CS-C3170 Web Software Development : Exam

February 2018

- All answers should easily fit on a single answer sheet.
- Write on each paper your **name**, your **student number** including any checking characters, the **code and name of this course**, **examination hall**, **date**, the **number of papers returned** and **your signature**.
- **If the question number is followed by an asterisk ✪, you should select all options that apply.** If there is no asterisk, select only one.
- No additional materials, calculators etc. allowed.

Questions

HTML & CSS

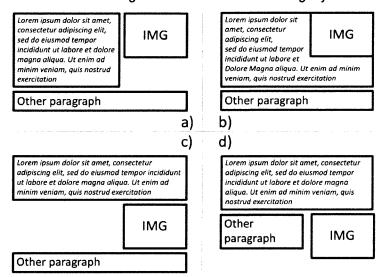1. For the following fragment of HTML

    ```
    <div>
      <p id="first">Short paragraph of text...</p>
      <img src="myFigure.png" alt="" width="100" height="200"/>
      <p id="second">More text..</p>
    </div>
    ```

    and CSS style definition

    ```
    img, li { float: right; }
    #second { clear: both; }
    #first  { font-style: italic;}
    ```

    Which of the following best describes the resulting layout?

    | |
    | Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation IMG |
    | Other paragraph |

    a)

    | Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et Dolore Magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation IMG |
    | Other paragraph |

    b)

    c)

    | Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation |
    | IMG |
    | Other paragraph |

    d)

    | Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation |
    | Other paragraph IMG |

2. The following CSS style definition
```
p i {
    font-style: italic;
}
```
makes the text italic in:
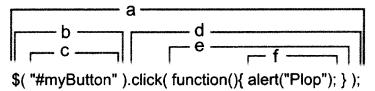   a. all p elements inside any i element
   b. all paragraph elements that include an i element
   c. all i elements that are anywhere *inside* any p element
   d. all p elements that have class i
   e. all p and i elements

HTTP

3. When some web page is browsed for the first time, we get a HTTP response code **200 OK** for most of the items the browser loads. The next time some items return a response code **304 Not Modified**. What is going on (1 point) and how does it work (1 point) ?
(max 2 sentences per point, **2 points**)

4. How does the server set cookies?
   a. Browser automatically requests for cookies from a standardized URL
   b. Cookies are included in the head element of html pages
   c. Cookies are given in the HTTP headers
   d. Cookies are given using the link tag and loaded as separate files
   e. None of the above

JavaScript & jQuery

5. Explain **briefly** what each element a-f **does** in the following piece of code. (1 point)
When are the different pieces of code executed? (1 point)
Don't explain the syntax but rather why the element is necessary

$("#myButton").click(function(){ alert("Plop"); });

Sessions & Security

6) Mallory is writing a blog post on a blogging service where she wrote the following as a title for the post. The **beginnings** of the texts she wrote are shown below. What is she trying to do in the two cases? When and where is the attack code executed? (**2 points**, 1 for each)
   a) `<script>` ..... *rest of the text not shown ....*
   b) `'); INSERT INTO` ..... *rest of the text not shown ....*

7) Briefly explain the basic idea of Cross-Site Request Forgery (CSRF) protection. (**max 5 sentences**)

Web Services

8) Why is api versioning a good idea? (~ two sentences, **1 point**)

9) You see a client sending the following requests to a REST API at example.com. Explain what you assume each request does. (one sentence per request, **2 points**)
   a. POST http://www.example.com/customers/12345
   b. GET http://www.example.com/customers/12345/orders
   c. POST http://www.example.com/customers/12345/orders

Django

10. (total 4 points)

The following snippets are a Django view and the corresponding urls.py that could have been in someone's course project (they weren't) The service hosts **private** photo albums for a number of users.

`views.py`

```
1    @login_required(login_url='/login/')
2    def del_page(request, album, page_num):
3        page = Page.objects.get_object_or_404(album=album,number=page_num)
4        page.delete()
5        return HttpResponseRedirect("/albums/")
```

`urls.py`

```
urlpatterns = [
    url(r'^delete/(?P<album>\w+)/(?P<page_num>[0-9]{2})$',
views.del_page),
    ...other urls...
```

(**3 points**) Briefly explain 3 fundamentally different scenarios that can what happen when a user visits the URL: http://someserver.com/delete/vacation/05
Use the line numbers if necessary to explain what happens.

(**1 point**) There is an authorization problem in the code. What is it? (Max 2 sentences.)