

MONDRIAN-RUUDUKKOJEN ETSIMINEN

Tekijät: Terhi Juntunen, Aino Kruth, Antti Laaksonen, Mirka Mäkinen, Juha Vartiainen ja Heli Virtanen

Laadittu syksyllä 2022.

Tämä teos on lisensoitu Creative Commons BY 4.0 -lisenssillä (<https://creativecommons.org/licenses/by/4.0/deed.fi>).

Kohderyhmä: Alakoulu, yläkoulu ja lukio

Esitiedot: Kertotaulu, pinta-alan käsite

Oppitunnin kesto: Vähintään 45 min. Osa aiheista käsittelee ratkaisemattomia, mutta helposti muo-
toiltavia ja kokeiltavia matematiikan kysymyksiä, joista innostuessa saa varmasti aikaa kulumaan!

Oppimistavoitteet: Oppitunnin jälkeen opiskelija osaa liittää matematiikan eri osa-alueita taiteisiin ja yleisemmin yhteiskuntaan. Oppitunti kehittää oppilaan hahmottamiskykyä ja kertotaulujen hallinta vahvistuu.

- Ongelmanratkaisu
- Looginen ajattelu
- Matemaattiset ongelmat
- Matematiikan kielellistäminen

Muut tavoitteet:

- Matematiikan kiinnostuksen herättäminen
- Vuorovaikutus ja ryhmätyötaitojen kehittäminen

Sisältö: Oppituntiin kuuluu opettajan kalvopaketti sekä oppilaiden tehtäväsarja. Opettaja voi valita sopivaksi katsomansa osajoukon tehtävistä tai antaa oppilaille kaikki tehtävät pohdittaviksi niin, että oppilaat valitsevat omat suosikkinsa joko aihepiirien tai vaikeusasteen perusteella.

Välineet:

- Konseptipaperia (1×1 cm)
- Suttupaperi
- Lyijykynä
- Pyyhekumi
- Värikynät (punainen, sininen, keltainen ja musta)
- Noppia
- Viivoitin

Toteutus: Opettajan alustus Mondrianista (ks. kalvopaketti Mondrian_pulmat), jonka jälkeen oppilaat pääsevät joko yksin tai pienissä ryhmissä pohtimaan erilaisia ongelmanratkaisuun liittyviä tehtäviä. Lisäksi mahdollisuus tehdä toteutus ohjelmoinnin avulla.

1. MONDRIAN-RUUDUKOT

Mondrian-ruudukko on $n \times n$ -kokoinen ruudukko, joka muodostuu väritetyistä suorakulmioista. Jokaisen suorakulmion väriin tulee olla sininen, punainen, keltainen tai valkoinen. Jokaisen suorakulmion tulee olla eri muotoinen (eli suorakulmiot eivät saa olla yhteneviä) ja suorakulmioita tulee olla vähintään kaksi.

Suorakulmiot ovat *erimuotoisia*, jos ei ole mahdollista saada niitä näyttämään samalta pyörittämällä (eli ne eivät ole *yhteneviä*). Esimerkiksi suorakulmiot 3×5 ja 4×5 ovat erimuotoisia. Suorakulmiot 3×5 ja 5×3 eivät ole erimuotoisia, koska kyseessä on sama suorakulmio pyöritettynä.

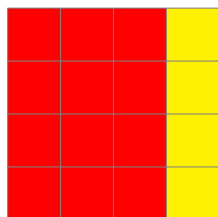
Yksinkertainen tehtävä on etsiä jokin tapa muodostaa suorakulmiot. Vaikeampi tehtävä on etsiä tapa, jossa suurimman ja pienimmän suorakulmion pinta-alan erotus on mahdollisimman pieni. Toinen haaste on etsiä ratkaisu, jossa vierekkäin olevien suorakulmioiden tulee olla eri värisiä.

Mondrian-ruudukoita voidaan etsiä myös tietokoneella. Tällä tavalla voidaan selvittää, mikä on optimaalinen ratkaisu tietyllä n :n arvolla. Tähän mennessä on onnistuttu selvittämään optimaaliset ratkaisut tapauksiin $n \leq 65$. Tätä suuremmat tapaukset ovat vielä avoimia ongelmia.

ruudukon koko n	pienin erotus
3	2
4	4
5	4
6	5
7	5
8	6
9	6
10	8

Optimaaliset tulokset tapauksiin $n = 3, \dots, 10^*$:

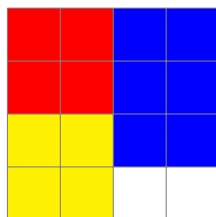
1.1. Esimerkkikuvia ruudukoista. Alla näet esimerkkejä väritetyistä ruudukoista. Joissakin esimerkeissä väritys ei ole kelvollinen, koska siinä on kaksi saman muotoista suorakulmiota.



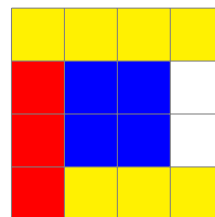
Punainen: $3 \times 4 = 12$
Keltainen $1 \times 4 = 4$
Erotus: $12 - 4 = 8$



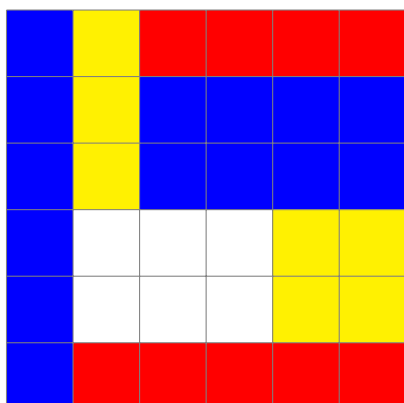
Punainen: $3 \times 3 = 9$
Sininen: $3 \times 1 = 3$
Keltainen: $1 \times 4 = 4$
Erotus: $9 - 3 = 6$



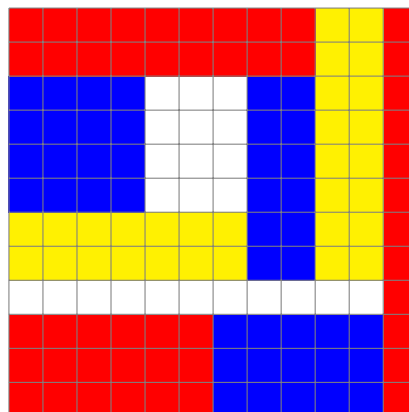
Tämä väritys ei kelpaa, koska on kaksi 2×2 suorakulmiota.



Tämä väritys ei kelpaa, koska on 1×3 ja 3×1 suorakulmiot.



Esimerkkikuva 6×6 -ruudukon värityksestä.



Esimerkkikuva 12×12 -ruudukon värityksestä.

*OEIS-sivuston (<https://oeis.org/A276523/list>) mukaan, viitattu 20.11.2022, voi myös tarkistaa myöhemmin esiintyvällä koodilla.

2. TEHTÄVIÄ

Tehtävä 1. Täytä 4×4 -ruudukko eri värisillä (punainen, sininen, keltainen ja valkoinen) ja muotoisilla suorakulmiolla.

Tehtävä 2. Laske suurimman ja pienimmän suorakulmion pinta-alat ja niiden erotus. Yritä saada erotuksesta mahdollisimman pieni.

Tehtävä 3. Täytä samalla tekniikalla myös isompia neliötä, kuten 5×5 , 6×6 ja 7×7 -ruudukot.

Tehtävä 4. Yritä täyttää neliö (esim. 8×8) käyttämällä vain 4 väriä niin, ettei vierekkäisissä suorakulmiossa ole samaa väriä.

Tehtävä 5. Millaisilla valinnoilla pinta-alojen erotus on pienimmillään?

Tehtävä 6. Tutustu annettuun Python-koodiin ja pohdi, miten se ratkaisee Mondrianin ruudukon niin, että suurimman ja pienimmän suorakulmion erotus on mahdollisimman pieni.

Tehtävä 7, haastava. Luo koodi esimerkiksi Pythonilla, joka ratkaisee Modrianin ruudukon niin, että suurimman ja pienimmän suorakulmion erotus on mahdollisimman pieni.

3. KERTOTAULU-PELI PARIN KANSSA

Kertotaulupelissä on idea harjoitella ja havainnollistaa kertotauluja 1-6. Idea on oppitunnin tehtävien tapaan täyttää annettu ruudukkopohja eli oma pelialusta erikoisilla suorakulmioilla. Pelissä ruudukkojen koko määräytyy heitettyjen nopan silmälukujen mukaisesti. Pelissä pyritään täyttämään oma pelialusta mahdollisimman nopeasti. Pelissä tarvitaan sekä onnea että hahmottamiskykyä. Ole nopeampi kuin parisi ja voita peli!

Välineet:

- Molemmille pareille oma 20×20 kokoinen -ruudukko (liitteenä A on valmis ruudukkopohja, jonka voi tulostaa oppilaille)
- Kyniä
- Kaksi noppaa

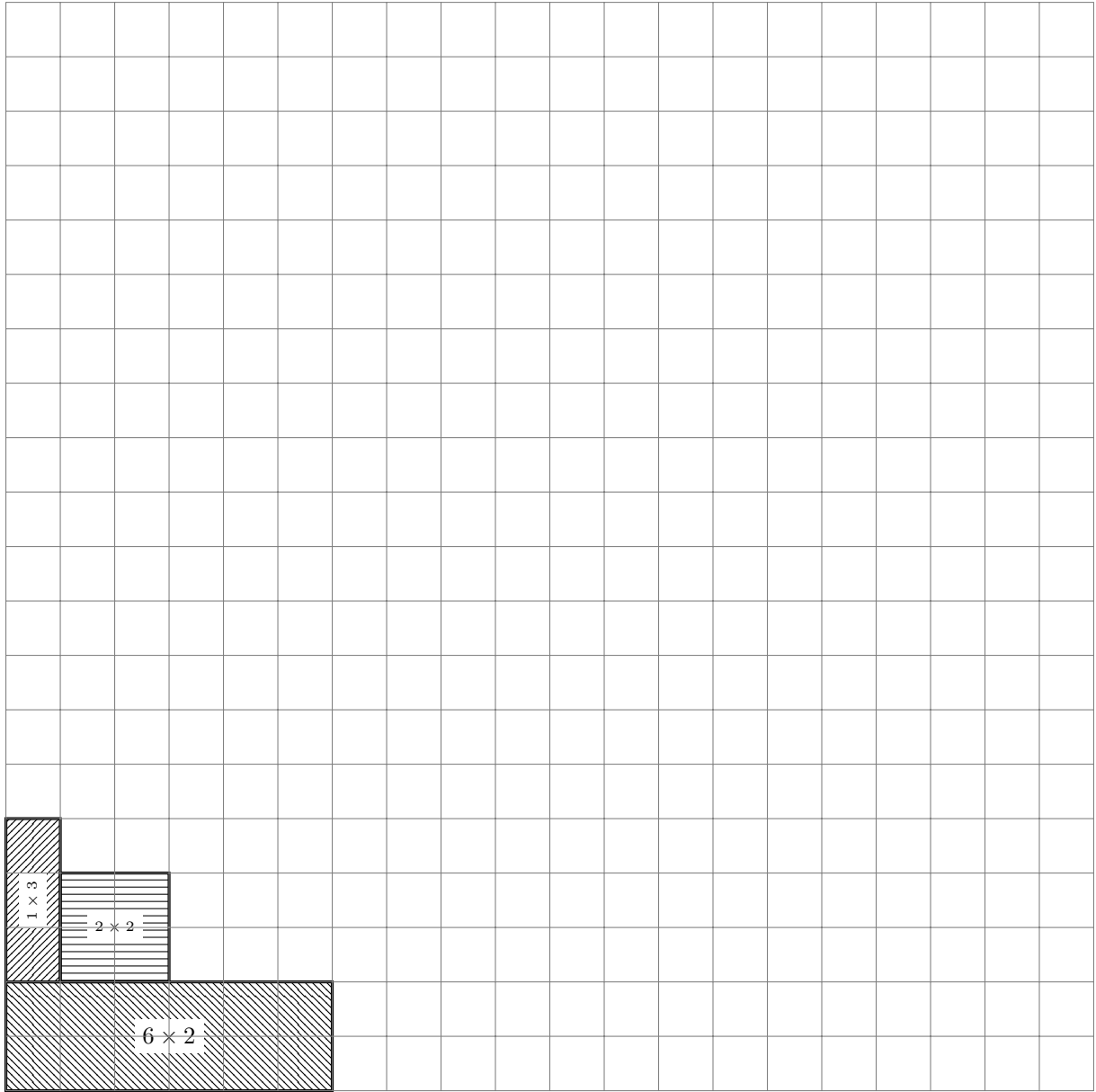
Tavoite: Täytä oma ruudukkosi täyteen ensimmäisenä.

Pelin kulku:

- Molempia noppia heitetään samanaikaisesti.
- Saaduista silmäluvuista muodostetaan suorakulmio. Tavoitteena on täyttää koko ruudukko suorakulmioilla.
 - Esimerkiksi, jos heitettäessä saadaan silmäluvut 3 ja 2. Ruudukkoon lisätään 3×2 -kokoinen suorakulmio.
 - Pelaajan ei tarvitse heti käyttää saamaansa suorakulmiota, vaan heittotulos voidaan kirjata ylös ruudukon viereen.
- Mikäli ruudukkoon ei mahdu heittämiäsi nopan silmäluvun kokoista suorakulmiota, menetät vuorosi.
- Vinkki: Ennen ruudukon täyttämistä, mieti omaa strategiaasi. Voit esimerkiksi alkaa täydentämään ruudukkoa kuvan mukaisesti yhdestä kulmasta alkaen. Muita täyttötapoja voi olla mm. sijoittaa suorakulmiot eri nurkkiin luokitellen suorakulmiot tiettyjen kertotaulujen mukaisesti. Tällöin ruudukon täyttäminen etenee reunoilta kohti keskipistettä.
- Vinkki: Pienet suorakulmiot voi jättää viimeiseksi, jolloin niillä voi "paikata" mahdollisia aukkoja.
- Pelin perusmuodossa ei käytetä värejä. Riittää, että pelialustaan piirretään selkeästi suorakulmiot sekä merkitään niiden koko.

Vaikeampi muunnos pelistä

- Samanväriset suorakulmiot eivät saa koskettaa toisiaan kuin kärjistä.
- Tarvittaessa värien lukumäärää voidaan vähentää, jolloin peli vaikeutuu entisestään.



Esimerkkinä nopanheittojen tulokset: 6 ja 2, 1 ja 3 sekä 2 ja 2 täytettynä ruudukolle.

4. RATKAISUN ETSIMINEN OHJELMOIMALLA

Ratkaisun etsimistä varten voidaan toteuttaa ohjelma, joka käy systemaattisesti läpi mahdollisia ratkaisuja ja etsii optimaalisen ratkaisun. Ohjelma voidaan toteuttaa rekursiivisesti niin, että se valitsee joka askeleella yhden suorakulmion sijainnin. Ohjelman tehokkuutta voidaan parantaa rajoittamalla hakua niin, että ohjelma välttää epäoptimaalisten ratkaisujen muodostamista.

Seuraava Python-ohjelma etsii optimaalisen ratkaisun halutulle $n:n$ arvolle:

```
def fill(y, x, h, w, c):
    for i in range(h):
        for j in range(w):
            grid[y+i][x+j] = c

def empty(y, x, h, w):
    for i in range(h):
        for j in range(w):
            if grid[y+i][x+j]:
                return False
    return True

def search(y, x, count, amin, amax):
    global best
    if amax-amin >= best:
        return
    if y == n:
        if count > 1 and amax-amin < best:
            best = amax-amin
            print("Uusi_tulos:_virhe", best)
            for i in range(n):
                print(grid[i])
    elif x == n:
        search(y+1, 0, count, amin, amax)
    elif grid[y][x]:
        search(y, x+1, count, amin, amax)
    else:
        for h in range(1, n-y+1):
            for w in range(1, n-x+1):
                if not used[h][w] and empty(y, x, h, w):
                    used[h][w] = used[w][h] = 1
                    fill(y, x, h, w, count+1)
                    search(y, x+1, count+1, min(amin, h*w), max(amax, h*w))
                    used[h][w] = used[w][h] = 0
                    fill(y, x, h, w, 0)
                else:
                    break

n = int(input("Anna_n:_"))
best = n*n
grid = [[0]*n for _ in range(n)]
used = [[False]*(n+1) for _ in range(n+1)]
search(0, 0, 0, n*n, 0)
print("Paras_tulos:_virhe", best)
```

Seuraavassa on esimerkki ohjelman suorittamisesta tapaksessa $n = 6$. Aina kun ohjelma löytää aiempaa paremman ratkaisun, se tulostaa näkyviin ratkaisun. Kun ohjelma pysähtyy, se on löytänyt optimaalisen ratkaisun.

```
Anna n: 6
Uusi tulos: virhe 8
[1, 2, 3, 3, 3, 4]
[5, 2, 3, 3, 3, 4]
[5, 6, 3, 3, 3, 4]
[5, 6, 7, 7, 8, 4]
[5, 6, 7, 7, 8, 4]
[5, 6, 7, 7, 8, 4]
Uusi tulos: virhe 7
[1, 2, 2, 2, 2, 3]
[4, 2, 2, 2, 2, 3]
[4, 5, 5, 5, 6, 3]
[4, 5, 5, 5, 6, 3]
[4, 7, 7, 8, 6, 3]
[4, 7, 7, 8, 6, 3]
Uusi tulos: virhe 5
[1, 1, 1, 1, 2, 3]
[4, 4, 4, 4, 2, 3]
[4, 4, 4, 4, 2, 3]
[5, 5, 6, 6, 2, 3]
[5, 5, 6, 6, 2, 3]
[5, 5, 7, 7, 7, 3]
Paras tulos: virhe 5
```

Yllä olevalla ohjelmalla voidaan etsiä nopeasti optimaalinen ratkaisu, kun $n \leq 10$. Optimaalisia ratkaisuja tunnetaan tällä hetkellä tapauksille $n \leq 65$. Kaikki tunnetut optimaaliset ratkaisut on listattu OEIS-sivustolla (<https://oeis.org/A276523/list>). Tätä suuremmissa tapauksissa on avoin ongelma, mikä on optimaalinen ratkaisu.

5. VINKKEJÄ

Saatavilla olevia lautapelejä, jotka sopivat oppitunnin teemaan:

- Blokus
- Second Chance
- Fits
- Mondrian blocks
- Katamino

Mondrian puzzle (Numberphile) <https://www.youtube.com/watch?v=49KvZrioFBO>

Mondrian art puzzle <https://mathcommunities.org/mondrian-art-puzzles/>

Can you solve the Mondrian squares riddle? - Gordon Hamilton <https://youtu.be/AWcY2-FBa9k>

Top 40 Piet Mondrian Paintings <https://www.youtube.com/watch?v=JLNhxf8fmDM>

Problem demonstration <https://demonstrations.wolfram.com/MondrianArtProblem/>

LIITE A. POHJA KERTOTAULU-PELILLE

