



## Pelikerhon helmi-maaliskuu

11.2., **18.2.** ja 25.2. (+4.3.)  
*Jypeli ja C# koulutusta*

4.3. ja 11.3.  
*Apua omien pelien työstöön ja kerhopeli*

18.3. / 25.3.  
*Vierailuluento/Unity-koulutus/verkkopelik.*



**SEPELI RY**



# Jypeli / C#

Peliohjelmoinnin alkeiden tehokoulutusta – osa 2

Pelikerho,  
Jussi Rasku

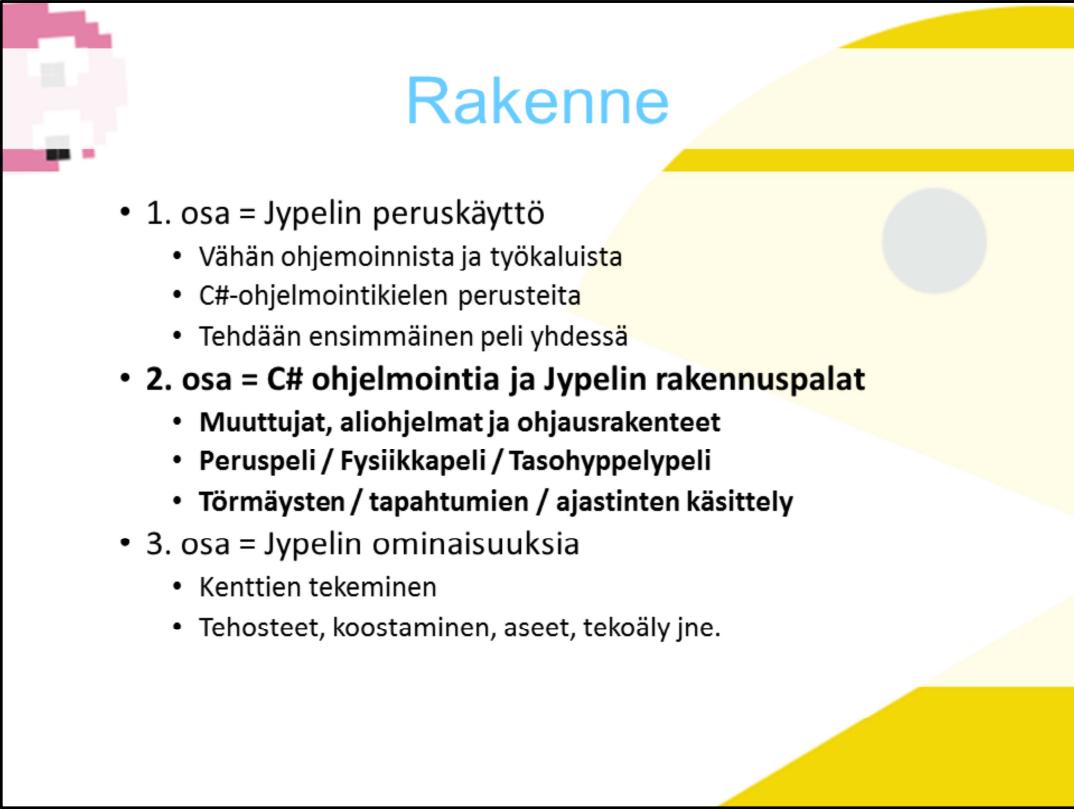
Seinäjoen kaupunginkirjasto  
11.2.2015



Prima  
Power



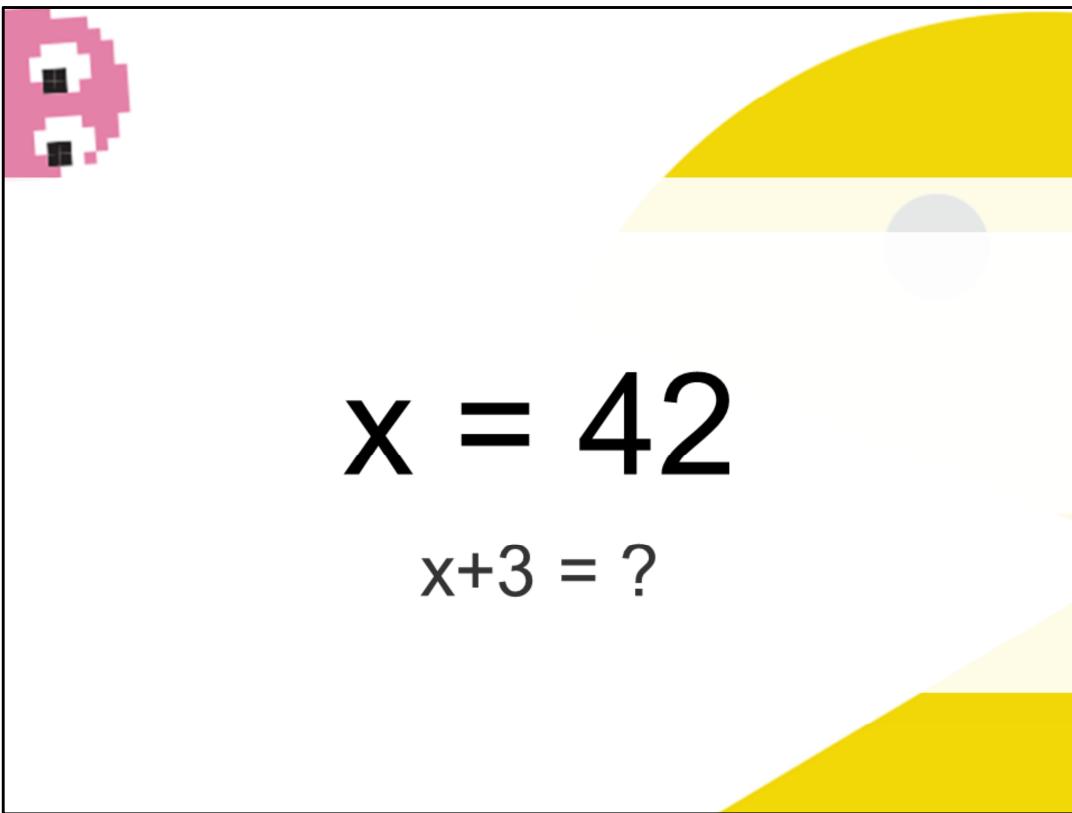
- Kertaa: Tiedotusta myös Facebookissa. Näytä sivu.  
<http://www.facebook.com/sepeliry>
- Tällä kertaa pääosassa C#.



# Rakenne

- 1. osa = Jypelin peruskäyttö
  - Vähän ohjemoinnista ja työkaluista
  - C#-ohjelointikielen perusteita
  - Tehdään ensimmäinen peli yhdessä
- 2. osa = C# ohjelointia ja Jypelin rakennuspalat
  - Muuttujat, aliohjelmat ja ohjausrakenteet
  - Peruspeli / Fysiikkapeli / Tasohyppelypeli
  - Törmäysten / tapahtumien / ajastinten käsittely
- 3. osa = Jypelin ominaisuuksia
  - Kenttien tekeminen
  - Tehosteet, koostaminen, aseet, tekoäly jne.

Emme ehkä ehdi tutustua eri peleihin Peruspeli / Fysiikkapeli / Tasohyppelypeli



Matematiikasta tuttu muuttuja

Voi käyttää säästämään kirjoittamisen vaivaa / monimutkaisissa laskutoimituksissa.

Matematiikasta poiketen x:n arvo voi muuttua!

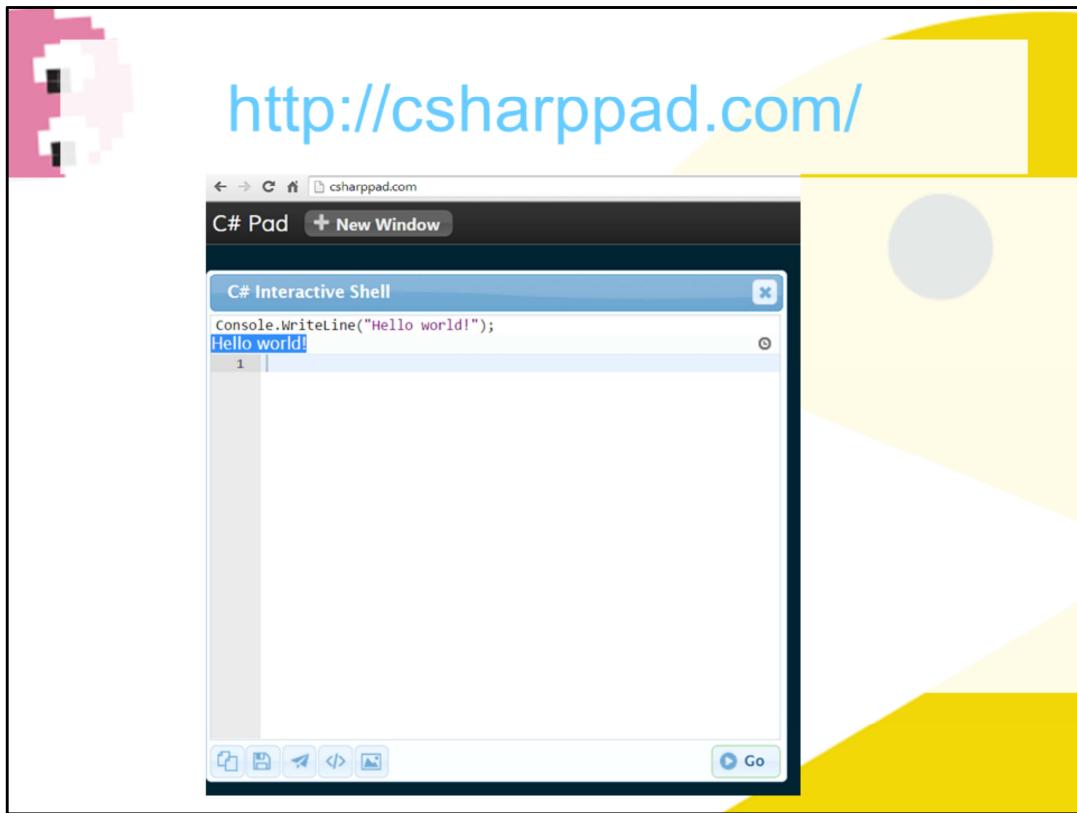


## Muuttujat

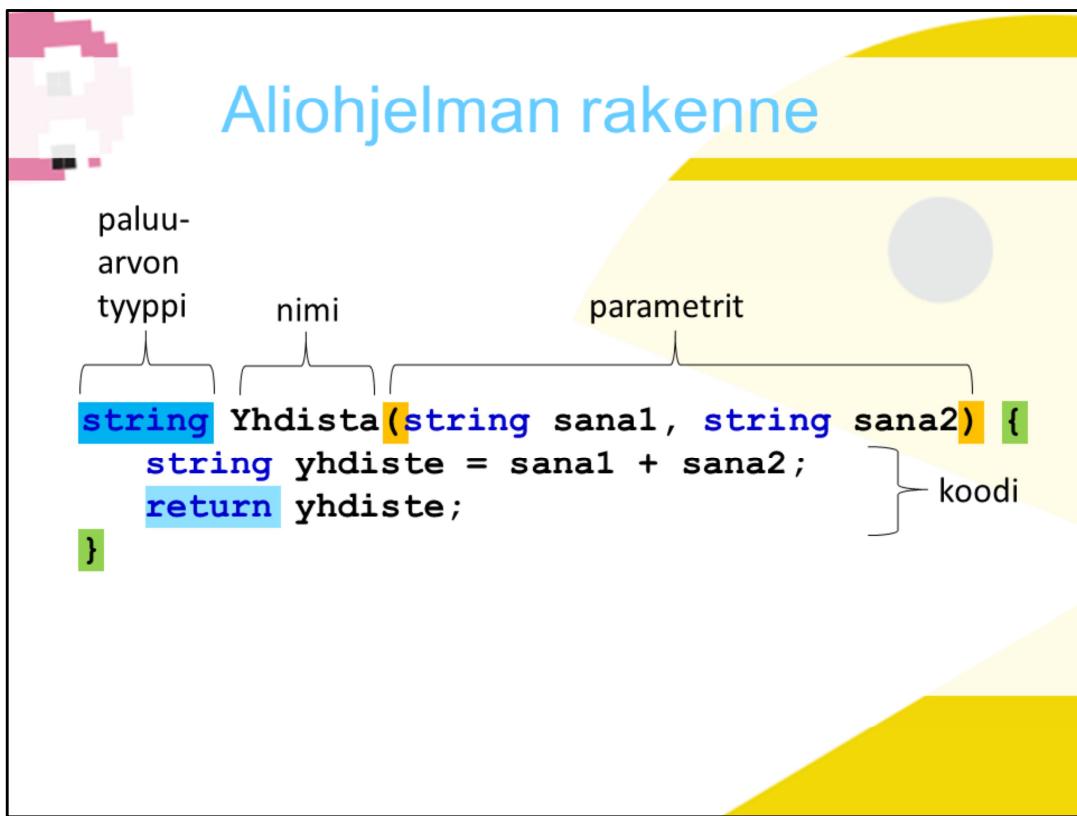
```
// Muuttuja on "laatikko", jonne voit säilöä
int kokonaislukuja = 1;
double desimaalilukuja = 3.14;
string tekstia = "Hello world";
bool totuusarvoja = true; // false

// Olioita
PhysicsObject pallo =
    new PhysicsObject(10,10);
Add(pallo);
```

Mainitse nimeäminen: Hyvät kuvaavat muuttujannimet ja EI ÄÄKKÖSIÄ.  
PhysicsObjectin kohtaan tullessa, voi kysyä ja tarkistaa, että muistetaanko kuinka viime koulutus kerran 1. pelissä lisättiin peliolio peliin.



Tehdään muutama esimerkkilasku csharppadillä.  
Esim paljon pitää maksaa jos 630€ näytönohjain on 20% alennuksessa.  
Kerro, että itse käytät REPLiä (selitä repl) taskulaskimen sijaan alvariinsa.



**Aliohjelmat** ovat ohjelman osia, joista jokainen hoitaa jotain omia erityistä tehtäväänsä. Voidaan sanoa, että ohjelmat koostuvat aliohjelmista, eli ne ovat ohjelmien rakennuspalkoita. Kun ohjelma voidaan jakaa pienempiin osiin (aliohjelmiin), sen hallinta helpottuu. Muun muassa virheiden jäljittäminen helpottuu.

- Käytetään etenkin silloin kun tehdään sama asia monta kertaa.
- Säästää vaivaa kirjoittamiselta, eli copy/pastelta (joka aiheuttaa vähintään 50% kaikista bugeista)
- Vähemmän koodia -> vähemmän virheitä

Aliohjelmilla on nimi, parametrit ja paluuarvo. Seuraavassa esimerkki aliohjelmasta, joka yhdistää kaksi merkkijonoa.

Aliohjelman odotettavaa vastausta sanotaan aliohjelman *paluuarvoksi*. Esimerkiksi aliohjelma `SummaaLuvut` voisi palauttaa kysyjälle syötetyn kokonaisluvun summan.

Paluuarvon tyyppi kirjoitetaan aina ennen aliohjelman nimeä, ja se kertoo, minkä tyyppistä tietoa aliohjelmasta saadaan ulos. Jos aliohjelma palauttaa jonkin arvon, on siinä oltava "return"-lause.

Jos palautusarvon tyyppiksi on määritetty jotain muuta kuin void, on itse palautusarvo palautettava aliohjelmasta erityisellä return-lauseella. Return-lauseen käyttö on yksinkertaista: kirjoita vain `return` ja paluuarvon tyyppin kanssa samaa tyyppiä oleva muuttuja tai lauseke.

Aliohjelman nimen perään sulkujen sisällä kirjoitettu lista kertoo sen *parametrit*, eli mitä aliohjelmalle syötetään. Jokainen parametri koostuu tyypistä ja nimestä. Nämä parametrit ovat käytettäväissä aliohjelmassa samaan tapaan kuin tavalliset muuttujat. Jos aliohjelmalle ei halua määrittää parametreja, voi listan jättää tyhjäksi. Sulut on silti jätettävä.

Merkityksellisin osa aliohjelmassa on sen runko, joka kirjoitetaan aaltosulkujen väliin. Runko voi sisältää mitä tahansa ohjelmointikielen lauseita.

**Kysy: Miten esitellään aliohjelma, joka ei palauta mitään arvoa?**

**Vast; void**



## Aliohjelman kutsuminen

```
// Alustetaan ensin merkkijonot  
string hei = "Hei ";  
string nimi = "Jussi";  
// Kutsutaan Yhdista()-aliohjelmaa  
string tervehdys = Yhdista(hei, nimi);  
MessageDisplay.Add(tervehdys);
```

Kysy: Mitä ohjelma tulostaa. Kokeillaan VC#:lla



If-lause on rakenne, jonka avulla voidaan tehdä jotakin, mikäli ”jokin ehto” on totta. Vastaavasti jos tämä ”jokin ehto” ei ole totta, tehdään jotain muuta. Esimerkiksi ehto voi olla ”ulkona sataa” (vastaus on aina kyllä tai ei).

**Jos ulkona sataa**

    ollaan sisällä

**Muuten**

    mennään ulos pelaamaan jalkapalloa

Kun ohjelmassa haluamme tehdä eri asioita riippuen esimerkiksi käyttäjän syötteistä tai aliohjelmien parametreista, tarvitsemme ehtolauseita

## Ehtolause 1/3

```
// Luetaan käyttäjän syöttämä luku
string syote = ikkuna.InputBox.Text
int luku = int.Parse(syote);

// Tarkistetaan ehto
if (luku == 1) {
    MessageDisplay.Add("Pariton");
}
else if (luku == 2) {
    MessageDisplay.Add("Parillinen");
}
else {
    MessageDisplay.Add("Ei hajua!?!?");
```

Jos vaihtoehtoja on useita, voit käyttää else if:iä ja elseä.

Ehtolauseen haaroista suoritetaan TASAN YKSI.

Elseiffejä voi olla vaikka kuinka monta. Elsejä tasan yksi (tehdään jos mikään muu ehto ei täsmää).

Kysy: Mitä ohjelma tulostaa. Kokeillaan...

HUOM! Tarvitset kalvolla olevan LueSyote(InputWindow ikkuna) aliohjelman lisäksi Begin:iin:

```
InputWindow lukulkkuna = new InputWindow("Anna luku");
lukulkkuna.TextEntered += LueSyote;
Add(lukulkkuna);

void LueSyote( InputWindow ikkuna ) {
    string vastaus = ikkuna.InputBox.Text;
    // tehdään jotain vastauksella
}
```



## Ehtolause 2/3

```
// Luetaan käyttäjän syöttämä luku
string syote = ikkuna.InputBox.Text
int luku = int.Parse(syote);

// Tarkistetaan ehto
if (luku % 2) {
    MessageDisplay.Add("Parillinen");
}
else {
    MessageDisplay.Add("Pariton");
}
```

Kysy: Mitä ohjelma tulostaa. Kokeillaan...

## Ehtolause 3/3

```
// Luetaan käyttäjän syöttämä luku
string syote = ikkuna.InputBox.Text
int luku = int.Parse(syote);

// Tarkistetaan
if (luku > 2) {
    MessageDisplay.Add("Suurempi kuin 2");
}
else if (luku != 2) {
    MessageDisplay.Add("Ei 2");
}
```

Kysy: Mitä ohjelma tulostaa. Kokeillaan...

# Silmukan rakenne

```
alustus           lopetuseheto    päivitys  
for (int i = 0 ; i < luku ; i++) {  
    // Toistettava koodi täähän  
    ...  
}
```

Ohjelmoinnissa tulee usein tilanteita, joissa samaa tai lähes samaa asiaa täytyy toistaa ohjelmassa useampia kertoja. Tällöin on järkevää käyttää jotain *toistorakennetta*. Toistorakenteista käytetään usein myös nimitystä *silmukat*.

Yleisin silmukka on for-silmukka. Sitä käytetään, kun suoritusten lukumäärä on tiedossa enakkoon, esimerkiksi tulostetaan kymmenen kertaa "Moi".

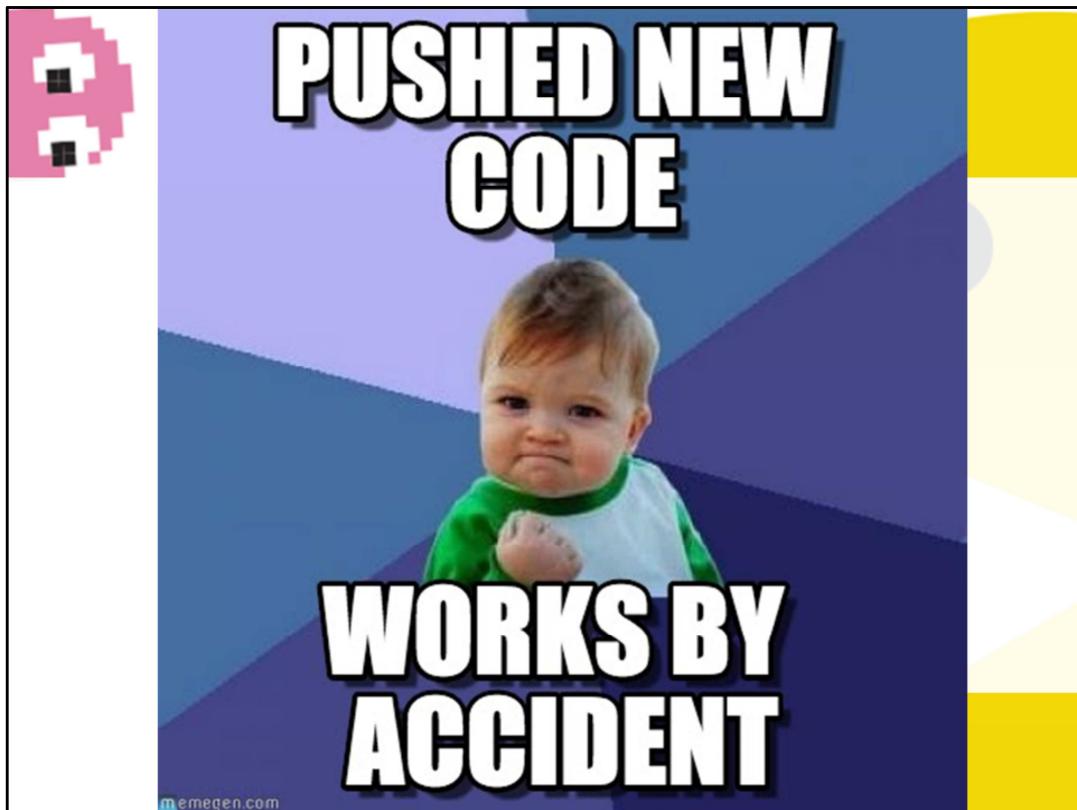


## Silmukka esimerkki

```
// Luetaan käyttäjän syöttämä luku
string syote = ikkuna.InputBox.Text;
int luku = int.Parse(syote);

// Luetellaan kaikki luvut siihen asti
for (int i = 0 ; i < luku ; i++) {
    MessageDisplay.Add( i.ToString() );
}
```

Kysy: Mitä ohjelma tulostaa. Kokeillaan...



Klikkaa näyttääksesi Asteroids videon.

Koodataan vastaava seuraavaksi.

1. Lisätään kolmionmuotinen GameObject alus. Sille joku kiva väri.
2. Pistetään se liikkelle kuuntelemalla KeyUp:ia ja työntämällä sitä (Push) vakiovektorilla. Alus kuuntelijan parametriksi.
3. Lisätään käänäminen. Kuuntelijaan kulma double parametrina. Samoin alus.
4. Mikä ei toimi? Vast: Ei työnnä sinne minne nokka osoittaa. Korjataan työntämällä aluksen suuntaan (alus.Angle).
5. Luultavasti alus menee 90 astetta väärään suuntaan +Angle.RightAngle korjaa..
6. Lisätään ase ja pistetään se ampumaan. Kokeillaan ja käännetään. Pistetään asteroidit tuhoutumaan kun niitä ammutaan.
7. Lisätään asteroideja. Tutustutaan GameObjekteihin. Pistetään ne satunnaiseen paikkaan ja lyödään (Hit) satunnaiseen suuntaan.
8. Lisätään reunat Level.CreateBorders. Zoomataan niin, että ne juuri ja juuri näkyvät (Camera.Zoom)
9. Loopataan luodut reunat foreach:illa ja poistetaan törmäysvaste päältä (IgnoreCollisionResponse). Pistetään törmäys muuttamaan paikkaa n. -1.75 kertaa seinän paikka (teleporttaa asteroidin / aluksen toiselle puolelle kenttää).
10. Jos ehditään, pistelasku, aseen tulinopeus ym. hienosäätöä.

## Ohjeita ja apua

<http://bit.ly/pelikurssi>

- harjoitustehtävät
- ohjeet ja vinkit
- esimerkkipeli oppaita

Tämä paistumaan jos on loppoaikaa.