

Nuottipiirturi - Yleissuunnitelma

kurssille Ohjelmointistudio2 (MOOC)

Niina Saarelainen

ei opiskelijanumeroa

(koulutus: pianonsoitonopettaja ja musiikkiteknologian maisteri)

8.2.2017

Yleiskuvaus

Toteutetaan ohjelma, jolla voidaan kirjoittaa yksinkertaisia sävelmiä nuottiviivastolle ja esittää nuotit tekstimuodossa. Ohjelmalle syötetään halutut nuotit tiedostosta tekijän itsensä suunnittelemassa formaatissa. Graafinen ulkoasu tulee olemaan sama kuin tehtävämallissa.

Projektisivulla https://plus.cs.hut.fi/studio_2/2017/studioprojekti/283/ olevista vaatimuksista toteutetaan kaikki paitsi tahtirajan ylittävät nuotit. Myös jotakin aika-arvoja jätetään toteuttamatta.

Tässä voisi olla tarpeen perustella, miksen valinnut projektisivulla olevaa syöteformaattia. Jos ohjelmaa käyttäisi joskus joku muusikko, niin hän ajattelee melodioita mielumminkin jatkuvana nauhana, kuin monimutkaisina yksittäisinä tapahtumina. Syötteen kirjoittamisen pitäisi olla nopeaa: kuin naputtelisi pianon koskettimia. Jos joka tapahtumassa pitäisi miettiä missä tahdinosassa ollaan (esim. projektisivun kohta: "aloitushetki : 4/32"), väittäisin että 90% muusikoista ei halua käyttää ohjelmaa. Syöteenanto pitää olla lähtöisin musiikkitermistöstä ja datan määrä pitää olla minimaalinen tiedoston luojalle. Ohjelman tehtävä on konstruoida tieto siitä, mikä on aloitushetki; se ei ole loppukäyttäjän tehtävä.

Aika-arvoista mukaan tulevat perusarvot kokonuotista kahdeksasosanuottiin. Ulkopuolelle jäävät siis kuudestoistaosanuotti ja sitä lyhyemmät nuotit sekä triolit ym. jaot. Kuudestoistaosa jätetään pois, koska projektisivulla puhutaan yksinkertaisista sävelmistä. Vaikka alkuperäisteoksessa olisi kuudestoistaosia, voi kaikki nuottiarvot kertoa kahdella ja toteuttaa kappale mukaantulevalla aika-arvovalikoimalla. Lisäksi väittäisin että kuudestoistaosat eivät ole merkkipohjaisesti toteutettuna kovinkaan luettavia: Palkit sekoittuvat herkästi nuottiviivoihin.

Pisteellisistä aika-arvoista toteutetaan pisteellinen puolinuotti ja pisteellinen neljäsosanuotti. Pisteellistä kokonuottia ei voi toteuttaa jos ja kun en toteuta tahtiviivan ylittäviä nuotteja (yleisin tahtilaji on 4/4 ja kuuteen laskettava nuotti ei mahdu tahtiin). Pisteellisen kahdeksasosan parina taas lähes aina on kuudestoistaosanuotti, jota en toteuta, joten pisteellinen kahdeksasosakin on tarpeeton.

Tauoista ei kannata toteuttaa koko- ja puolitaukoa, koska tällä grafiikan granulariteetilla niitä ei saada erottumaan toisistaan eikä ylipäättään saada näyttämään mustalta suorakulmiolta. Koko- ja puolitauko korvataan sopivalla määrällä neljäsosataukoja toteutuksessa. Muita yllä esiteltyjä nuotteja vastaavat tauko-aika-arvot toteutetaan.

Äänimäärää ei ole rajoitettu. Ainoa rajoite on että, samanaikaisesti soivien äänten tulee olla yhtä pitkiä keskenään. Lyriikat toteutetaan siten, että nuottien alla voi olla maksimissaan yksi rivi sanoja, ei siis toteuteta ns. kakkossäkeistön sanoja. Kappale voi olla myös ilman sanoja. Lyriikat alkavat ensimmäisen nuotin kohdalta (ensimmäisessä tahdissa saattaa olla taukoa, jos sävellys alkaa kohotahdilla).

Käyttöliittymän luonnos

Aluksi Eclipse-konsolille (tai komentoriville jos ohjelma suoritetaan sieltä käsin) tulee ohjeet kuinka toimia. Sitten käyttäjä saa valita syötetiedostolistauksesta tiedoston, joka tehdään nuoteiksi. Käyttäjän tulee kirjoittaa tiedoston nimi. Jos ei kirjoita mitään, ohjelma valitsee oletussyötteen.

Käyttäjä voi myös milloin tahansa (ohjelman käynnissäolon ulkopuolellakin) tehdä uusia syötetiedostoja input-kansioon.

Enterin painamisen jälkeen syöte piirtyy nuottiviivastoille. Lopuksi ohjelma kysyy millä nimellä ohjelman tuotos eli kappale, eli nuotinnos talletetaan. Jos painaa vain Enteriä, mitään ei tallennu.

Tiedostot ja tiedostoformaatit

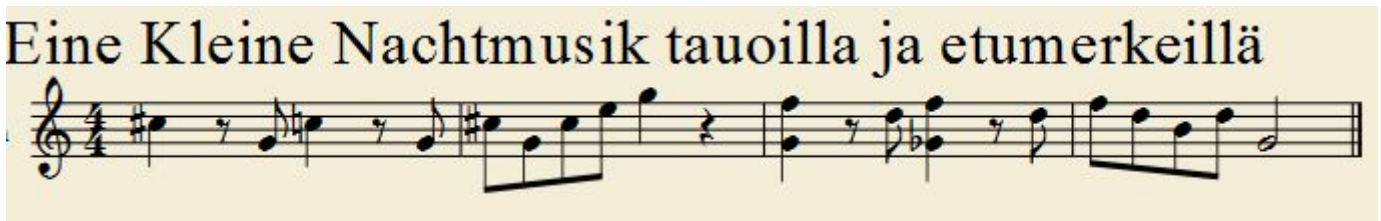
Tieto luetaan UTF-8-muotoisesta tekstitiedostosta. Seuraavassa voi vertailla oikeaa nuottikuvaa ja nuottidatan esittämistä ohjelman syöteformaatissa.



#Nimi Eine Kleine Nachtmusik

#4 c2-. g1 c2-. g1 c2 g1 c2 e2 g2-- <f2-.,d2-.,g1-.> d2 <f2-.,d2-.,g1-.> d2 f2 d2 h1 d2 g1--

Esitellään myös tauot ja etumerkkien (#, b, §) syntaksi:



#Nimi Eine Kleine Nachtmusik tauoilla ja etumerkeillä

#4 c#2- z g1 c2- z g1 c#2 g1 c#2 e2 g2- z- <f2-,g1-> z d2 <f2-,gb1-> z d2 f2 d2 h1 d2 g1--

Tunnisteita on kaksi. #Nimi-tunnisteen jälkeen on kappaleen nimi. #4 kertoo että kappale on 4/4-tahtilajissa. Myös ylennysmerkki on #, mutta se ei sekoitu tunnistemerkkiin, koska tunnistemerkkin pitää olla rivin ensimmäinen merkki. Tämän jälkeen nuotteja/taukoja luetellaan peräkkäin ilman tahtiviivoja. Ohjelma laskee tahtiviivojen paikat. Vaakaviivojen määrä nuotin perässä kertoo pituuden, esim. kokonuotti on "c1----" (lasketaan neljään), puolinuotti on "c1--" (lasketaan kahteen). Pisteellisissä nuoteissa on piste.

Jos usea nuotti soi yhtäkaa, kirjoitetaan ne "<>"-merkkien sisään, erotetaan toisistaan pilkulla, eikä käytetä välilyöntiä. Numerot 1 ja 2 viittaavat yksiviivaiseen ja kaksiviivaiseen oktaavialaan ¹. Tauot piirretään vakioapaikkaan y-akselilla, niille ei tarvitse määrittää korkeutta.

Nuottien välissä pitää olla välilyönti, saa olla useampikin. Rivinvaihtoja saa tehdä haluamiinsa kohtiin tiedostoa, ne eivät vaikuta nuottikuvaan. Ekstrakommentteja saa tehdä sekaan, esim. #B-osa. Ne eivät näy nuoteissa, mutta helpottaa pidempien kappaleiden kirjoittamista nuoteiksi tai ainakin mahdollisen virhepaikan löytämistä.

Kappaleessa ei ole pakko olla sanoja. Jos on, ne kirjoitetaan tyyliin:
#Sanat Uk-ko Noo-a Uk-ko Noo-a. Taasen # pitää olla rivin alussa.

Valmis musiikkikappale, eli nuotinnos tallennetaan tiedostoon, jonka pääte on .txt. Käyttäjä saa valita tallennettavan tiedoston nimen ja ohjelma liittää perään tiedostopäätteen .txt. Hieno feature olisi jos ohjelma osaisi valittaa jos yrittää kirjoittaa samannimisen tiedoston päälle. Toteutan tämän, jos jää aikaa.

Järjestelmätestaussuunnitelma

Testattavia asioita:

* nuottikuva vastaa syötettä: 1) yhtäkään elementtiä ei puutu, 2) rivinvaihdot toimivat sekä graafisesti että tallennuksen kannalta, 3) etumerkit (ylennys-, alennus- ja palautusmerkit) ovat oikeaoppisesti kuvassa, 4) tahtiviivat tulevat oikeisiin paikkoihin, 5) yhtäkaiset tapahtumat ovat allekkain, 6) kahdeksasosapari on loogisesti ja piirtämisen kannalta vaikein, sille useita testejä liittyen varsien ja palkin piirtämiseen, mahduttamiseen viivastolle jne, 7) lyriikat ovat oikeissa paikoissa.

* vääränlainen syöte: 1) nuotin nimi ei sallittu, 2) nuotin pituus ei sallittu, 3) tahti ei tule täyteen, 4) sanoitus pidempi kuin sävellys. Lyhyempi sallitaan, koska lopussa voi olla instrumentaaliosuus, 5) käyttäjän kirjoittamaa tiedostoa ei ole, 6) tallettaessa ohjelman pitäisi napata mahdolliset virhetilanteet, kuten epäkelpo tiedoston nimi: "1/8-nuotteja".

* raja-arvotestausta: 1) testataan piirtoalueen reunanuotteja, varsinkin varsien pituudet voivat mennä herkästi yli kahdeksasosapareissa, 2) hyvin pitkä syöte, 3) nollan mittainen syöte 4) mahtuuhan mahdollisimman tilaavievä syöte nuottiviivastolle, eli testataan pelkillä kahdeksasosapareilla.

* testataan että tallennettu tiedosto vastaa kuvaruudulla näkynyttä.

* suorituskyky ei liene tässä ohjelmassa pullonkaula. Yksinkertaisten kappaleiden vaatima tietomäärä ei kaatane ohjelmaa, eikä ohjelmaa ole tarkoitus käyttää kuin yhden henkilön kerrallaan.

* vakaustestaus ei liene tarpeellinen. Ohjelmaa on tarkoitus käyttää lyhyitä aikoja kerralla.

* käytettävyydestä voin tehdä parilla ulkopuolisella henkilöllä (yksi tietojenkäsittelijä ja yksi muusikko).

¹ <http://www2.siba.fi/muste1/index.php?id=8&la=fi>