

Karno karta web site dasturini tuzish

Abrorjon Maxamatsolievich Turgunov

a.turgunov@tuit.uz

Elmrod Toshmuhammad o'g'li Kabirov

Muhammad al-Xorazmiy nomidagi Toshkent axborot texnologiyalari
universiteti

Annotatsiya: Ushbu maqolada mantiqiy funksiyalarni soddalashtirish jarayonini avtomatlashtirishga mo'ljallangan veb-ilova taklif etiladi. Ilova Karno xaritasi (Karnaugh Map) asosida MDNSH va boshqa mantiqiy ifodalarni minimallashtirishni tez va aniq bajaradi. Foydalanuvchi kiritgan qiymatlar server tomonda qayta ishlanib, optimallashtirilgan mantiqiy formula ko'rinishida natija qaytariladi. Tizim PythonAnywhere platformasida joylashtirilgan bo'lib, minimal resurslarda ham barqaror ishlashi va kengaytirilishi mumkinligi bilan ajralib turadi. Taklif etilgan yondashuv mantiqiy sxemalar bilan ishlovchi talabalar, o'qituvchilar va dasturchilar uchun qulay hisoblanadi hamda hisoblash jarayonini avtomatlashtirib beradi.

Kalit so'zlar: Karno xaritasi, Karnaugh Map, mantiqiy funksiya, minimallashtirish, Boolean algebra, digital logic, MDNSH, mantiqiy soddalashtirish, kombinatsion sxemalar, raqamli tizimlar, optimallashtirish, Venn mantiqiy usuli, avtomatlashtirilgan soddalashtirish

Creating a Karno map website program

Abrorjon Makhamatsolievich Turgunov

a.turgunov@tuit.uz

Elmrod Toshmuhammad og'lu Kabirov

Tashkent University of Information Technologies named after Muhammad al-Khwarizmi

Abstract: This article proposes a web application designed to automate the process of simplifying logical functions. The application performs minimization of logical expressions using Karnaugh Maps (K-Maps), including SOP (Sum of Products) and other logical forms, quickly and accurately. User-provided values are processed on the server side, returning the optimized logical formula as the result. The system is hosted on the PythonAnywhere platform, demonstrating stability and scalability even with minimal resources. The proposed approach is convenient for

students, teachers, and developers working with logical circuits, automating the computation process efficiently.

Keywords: Karnaugh Map, Logical function, Minimization, Boolean algebra, SOP (Sum of Products), Digital logic, Simplification of logical expressions, Combinational circuits, Optimization, Web application, Computation automation

Mantiqiy funksiyalar va kombinatsion sxemalar raqamli elektronika va kompyuter arxitekturasi sohalarida muhim rol o'ynaydi. Ushbu funksiyalarni soddalashtirish orqali sxemalarni optimallashtirish, tranzistorlar sonini kamaytirish va hisoblash jarayonini tezlashtirish mumkin. An'anaviy soddalashtirish usullari ko'pincha qo'lda murakkab va vaqt talab qiluvchi hisoblashlarni talab qiladi, bu esa talabalar va dasturchilar uchun noqulaylik tug'diradi. Ushbu maqolada, Karno karta (Karnaugh Map) asosida mantiqiy ifodalarni avtomatik soddalashtirishga mo'ljallangan veb-ilova taklif etiladi, bu jarayonni tez va aniq bajarish imkonini beradi.

Ushbu jarayonlarni interaktiv va amaliy tarzda o'rganish maqsadida "Karno Karta" veb-ilovasi ishlab chiqildi. Ilova foydalanuvchiga 1 dan 6 gacha o'lchamdagi o'zgaruvchilarni kiritish va ularning mantiqiy ifodalarini optimallashtirilgan shaklda soddalashtirish imkonini beradi.

Web-Ilova tuzilmasi va funksional imkoniyatlari

Asosiy menyu va interfeys

Web-site bosh sahifada 2x2 kabi $F(ab)$ qiymatlarni berish mumkin. O'zgaruvchilarni o'zgartirish mumkin. 1 dan 6 gacha o'zgaruvchilar sonini kiritish mumkin. Chunki, ko'proq o'zgaruvchilar bo'lsa, foydalanuvchilar uchun qiyinchilik paydo bo'lishi mumkin.

Karno Karta

F(00) 0

F(01) 1

F(10) 1

F(11) 0

O'zgaruvchilar soni 2 Ok

Soddalashtirilgan MDNSH

$$(A * B) + (B * A)$$

	0	1
0	0	1
1	1	0

1-rasm. Web-siteni bosh sahifasi.

MDNSHni olish

Foydalanuvchi $F(...)$ qiymatlarni kiritgandan so'ng, "Karno Karta" tugmasini bosiladi. Bu amal natijasida kiritilgan qiymatlar serverga yuboriladi va backend tomonidan qayta ishlanadi. Hisoblash jarayoni yakunlangach, optimallashtirilgan

natija saytga qaytariladi. Shu tarzda, foydalanuvchi Karno karta diagrammasini ko‘rishi va minimal disjunktiv normal shakl (MDNF) natijasini olish imkoniga ega bo‘ladi.

```
document.querySelector('.f_changer_much').addEventListener('submit', async function
(e) {
  e.preventDefault()
  let csrftoken = getCookie('csrftoken')

  let formdata = new FormData(this)
  let res = await fetch('http://127.0.0.1:8000/f_qiyamat_much/', {
    method : "POST",
    headers:{
      'X-CSRFToken' : csrftoken
    },
    body: formdata
  })
  let data = await res.json()

  let f_qiyamatlari = document.querySelector('.f_qiyamat_div')
  f_qiyamatlari.innerHTML = ``
  data.f_qiyamat.forEach(item => {
    f_qiyamatlari.innerHTML += `
    <div>
    <span>
    F(${item})
    </span>
    <input type="hidden" name="f_${item}" id="f_${item}" value="0">
    <p id="f_${item}_btn" onclick="f_qiyamat_give('${item}')">0</p>
    </div>
    `
  });
});

})
```

Bu o‘zgaruvchilarga moslab F(...) larni chiqarib beradi.

```
document.querySelector('.f_qiyamatlari').addEventListener('submit', async function
(e) {
  e.preventDefault()
  const csrftoken = getCookie('csrftoken');
  let formdata = new FormData(this)
  let res = await fetch('http://127.0.0.1:8000/f_qiyamatlar/', {
    method: "POST",
    headers: {
      'X-CSRFToken': csrftoken,
    },
    body: formdata
  })
  let data = await res.json()
  let arr = data.mdnf.split('')
  for(let i=0; i<arr.length ; i++){
    if (arr[i]=="") {
      arr[i] = ``
      arr[i-1] = `\\overline{${arr[i-1]}}`
    }
  }
  data.mdnf = arr.join('')
  document.querySelector('#simple_misol').innerText = `$$ ${data.mdnf} $$`

  let f_karno_karta_div = document.querySelector('.f_karno_karta_div')
  f_karno_karta_div.innerHTML = ``
```

```

let satr = true
let ustun = true
data.changer_1_qiymat.forEach(row => {
let div = document.createElement('div')
if(ustun){
let div_main = document.createElement('div')
div_main.classList.add('span_span')
data.changer_2_qiymat.forEach(column=>{
let span = document.createElement('span')
span.innerText = column
div_main.appendChild(span)
})
f_karno_karta_div.appendChild(div_main)
ustun = false
}
let span_div = document.createElement('span')
span_div.innerText = row
div.appendChild(span_div)
data.changer_2_qiymat.forEach(column =>{
let p = document.createElement('p')
p.innerText = document.getElementById(`f_${row}${column}`).value
div.appendChild(p)
})
f_karno_karta_div.appendChild(div)
});
MathJax.typesetPromise()
})

```

Bu funksiya esa, mdnshni va karno kartasini chizib beradi.

Natija sahifasi

Natija sahifasi foydalanuvchiga kiritilgan $F(\dots)$ qiymatlar asosida minimal disjunktiv normal shakl (MDNF) natijasini ko'rish imkonini beradi. Shu bilan birga, sahifa Karno karta diagrammasini vizual tarzda ko'rsatadi, bu esa foydalanuvchiga mantiqiy ifodaning soddalashtirilgan ko'rinishini tushunishni osonlashtiradi. Natijalar interaktiv tarzda taqdim etilgan bo'lib, foydalanuvchi kiritilgan qiymatlarni o'zgartirganda sahifa natijalarni avtomatik yangilaydi.

Karno Karta

F(00)	1
F(01)	0
F(10)	1
F(11)	1

2-rasm. $F(\dots)$ qiymatlarni berish.

Soddalashtirilgan MDNSH

$$A + \overline{B}$$

3-rasm. "Karno Karta" button bosilgan chiqadigan MDNSH.

	0	1
0	1	0
1	1	1

4-rasm. "Karno Karta" si.

Xulosa

Ushbu maqolada "Karno Karta" veb-ilovasi orqali mantiqiy funksiyalarni soddalashtirish jarayoni ko'rsatildi. Ilova foydalanuvchiga 1 dan 6 gacha o'lchamdagi o'zgaruvchilar bo'yicha $F(\dots)$ qiymatlarni kiritish, Karno karta diagrammasini ko'rish va minimal disjunktiv normal shakl (MDNF) natijasini olish imkonini beradi. Backend server tomonida qiymatlar qayta ishlanib, optimallashtirilgan natija foydalanuvchiga tez va aniq tarzda taqdim etiladi.

Taklif etilgan yondashuv talabalar, o'qituvchilar va dasturchilar uchun qulay bo'lib, mantiqiy ifodalarni qo'lda hisoblash zaruratini kamaytiradi va hisoblash jarayonini avtomatlashtiradi. Kelajakda ilovani kengaytirish va murakkab mantiqiy sxemalarni ham qo'llash imkoniyatlari mavjud.

Foydalanilgan adabiyotlar

1. Karnaugh, M. (1953). The Map Method for Synthesis of Combinational Logic Circuits. Transactions of the American Institute of Electrical Engineers.
2. Roth, C. H., & Kinney, L. (2015). Fundamentals of Logic Design. Cengage Learning.
3. Mano, M. M., & Ciletti, M. D. (2017). Digital Design. Pearson.
4. Brown, S., & Vranesic, Z. (2013). Fundamentals of Digital Logic with Verilog Design. McGraw-Hill Education.
5. ythonAnywhere Documentation. (2025). Hosting Python web applications. <https://www.pythonanywhere.com>
6. SymPy Documentation. (2025). Symbolic Mathematics in Python. <https://www.sympy.org>