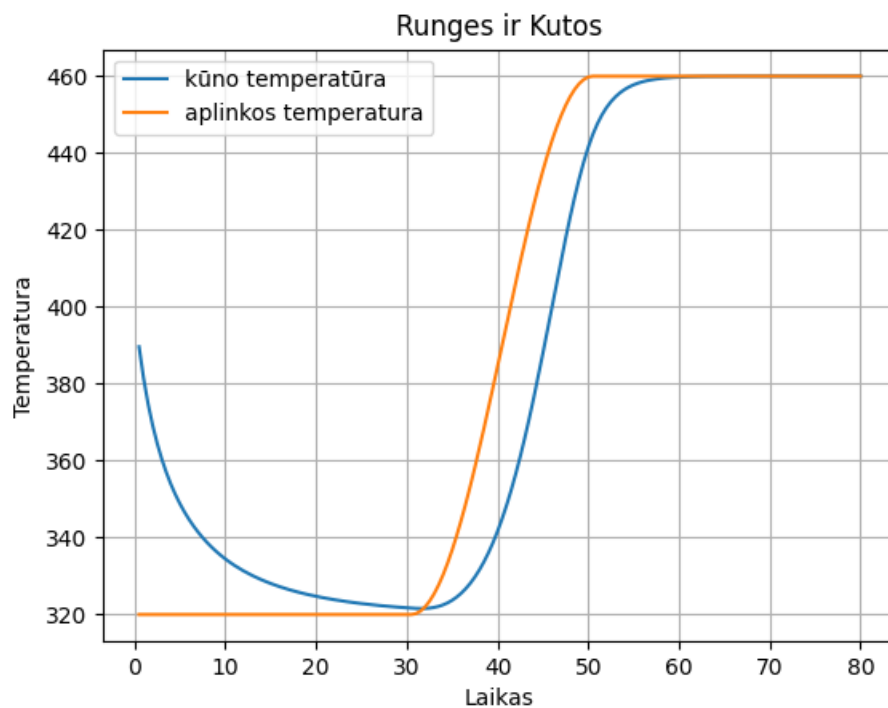
$$T = T + z/6 * (dt1 + 2 * dt2 + 2 * dt3 + dt4)$$

2.2 Gauti sprendinio grafikai

2.2.1 Eulerio ir Rungės ir Kutos grafikai(žingsnis 0.5)

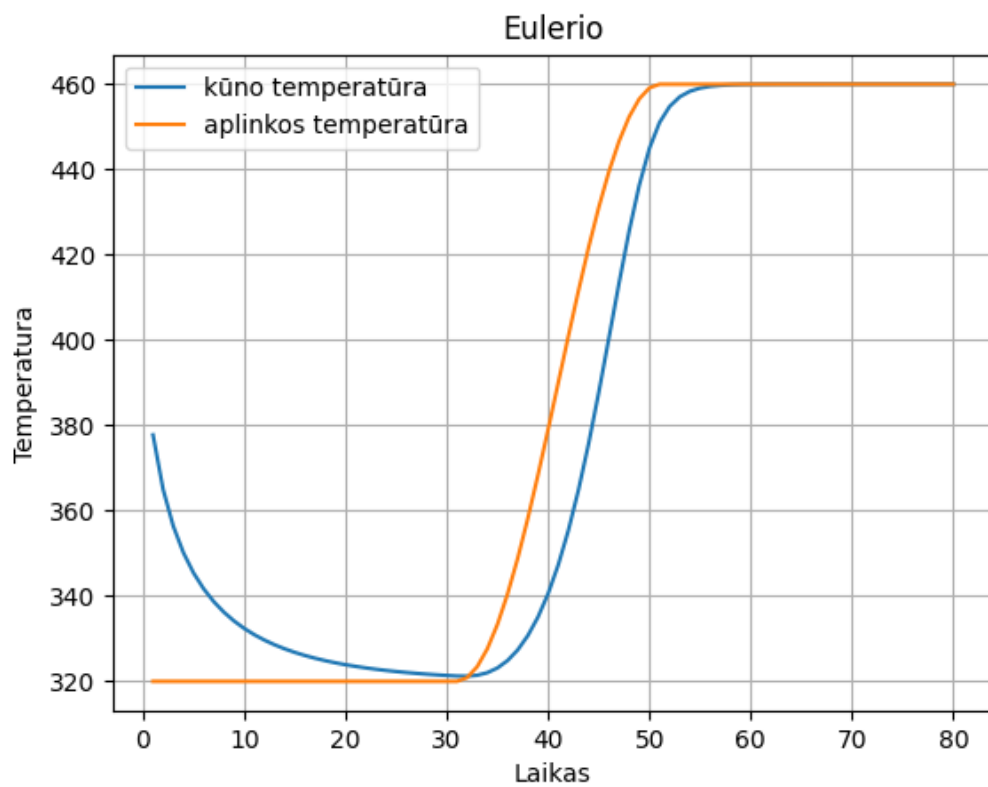


Grafikas. 1

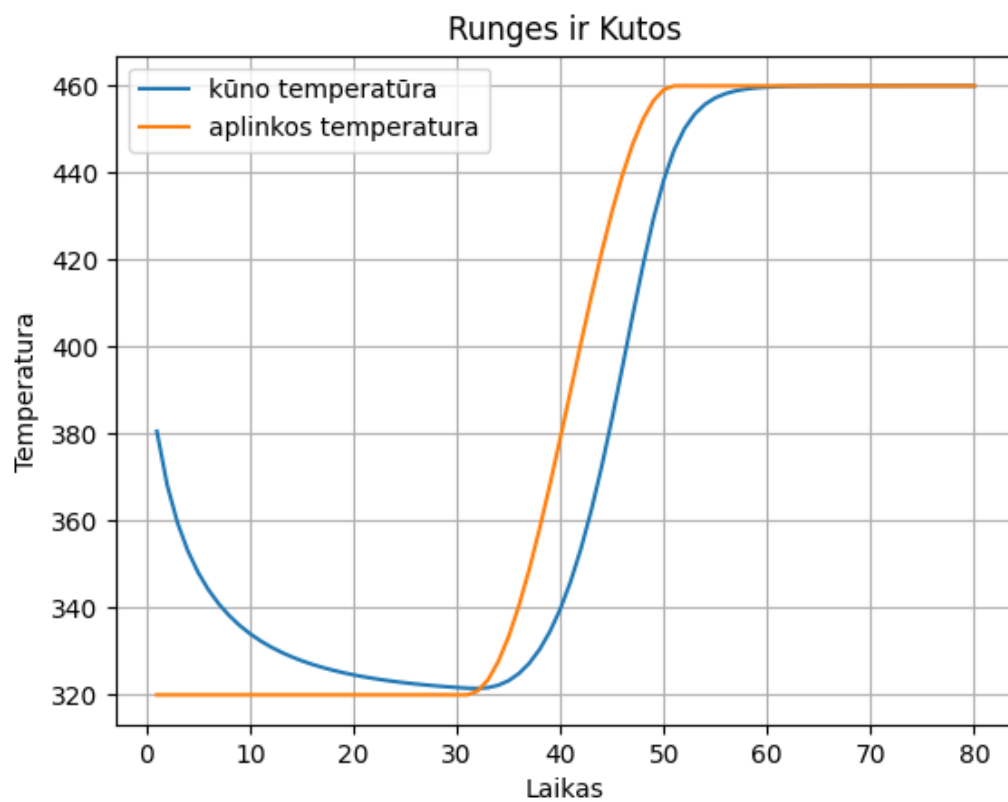


Grafikas. 2

2.2.2 Eulerio ir Rungès ir Kutos grafikai(žingsnis 1)

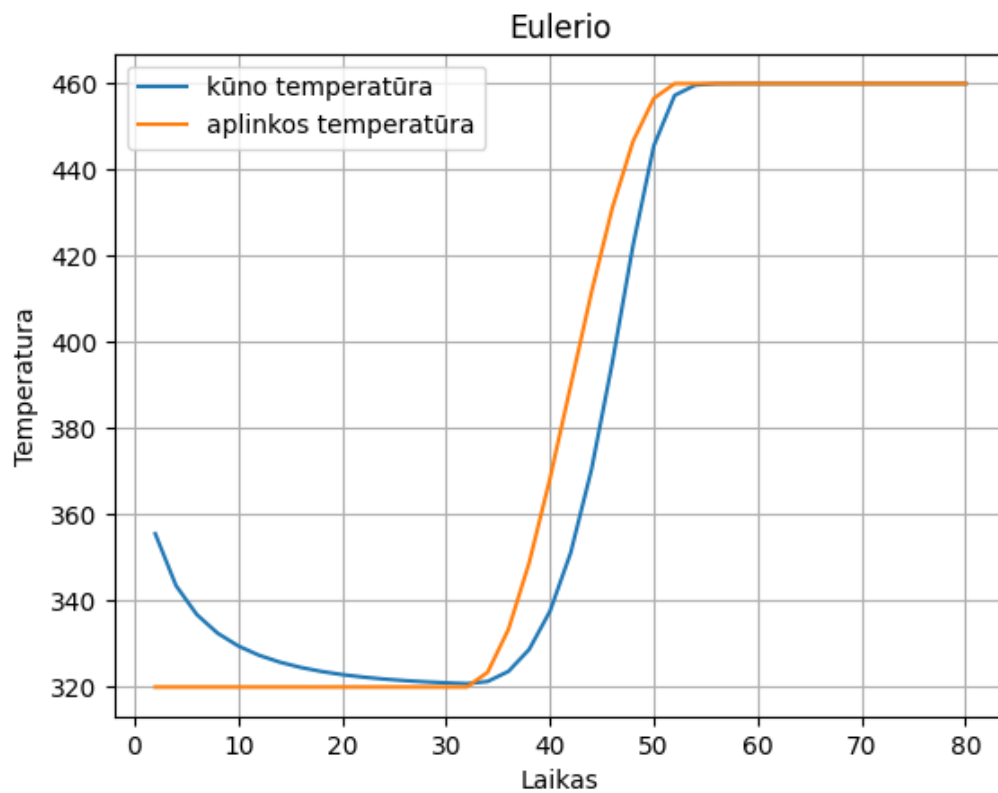


Grafikas. 3

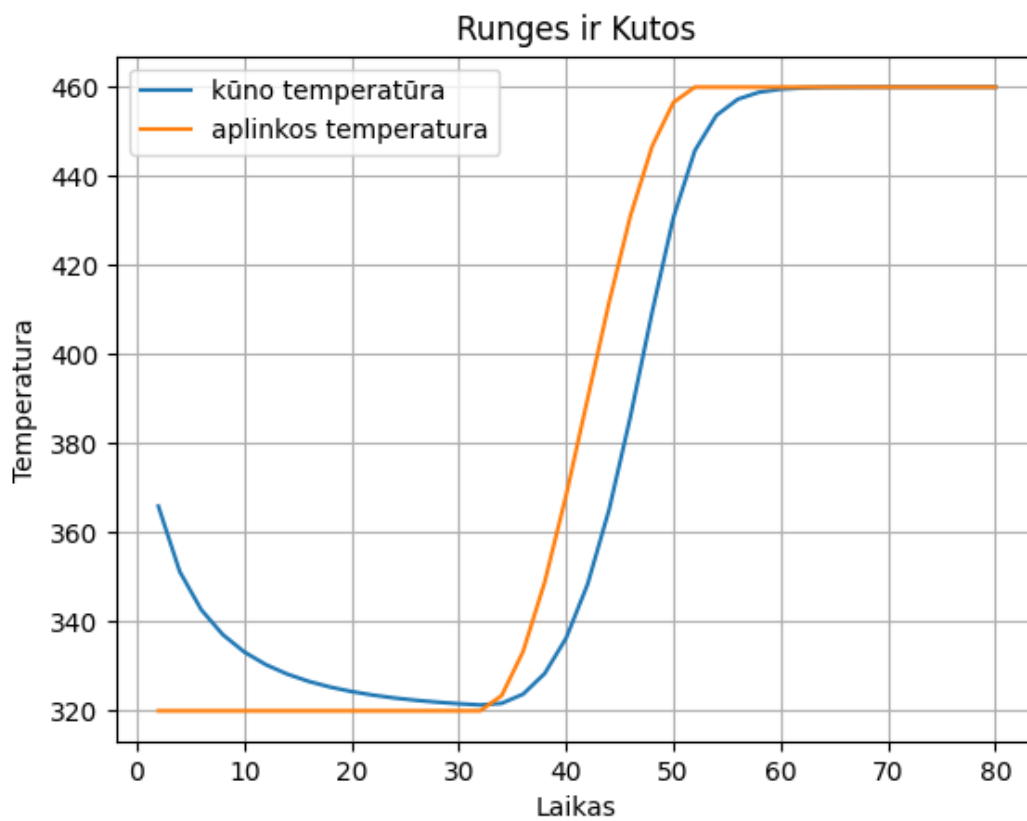


Grafikas. 4

2.2.3 Eulerio ir Rungės ir Kutos grafikai(žingsnis 2)

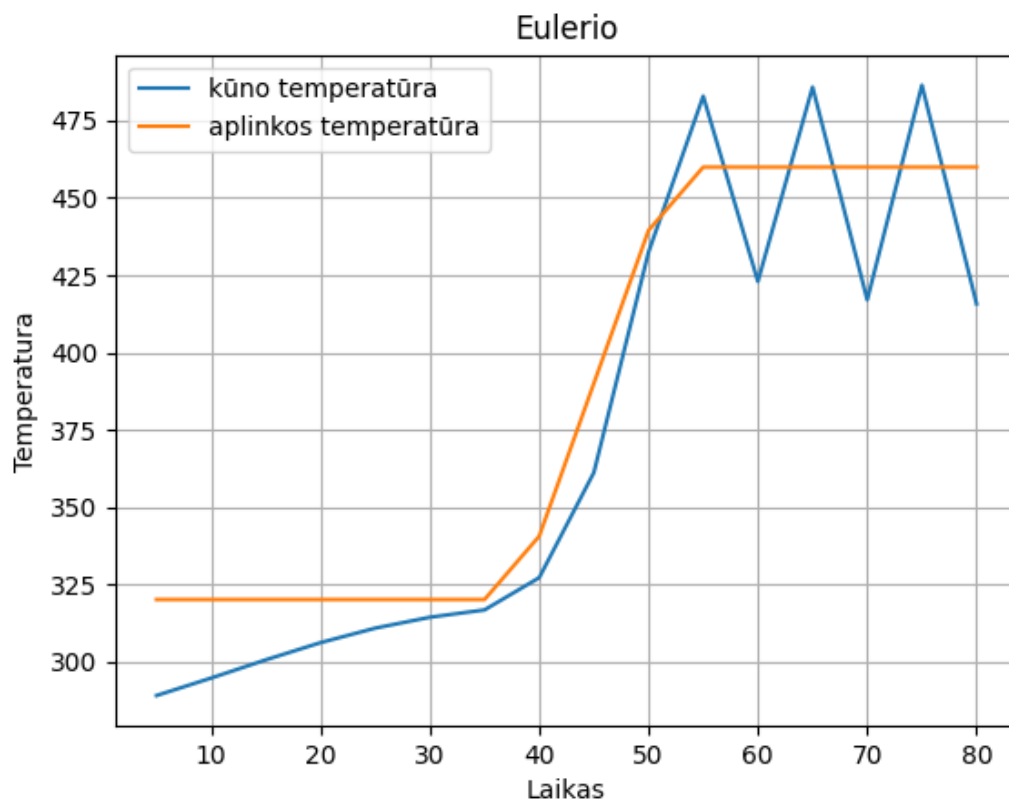


Grafikas. 5

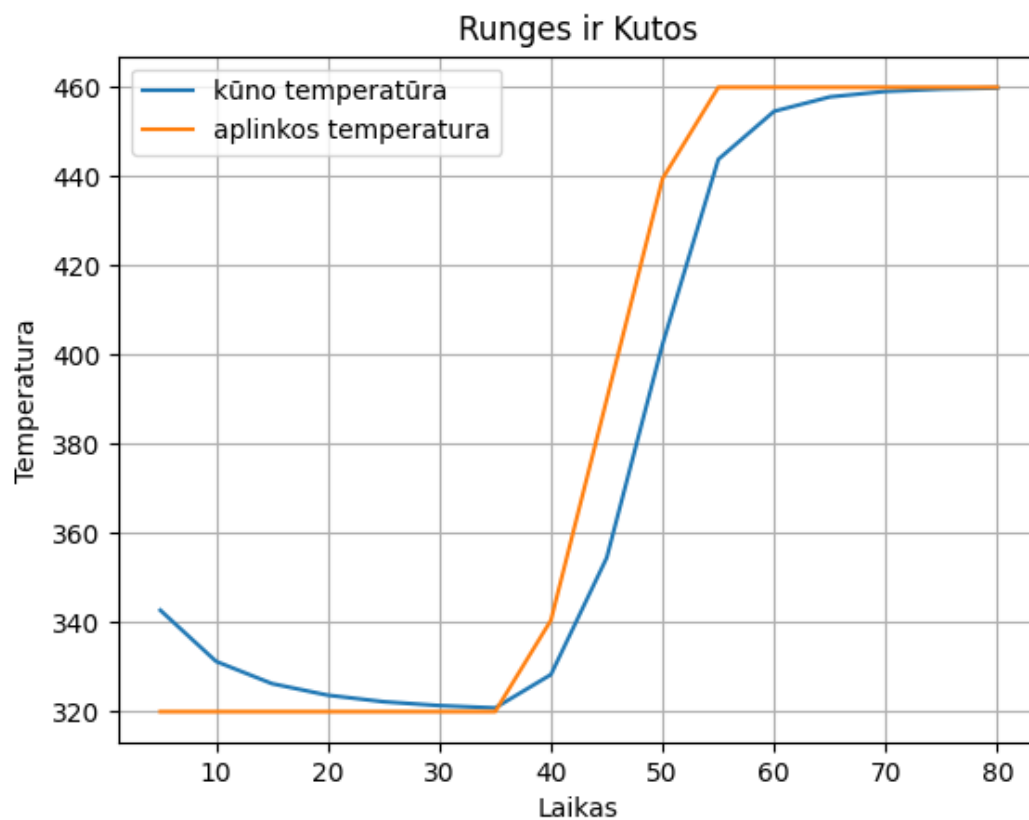


Grafikas 6

2.2.4 Eulerio ir Rungės ir Kutos grafikai(žingsnis 5)

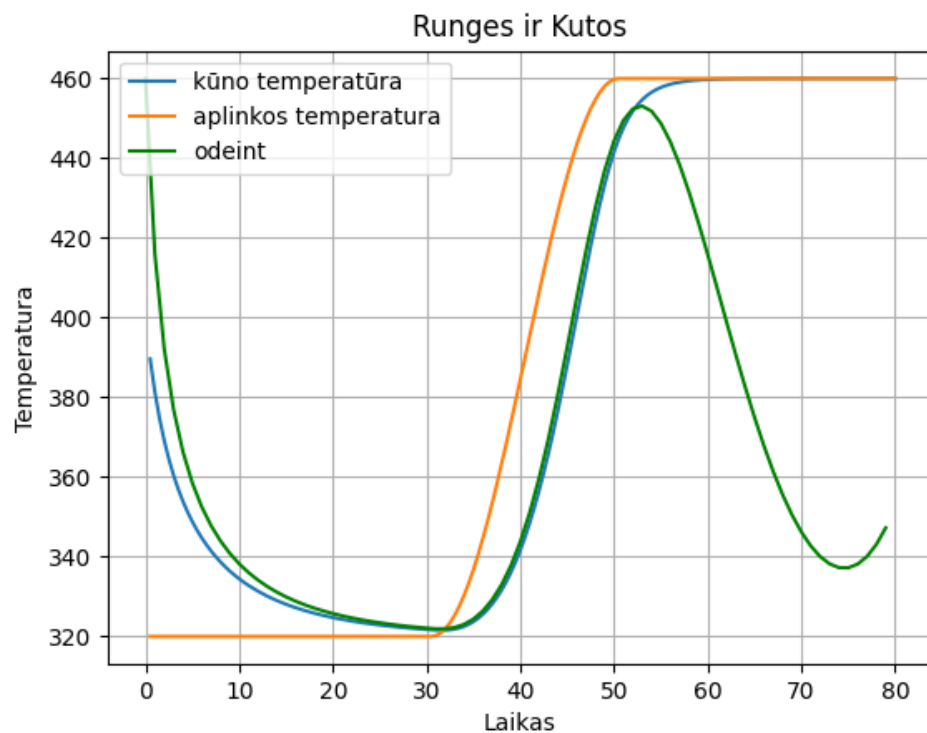


Grafikas. 7

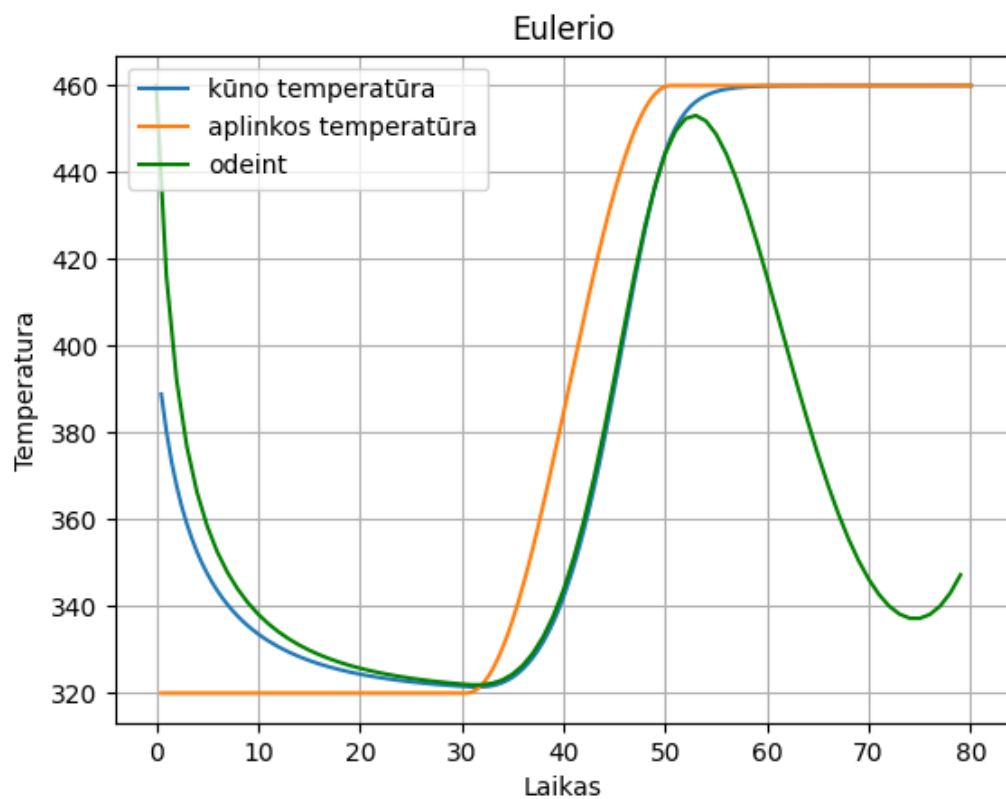


Grafikas. 8

2.3 Patikrinkite gautą sprendinį su MATLAB standartine funkcija ode45 ar kitais išoriniais šaltiniais. Tikrinau su funkcija ODEINT iš scipy bibliotekos



Grafikas. 9

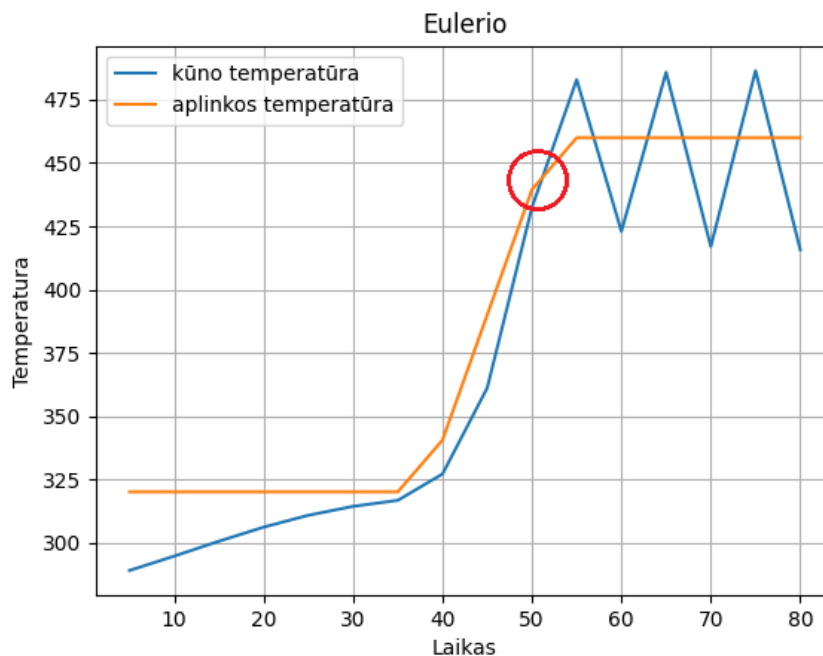


Grafikas. 10

2.4 Atsakymai į pateiktus klausimus uždavinyje

2.4.1 Kada kūno temperatūra pasiekia aplinkos temperatūrą?

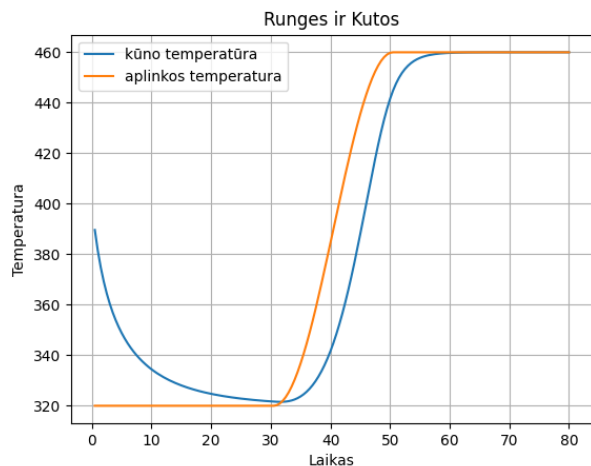
Kūno temperatūra pasiekia aplinkos temperatūrą ties **51s**



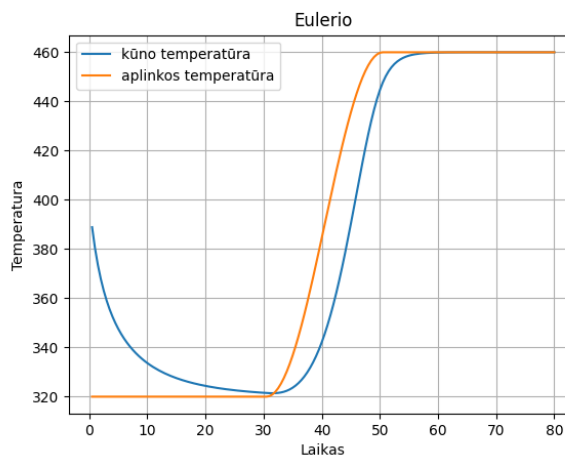
Grafikas. 11

2.4.2 Išspręskite tą pačią lygtį, jeigu pradinė kūno temperatūra lygi T_2

Pradinė temperatūra T_1 (žingsnis 0.5)

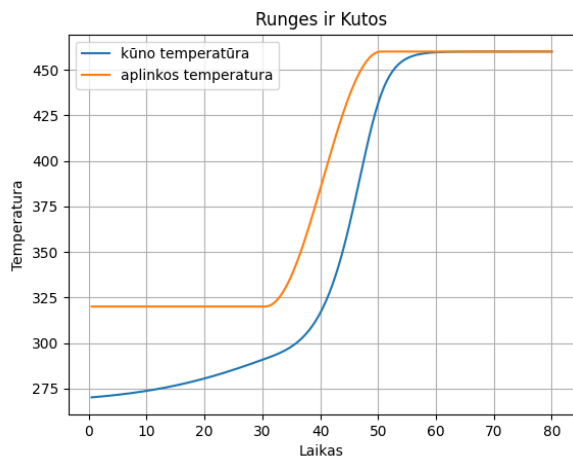


Grafikas. 12

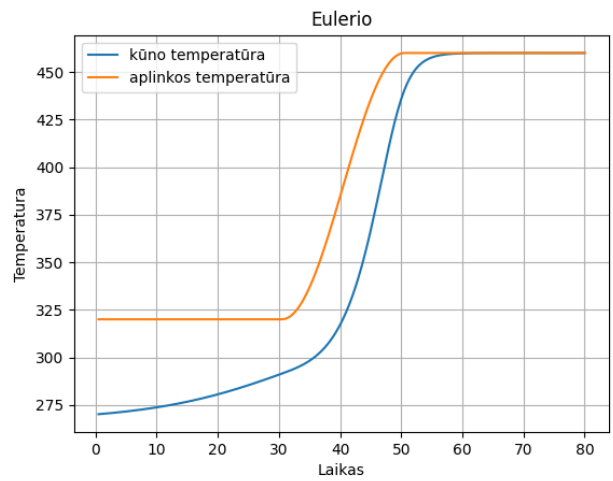


Grafikas. 13

Pradinė temperatūra T_2 (žingsnis 0.5)



Grafikas. 14



Grafikas. 15

2.4.3 Kaip skiriasi su skirtingomis pradinėmis kūno temperatūromis gauti sprendiniai ir jų savybės?

T1 - Kūno temperatūra pasiekia aplinkos temperatūra ties **51s**

T2 - Kūno temperatūra pasiekia aplinkos temperatūra ties **58s**

3. Kodo fragmentas

3.1 Eulerio kodo fragmentas

```
def Euler():  
  
    t = t0 # pradinis laikas  
    TA = TA1 # Pradinė aplinkos temperatūra  
    T = T1 # Pradinė kūno temperatūra  
  
    print("Pradine kuno temperatura:", T)  
    print("Pradine aplinkos temperatura:", TA)  
  
    temperatura = []  
    laikas = []  
    aplinkosTemperatura = []  
  
    while t != tmax: # Kol nepasiekė viso laiko momento  
        if t == ts and TA != TA2: # kaip praeina ts sekundžių ir tol kol nepasiekė  
max temp TA2  
            print("Pradeda kisti aplinkos temperatura")  
            while TA != TA2: # tol kol nepasiekė max temperatūros TA2  
                TA = TA1 + ((TA2-TA1)/2)*(1-np.cos(((math.pi)/20)*(t-ts))) # TA(t)  
dėsnis  
                if TA == TA2:# kai pasieke aplinkos temperatūra  
                    print("pasiekta aplinkos temperatura!", T, TA, t)  
                    TA = TA2  
                k = -0.01 - 0.16*((T-273)/100) - 0.04*((T-273)/100)**2 # k(T)  
dėsnis  
                dt = k * (T - TA) # niutono temperatūros kitimo dėsnis  
                T = T + z * dt # eulerio, kūno temperatūra + žingsnis * išvestinė  
                print("esamos temperatūros:",T, TA)  
                t += z  
  
                temperatura.append(T)  
                laikas.append(t)  
                aplinkosTemperatura.append(TA)  
            else:  
                k = -0.01 - 0.16*((T-273)/100) - 0.04*((T-273)/100)**2 # k(T) dėsnis  
                dt = k * (T - TA) # niutono temperatūros kitimo dėsnis  
                T = T + z * dt # eulerio, kūno temperatūra + žingsnis * išvestinė  
                print("esamos temperatūros:",T, TA)  
                t += z  
  
                temperatura.append(T)  
                laikas.append(t)  
                aplinkosTemperatura.append(TA)
```

3.2 IV eilės Rungės ir Kutos kodo fragmentas

```
def RungėsKutos():  
  
    t = t0  
    TA = TA1  
    T = T1  
  
    print("Pradine kuno temperatūra:", T)  
    print("Pradine aplinkos temperatūra:", TA)  
  
    temperatura = []  
    laikas = []  
    aplinkosTemperatūra = []  
  
    while t != tmax:  
        if t == ts and TA != TA2: # kaip aplinkos temperatūra kinta  
            print("Pradeda kisti aplinkos temperatūra")  
            while TA != TA2: # kol nepasiekė max temp  
                TA = TA1 + ((TA2-TA1)/2)*(1-np.cos((math.pi)/20)*(t-ts))) # TA(t)  
                dėsniis  
  
                if TA == TA2:  
                    print("pasiekta aplinkos temperatūra!", T, TA, t)  
                    TA = TA2  
                k = -0.01 - 0.16*((T-273)/100) - 0.04*((T-273)/100)**2 # k(T)  
                dėsniis  
  
                #pirmas etapas  
                dt1 = k * (T - TA)  
                T11 = T + z/2 * dt1  
                #antras etapas(atgalinis)  
                dt2 = k * (T11 - TA)  
                T2 = T + z/2 * dt2  
                #trečias etapas(vidurinis taškas)  
                dt3 = k * (T2 - TA)  
                T3 = T + z * dt3  
                #ketvirtas etapas(simpsono)  
                dt4 = k * (T3 - TA)  
                T = T + z/6 * (dt1 + 2 * dt2 + 2 * dt3 + dt4)  
                print("esamos temperatūros:",T, TA)  
                t += z  
                if T == TA:  
                    print("Temperatūros susivienodino1:", T, TA)  
                    temperatura.append(T)  
                    laikas.append(t)  
                    aplinkosTemperatūra.append(TA)  
            else: # kol nekinta aplinkos temperatūra  
                k = -0.01 - 0.16*((T-273)/100) - 0.04*((T-273)/100)**2 # k(T) dėsniis  
                #pirmas etapas  
                dt1 = k * (T - TA)  
                T11 = T + z/2 * dt1  
                #antras etapas(atgalinis)  
                dt2 = k * (T11 - TA)  
                T2 = T + z/2 * dt2  
                #trečias etapas(vidurinis taškas)  
                dt3 = k * (T2 - TA)  
                T3 = T + z * dt3  
                #ketvirtas etapas(simpsono)  
                dt4 = k * (T3 - TA)  
                T = T + z/6 * (dt1 + 2 * dt2 + 2 * dt3 + dt4)  
                print("esamos temperatūros:",T, TA)  
                t += z  
                if T == TA:  
                    print("Temperatūros susivienodino3:", T, TA)  
                    temperatura.append(T)
```

```
laikas.append(t)  
aplinkosTemperatura.append(TA)
```

4.IŠVADOS

Sprendinio tikslumas priklauso nuo metodo formulės ir nuo žingsnio

Gauto sprendinio tiksluma galima patikrinti, sulyginus sprendinius, gautus esant tam tikram ir 2 kartus mažesniai žingsniui

IV RK metodo privalumas prieš kitus aukštesniosios eilės metodus toks, kad nereikia analitiškai apskaičiuoti aukštesniųjų eilių išvestinių. Pakanka tik paties ieškomo sprendinio y ir jo pirmosios išvestinės $f(x,y)$ reikšmių;