

Harjoitustehtäväkierros 4: Tiedon tallennus, grafiikka ja multimedia

T-110.1100 kurssihenkilökunta

deadline Torstai 14.4.2011 18:00

Johdanto

Tällä tehtäväkierroksella käsittelemme multimedian, grafiikan ja tiedon tallentamisen perusteita. Ensimmäiset 2 tehtävää teroittavat mieleesi vektorij- ja rasterigrafiikan eron. Jälkimmäinen tehtävä lisäksi päästää sinut tutustumaan XML-pohjaisten formaattien ihanaan maailmaan.

Kolmannessa tehtävässä pääset pohtimaan teoreettisemmalta pohjalta Internetin www-istymistä ja ohjelmien siirtymistä käyttöjärjestelmästä selaimen.

Tehtävä palautetaan Rubyric-järjestelmään kurssin Noppa-sivuilla löytyvien ohjeiden mukaisesti. *Kiinnitä huomiota palautuksen formatointiohjeisiin. Väärästä tiedostoformaattista tai merkistöstä sakotetaan pisteitä!*

1 Rasterigrafiikka (6p)

Tehtävänannon yhteydessä on annettu tiedosto pohja.bmp. Tehtävänäsi on piirtää assarille jotain käyttämättä piirtotyökaluja (Paint, Photoshop, gimp). Sensijaan sinun tulee käyttää heksaeditoria ja editoida käsin. Heksaeditorina voit käyttää windowsissa HexEdittiä, jota ei tarvitse asentaa erikseen¹. Unix-ympäristöön löytyy lukuisia editoreita. Esimerkiksi GHex on varsin yksinkertainen.

Tiedosto on 7x7 Windows Bitmap -tiedosto. BMP:ssä värit esitetään kolmena peräkkäisenä tavuna järjestyksessä Sininen, Vihreä, Punainen. Katsoessasi tiedostoa heksaeditorilla näet alussa otsikkotietoja. Niiden jälkeen

¹<http://www.physics.ohio-state.edu/~prewett/hexedit/>

	1.pikseli	2.pikseli	3.pikseli	4.pikseli	5.pikseli	6.pikseli	7.pikseli	padding
7. rivi	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	00 00 00
7. rivi	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	00 00 00
6. rivi	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	00 00 00
5. rivi	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	00 00 00
4. rivi	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	00 00 00
3. rivi	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	00 00 00
2. rivi	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	00 00 00
1. rivi	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	FF FF FF	00 00 00

alkaa itse kuvan data, joka on helppo havaita. Valkoinen on heksaesitysmuodossa FF FF FF. Kuvaformaatin rajoituksista johtuen jokaisen rivin lopussa on ns. paddingia 3 tavua, jotka ovat nollia.

Sinun tulee käyttää väriä, jonka komponentit ovat *opiskelijanumeromod256*, 00 tai FF. Vähintään yhden värin pitää käyttää opiskelijanumerostasi saatavaa arvoa. Googlen laskuri osaa auttaa sinua kun haet käskyllä "12345 mod 256 in hex".

Tiedoston kuvaosan rakenne on oheisen taulukon mukainen. Huomaa, että rivit menevät väärinpäin eli ensimmäisenä tiedostossa on alin rivi. Pikselit menevät intuitiivisesti vasemmalta oikealle. Rivin vaihtumisen huomaa kolmesta padding-tavusta, joiden arvo on 00. Kannattanee ensin suunnitella mitä väriä haluaa mihin pikseleihin ja sitten huolellisesti asettaa oikeat tavut tiedostosta. Ruutupaperi säästää suurelta päänsäryltä.

Laita piirroksesi kotisivutilallesi verkkoon ja laita palautukseen URL, josta assari voi hakea sen (muista testata toimivuus!). Laita myös vastaukseen myös käyttämäsi väri ja mitä piirros esittää. Monimutkainen ei tarvitse olla. Hymiö riittää ihan hyvin. Jos teet ei-esittävän taideteoksen (s.o. sotkua), voit saada maksimissaan 3p.

Huomaathan, että rukkasimme tiedostoa siten että näemme jos olet muokannut sitä kuvankäsittelyohjelmalla. Ohjelmalla muokattu tiedosto tuottaa 0p.

2 Vektorigrafiikka(8p)

Kirjoita oheisen dokumenttityyppimäärittelyn (DTD) mukainen taidekokoelma, johon teet ainakin yhden teoksen. Määritelmän mukaan kokoelmassa pitää olla ainakin yksi teos ja teoksessa ainakin yksi nelikulmio, ympyrä tai viiva.

```
!ELEMENT taidekokoelma (teos+)>
<!ELEMENT teos (artisti, (ympyra|viiva|nelikulmio)+)>
<!ELEMENT ympyra (x, y, sade, vari)>
<!ELEMENT nelikulmio (x, y, pituus, leveys, vari)>
<!ELEMENT viiva (x, y, loppux, loppuy, vari)>
<!ELEMENT artisti (nimi, opiskelijanumero)>
<!ELEMENT vari (#PCDATA)>
```

```
<!ELEMENT x (#PCDATA)>
<!ELEMENT y (#PCDATA)>
<!ELEMENT loppux (#PCDATA)>
<!ELEMENT loppuy (#PCDATA)>
<!ELEMENT pituus (#PCDATA)>
<!ELEMENT leveys (#PCDATA)>
<!ELEMENT sade (#PCDATA)>
<!ELEMENT nimi (#PCDATA)>
<!ELEMENT opiskelijanumero (#PCDATA)>
```

DTD on myös verkossa osoitteessa <http://www.cse.hut.fi/u/jsuvileh/jotime/taidekokoelma.dtd>. Siihen täytyy viitata tekemässäsi XML-tiedostossa.

A Laita palautukseen tähän kohtaan ainakin yhden taideteoksen kuvaus XML-kielillä. (3p)

B Tee luento-esimerkin kaltainen XSLT-muutos, jossa muunnat XML:n XHTML-sivuksi, jossa taideteos tai -teokset ovat SVG-elementteinä. (4p) Laita tuotoksesi kotisivutilallesi (tarkista taas toimivuus!). Laita tähän kohtaan XML-tiedoston URL-osoite. (5p)

Voit tarkistaa rakentamasi XML-dokumentin oikeellisuuden selaimella. Kun avaat .xml-päätteisen tiedoston joka viittaa dokumenttityypimäärittelyyn, selain hyvin todennäköisesti kertoo sinulle virheistä.

Koska SVG-elementtien upottaminen XHTML-tiedostoon on selainriippuvaista ja muutenkin vaikeaa, on ohessa esimerkki. Sen on testattu toimivan XSLT-muunnoksen kanssa Firefox-selaimessa. Chromessa dokumentti näkyy, mutta dynaamisesti muodostettuna ei. Jäämme odottamaan HTML5:n mukana tulevaa täydellistä tukea.

Huomaa, että esimerkin tapauksessa <vari>-tagin sisällä oli englanninkielinen värin nimi, joita voi antaa SVG:lle. Esimerkiksi red, blue, maroon tai black ovat valideja värejä².

Jos et keksi mistä kenkä puristaa, ja selain ei vaan näytä sitä mitä haluat, erilaiset XSLT-prosessorit kuten Xalan voivat muuntaa xml-tiedostosi. Tällä tavoin saat tiedoston, josta näet miltä xslt-muunnettu sivusi näyttää selaimelle.

```
<?xml version="1.0" encoding="UTF-8"?>
```

²Yksi DTD-formaatin puute on datan tyyppin määrittäminen. XML Schema tarjoaisi paremman tuen tälle mutta on mielestämme vaikeampi lukea.

```

<html xmlns="http://www.w3.org/1999/xhtml"
xmlns:svg="http://www.w3.org/2000/svg">
<head>
  <title>Taidekokoelma</title>
</head>
<body>
<h1> Taideteos jonka piirsi Teemu Teekkari(42)</h1>

<svg xmlns="http://www.w3.org/2000/svg" width="100%" height="100%"
version="1.1">
<rect width="30" height="30" x="40" y="40" fill="red">
</rect>
<circle r="10" cx="20" cy="40" fill="blue">
</circle>
<line x1="10" y1="10" x2="80" y2="80" stroke="brown" style="stroke-width:3;">
</line>
</svg>
</body>
</html>

```

3 HTML5(6p)

- Mitä lisäyksiä edellisiin HTML 4.01:een ja XHTML 1.1:een verrattuna uuteen HTML5-standardiin on tehty? Käsittele ainakin Multimediajärjestelmätluennon kalvoissa taulukoituja mediatyyppejä. (2p)
- Miten nämä HTML-kieleen lisätyt asiat toteutetaan/toteutettiin ennen niiden tukea kielen tasolla? (1p)
- Arvio HTML5:ttä luennolla esitettyjen datan siirtoformaatin vaatimusten kannalta. Mitä puuttuu? (Vaikka kalvoissa onkin annettu osviittaa, selvitä kalvojen sisältöä syvemmälle) (3p)

Materiaalia

- <http://www.w3schools.com/> W3Schoolsin tutoriaalit mm. XML-tekniikoihin liittyen.
- <https://noppa.tkk.fi/noppa/kurssi/t-110.1100/etusivu> Kurssin kotisivut

- <http://www.cse.tkk.fi/Tietoliikenne/Opinnot/Ohjeet/> T-110 -kurssien yleisiä ohjeita
- <http://opetuki2.tkk.fi/p/top/> Teekkarin ohjauspakki
- Internetin hakupalvelut (jos et löydä, googlaa)