

Toteutus

Projekti rakentuu Game- luokan ympärille. Luokka Huolehtii pelaajien vuorotuksesta, pelin aloituksesta, pelilaudan päivityksestä ja voittajan julkistamisesta. Pelilauta ja pelaajat sisältävät omat tietorakenteensa, joiden palveluja ne tarvittaessa kutsuvat. Luokka hierarkian pohjalta löytyvät kortteja kuvaavat luokat, joiden säilytyksestä vastaavat itsetoteutetut CardList ja WeatherList.

Luokat

Game

Vastaa pelin kulusta. Pelin elinkaari sisältää vähintään kaksi kierrosta, joiden aikana Game kutsuu pelaajia pelaamaan vuoronsa. Sisältää kaksi Player rajapinnan toteuttavaa oliota ja yhden pelilautaa kuvaavan Board olion.

Players

GwentPlayer

Ihmisen pelaajan toiminnasta vastaava luokka. Pelaajan vuoron alussa kysyy pelattavaa korttia ja tarkistaa onko annettu komento hyväksyttävä. Suorittaa halutun "liikkeen" pelilaudalla ja tekee muutokset pelaajan käteen. Toteuttaa rajapinnan Player ja sisältää pelaajan numeron, nykyisen pelilaudan kopion, pelaajan kortit sisältävän käden ja scannerin.

GwentAI

Tekoäly pelille. Pelaa joko ihmistä tai toista tekoälyä vastaan. Tekee peli päätöksensä laudalta ja omasta kädestään saatavan datan perusteella. Päätöksenteon jälkeen pelaa halutun kortin laudalle tai skippaa. Toteuttaa rajapinnan Player ja sisältää pelaajan numeron, nykyisen pelilaudan kopion ja omat korttinsa sisältävän käden.

Board

Pelin tilannetta kuvaava luokka. Pelaajat pelaavat vuorotellen laudalle kortteja. Board tarjoaa toimintoja, joilla laudan linjojen sää tilannetta voidaan muuttaa. Antaa pelaajille tietoja pistetilanteesta ja pitää kirjaa erävoitoista. Sisältää pelaajien erän skippaamista kuvaavat muuttujat, erävoitot, kummallekin pelaajalle oman BoardSide olion ja säätilaa kuvaavan taulukon.

Hand

Pitää kirjaa korteista, jota pelaajalla on vielä pelattavana. Käsi voi sisältää korkeintaan kymmenen korttia, joista kukin voi kuulua joko Card- tai weatherCard- luokkaan. Ihminen pelaajalle käsi tarjoaa yksinkertaisen indeksointiin perustuvan kortin haun. Tekoälylle on tarjolla enemmän toimintoja, joilla se voi pyytää esimerkiksi

pienimmän tai suurimman kortin tietyltä riviltä tai tiettyyn riviin vaikuttavan sään jos sellainen löytyy kädestä. Sisältää kädessä olevien korttien määrästä kertovan muuttujan, Pistekortit sisältävän CardListin ja sääkortit sisältävän WeatherListin.

Containers

CardList

Itsetoteutettu listana toimiva tietorakenne. Sisältää Card-luokan ilmentymiä ja tarjoaa toiminnallisuuden lisäykseen, poistoon ja indeksin perusteella hakuun. Voi myös ilmoittaa kokonsa ja tyhjentää sisältönsä. Kaikki muut operaatiot poistoa lukuunottamatta vakioaikaisia. Poisto pahimmassa tapauksessa luokkaa $O(n)$.

WeatherList

Itsetoteutettu listana toimiva tietorakenne. Muuten täysin sama kun CardList, mutta sisältää pistekorttien sijasta sääkortteja. Samat operaatiot samoilla aikavaativuuksilla.

BoardSide

CardList olioista koostuva tietorakenne, joka kuvaa pelaajan laudan puolikasta. Sisältää vähän toiminnallisuutta ja luottaa paljon CardListin operaatioihin. BoardSide voi palauttaa haluttua riviä kuvaavan CardListin, lisätä halutulle riville annetun kortin ja tyhjentää itsensä. Sisältää kolme CardList-oliota.

Cards

Card

Kuvaa pelin aikana laudan riveille pelattavia pistekortteja. Oma pistemäärän, rivin jolle kortti kuuluu ja boolean muuttujan, joka kertoo onko kortti sään vaikutuksen alaisena. Vähän toiminnallisuutta. Osaa palauttaa oman rivinsä ja pisteensä, joiden määrä riippuu säästä.

WeatherCard

Kuvaa pelin aikana pelilaudalle pelattavia sääkortteja. Hyvin yksinkertainen sisällöltään ja toiminnoiltaan. Voi kertoa mille riville kortin voi pelata. Säällä on sitä kuvaava symboli, joka näytetään pelilaudalla, kun sää pelataan.

Parannettavaa

Alkuperäiseen peliin verrattuna tässä projektissa käytettävä versio on hyvin yksinkertainen. Pistekorteilla ei ole mitään toisiin kortteihin vaikuttavia ominaisuuksia ja pelissä ei ole muita erikoiskortteja kuin sää. Peliä olisi mahdollista parantaa lisäämällä kyseisiä ominaisuuksia, mutta niiden käyttöön otto vaatisi melko paljon uudistuksia koodiin. En ole tyytyväinen kaikkiin rakente ratkaisuihin joita tällä hetkellä joistain luokista löytyy, joten jonkinlainen refaktorointi olisi paikallaan.