

**Helsingin yliopisto**  
**Tietojenkäsittelytieteen laitos**  
**Digitaalisen median tekniikat, 18.5.2005**

1. Hypertekstissä yleisesti solmuja yhdistetään toisiinsa erityyppisten linkkien avulla. Selvitä lyhyesti miten HTML hyperteksti suhtautuu yleiseen hypertekstimalliin ja mitkä ovat HTML:n tavat yhdistää erilaisia resursseja. (10p)

Yleisessä hypertekstissä linkit voivat olla kaksisuuntaisia. HTML:n linkit ovat yksisuuntaisia. Yleisessä hypertekstissä voi olla semantiikaltaan erilaisia linkkejä esim. kannattava tai vastustava. HTML:n a-linkkeihin (Anchor-linkit) ei voi liittää mitään semantiikkaa, sen sijaan dokumentin otsakeosaan sijoittuviin link-linkkeihin liittyy semantiikan esittävä tyypimäärittely, eri tyyppisiä on kuitenkin varsin vähän..

HTML:n tarjoaa joukon tekniseltä luonteeltaan erilaisia kytkentätapoja

- a-elementti: yleinen dokumentin runkoon upotettu näkyvä linkki, jonka näpäytys tuo selaimen elementtiin kytketyn dokumentin
- link-elementti: dokumentin otsakeosaan sisältyvä linkki, mm. tekijän kytkentä, tyylitiedoston kytkentä, skriptitiedoston kytkentä, kaavion kytkentä,...
- img-elementti kuvien liittämiseen
- form action: lomakkeen käsittelijän ja sen generoiman sivun liittäminen
- object: muiden resurssien esim. applettien, äänitiedostojen, videoiden yms liittäminen
- ---- yllä olevat tuottavat täydet pisteet ----
- background määreellä voi määritellä taustakuvan
- kuvakartat: kuvaan määritelty alue toimii linkkinä

2. Selvitä lyhyesti mitä tarkoitetaan tyylimäärittysten periytymisellä CSS:n yhteydessä. Käytä apuna pientä esimerkkiä. (10p)

Tyylimäärittysten periytymisellä tarkoitetaan sitä, että elementin tyyliminaisuuden arvo periytyy kaikille elementtiin sisältyville elementeille, ellei näille erikseen määritellä uutta arvoa samalle tyyliminaisuudelle.

(4p)

CSS:ssä periytyvät vain ns. pintamääreet esim. merkkikoko, -laji ja väri. Taustamääreet kuten taustakuva eivät periydy. (3p)

esim: (3p)

```
<div style="color:blue; margin-left:30px; font-size:1em;">
<p>Tässä tulee sinistä tekstiä paitsi nyt tuleekin<span style="color:red">punainen</span>.</p>
<p style="font-size:1.5em">Lisää sisennetty ja <span style="font-weight:bold;">
isompi</span></p>
</div>
```

font-size=1em periytyy ensimmäiselle p-elementeille samoin ensimmäiselle span-elementille.  
color=blue periytyy kummallekin p-elementeille ja toiselle span-elementille.

Tässä voisi piirtää kuvan dokumenttipuusta ja selittää asiaa sen avulla.

3. WWW-sivulla listataan kymmenen henkilön nimeä. Kuhunkin nimeen liittyy lisätietopainike, jolla saadaan näkyviin henkilön yhteystiedot. Yhteystiedot näytetään 'samalla' sivulla ts. sivulla, jonka sisältö on muuten sama kuin lähtösivun, mutta lisäksi näkyy valitun henkilön yhteystiedot esimerkiksi henkilön nimen alapuolella.
  - a) esitä lyhyesti periaate, miten tällainen toiminnallisuus voidaan toteuttaa JavaScriptillä.
  - b) esitä lyhyesti periaate miten tällainen toiminnallisuus voidaan toteuttaa PHP:llä
  - c) vertaile JavaScript ja PHP-ratkaisuja (3+3+4p)
    - a) Vaihtoehtoja on erilaisia. Yksinkertaisin lienee ratkaisu, jossa osoitetiedot ovat valmiiksi oikeilla paikoillaan dokumentissa, mutta elementti on määritelty näkymättömäksi (visibility attribuutti). Lisätietopainikkeeseen liitetty Javascript muuttaa visibility attribuutin arvon ja tekee elementin näkyväksi. Katso kurssilla ollut menu-esimerkki.
    - b) Tässä ratkaisussa lisätietopainike on lomakkeen submit-nappi. Lomakepyynnön yhteydessä välitetään parametrina kenestä halutaan lisätietoja. PHP generoi uuden sivun, johon se lisää kysytyn henkilön osoitetiedot.
    - c) Vaihtoehdon a sivu on ehkä hieman vaihtoehdon b sivua suurempi. Sivun sisältöä javascript-koodin ja kaikki osoitetiedot. Toisaalta vaihtoehdon b sivu sisältää kutakin henkilöä varten lomakkeen piilokenttineen. Tietoliikenteen kannalta vaihtoehto a on selvästi parempi ratkaisu, tarvitaan vähemmän yhteydenottoja palvelimeen. Vaihtoehto b on selain- ja asetusriippumaton eikä edellytä javascriptin päällä oloa.
4. Ovatko seuraavat väittämät oikein vai väärin? Oikea vastaus 1 piste, väärä -1 pistettä ja tyhjä 0 pistettä.
  - a) Div-elementillä voidaan kontrolloida mm. dokumentin palstoitusta.  
Oikein. Div-elementti kokoaa yhteen joukon elementtejä. Tämä kokonaisuudelle voidaan määrittellä sijainti ja leveys, jolloin muodostuu palsta.
  - b) Div-elementti ei voi sisältää muita html-elementtejä.  
Väärin. Sen on nimenomaan tarkoitus sisältää muita elementtejä.
  - c) Span-elementtiä voidaan käyttää esimerkiksi yksittäisen sanan värin määrittelyyn.  
Oikein. Span eristää juoksevista tekstistä osan, jolle voidaan antaa omat tyylimääreet..
  - d) Elementin class-attribuutti määrittelee tyyppihierarkian.  
Väärin. Class on tyyplejä varten määritelty attribuutti eikä sillä määriteltyjen luokkien välillä ole mitään hierarkiaa.
  - e) Elementin class-attribuutin arvona annettavilla luokkanimillä määritellään dokumentin luokkarakenne JavaScriptiä varten.  
Väärin. Luokkia käytetään CSS:ssä.
  - f) Rakenne `<option value="10" checked>10</option>` on sallittu HTML:ssä, mutta ei XHTML:ssä  
Väärin, mutta eri tavoin kuin tehtävää määriteltäessä oli ajateltu. Option-määreen oikea attribuutti on selected. Tässä ajettiin kuitenkin takaa sitä, että XHTML:ssä attribuutilla pitää aina olla arvo eli checked="checked" . Kaikki vastaukset (myös tyhjä) tuottivat yhden pisteen.
  - g) JavaScript ohjelma saa kutsuparametrien arvot selville muuttujien \$GET ja \$POST kautta.  
Väärin. JavaScript ohjelma toimii selaimessa eikä sille välitetä kutsuparametreja.
  - h) JavaScript ohjelmalla on pääsy vain palvelimen levyllä oleviin tiedostoihin.

Väärin. Ei ole pääsyä palvelimen eikä asiakkaan normaalitiedostoihin.

i) CGI-palvelupyynnön yhteydessä välitetään aina käyttäjän käyttäjätunnus.

Väärin. Ei välitetä ellei välitystä ole erikseen speksattu ja tunnusta kysytä käyttäjältä.

j) Koska http-protokollassa yhteys katkeaa aina palvelupyynnön suorittamisen jälkeen, on jokin istunnon ylläpitämiseen liittyvä tieto aina kierrätettävä selaimen kautta.

Oikein. Vaikka jotkut selaimet / palvelimet pitävätkin yhteyden hengissä ei tämän varaan voi mitään rakentaa. Huom: Sessiot välittävät tunnuksensa selaimen kautta (eväste tai url:n täydennys). Istunnon tunnistus asiakkaan IP:n avulla ei toimi. Usein saadaan vain proxy:n IP.

k) Selain antaa virheilmoituksen, jos samaan elementtiin liittyy kaksi ristiriitaista tyylimääritystä.

Väärin. Valitsee paikallisemmän.

l) JavaScript ohjelma voi muokata muitakin kehysrakenteisen sivun kehyksiä kuin sitä, mistä se käynnistettiin.

Oikein. Esimerkiksi pienoiskuvan näpäys yhdessä kehyksessä vaihtaa isomman kuvan toiseen kehykseen.

(12p)