UNIVERSITY OF BRISTOL

Summer Examination Period

FACULTY OF ENGINEERING

2025 Examination for the Degree of
Bachelor and Master of Engineering and Bachelor and Master of Science

COMS10012 and COMSM0085
Software Tools

TIME ALLOWED:
1 Hour

Answers to COMS10012 and COMSM0085: Software Tools

Intended Learning Outcomes:

1. Please make sure you read the instructions on the answer sheet.
2. Each question is worth 1 mark.
3. Only the answer sheet will be marked, the empty pages at the back of the exam are only used for your calculations.
4. When selecting answers, make clear, horizontal marks within the two sets of brackets, making sure that the contained letter is struck through.
5. Avoid marking the answer sheet outside specified areas.
6. Do not crease, dog-ear or otherwise damage the answer sheet.

Q1. Georgia is writing code to handle the responses from an unfamiliar HTTP server. She's testing out various requests to see what the response looks like, and then implementing her client-side code to handle the responses.

Here are the headers for one request:

```
HEAD /1000bytes.txt HTTP/1.1
Host: www.requestplusonebyte.co.uk
Connection: close
```

And here are the headers from the corresponding response:

```
HTTP/1.1 200 OK
Content-Type: text/plain
Content-Length: 1009
```

How many bytes of message body will Georgia have to handle in her code for the response to this HEAD request?

A. 0

B. 1000

C. 1001

D. 1009

Solution: HEAD requests have no message body. The content-length reported (D) is reflecting what it would be if this request were a GET. We covered what HEAD does in the lectures and this exact point comes up in the assigned reading.

Q2. In a HTTP URI, which of the *host*, *scheme* and *path* components are case-insensitive?

A. None of them, all components are case-sensitive.

B. Only the *scheme* is case-insensitive.

C. The *scheme* and *host* are case-insensitive.

D. All of the *scheme*, *host* and *path* are case-insensitive.

Solution: Per the RFC assigned for reading, section 2.7.3. It is possible for a particular host to handle paths in a case-insensitive manner, but this does not affect the URI schema.

Q3. You encounter an unfamiliar HTTP status code 204. Without knowing more than this code, which of the below might you reasonably conclude?

A. The request has completed successfully.

B. The server needs to redirect you to another location.

C. The server has encountered an internal error.

(cont.)

D. There is an error in the request sent by the client.

Solution: We covered the meaning of HTTP status codes by first digit in the lecture.

Q4. Which of the below is NOT a HTML5 element?

A. select

B. thead

C. input

D. value

Solution: All the other options not only exist but have been seen in provided material. ('HTML elements' video and slides).

Q5. Why should you use `h1` for a page's top-level heading, instead of some other element?

A. It sets the correct font size and weight for a top-level heading.

B. You don't need to, this is just done by convention.

C. It allows programs to understand that the contained text is intended as a top-level heading.

D. You need at least one `h1` element on the page for the page to be valid HTML5, otherwise the browser will enter 'quirks' mode and this will affect page rendering.

Solution: A is false, the 'correct' values for this depend on your own page style and are best controlled through CSS. B is misleading – you indeed don't need to, but the question is about why you should, and this answer ignores why the convention exists. D is nonsense, you don't need a H1 on the page and it being absent would not activate quirks mode. This exact semantics question was a focus of one of the required reading MDN pages.

Q6. An *absolute* URL:

A. Points to a location using a protocol and domain name.

B. Is a permanent link that will never change.

C. Has the same value whether it is positive or negative.

D. Points to the top-level directory on a web server.

Solution: The definitions of absolute and relative URLs were in set reading. B and D would make for a severe misunderstanding, C does not even make sense for a URL.

Q7. Which of these CSS selectors will apply a style to a `div` element that is a descendant of a `span` with the ID 'jumble'?

A. `span.jumble div`

B. `span#jumble > div`

C. `span.jumble > div`

D. `span#jumble div`

Solution: CSS selectors and combinators were a big focus of the lecture and assigned reading, and a question very similar to this was provided in the example paper. To answer this correctly you just need to know the difference between the child and descendent combinators and the class and id selectors.

Q8. A rule with a selector of `form+button, .stray` would be applied to:

A. A `stray` element inside a `form` with class 'button'.

B. Any `button` element inside a `form`, and any element with the ID 'stray'.

C. Any `button` element immediately following a `form`, and any element with the class 'stray'.

D. Any element with the class 'stray' which is inside a `form` element followed by a `button`

Solution: This is another combinator question like those seen in past papers, the important parts being ',' for a list, '+' being a way to select an element immediately following another, and '.' being a class.

Q9. Which of these statements is true of a HTML box with a `display` of `inline`?

A. The `width` and `height` properties are respected.

B. The box will not break onto a new line.

C. Top and bottom padding will cause other elements to be pushed away from the box.

D. Left and right margins will not apply to other `inline` elements.

Solution: This question is the focus of the provided reading on the box model. A and C are definitions for the `block` mode, D is the opposite of the case for inline.

Q10. Jack is browsing using a 700px-wide device when he encounters a webpage with the following CSS stylesheet:

```
body {
 color: blue;
}
print {
 color: black;
}
@media screen and (max-width: 600px){
 h1, p {
  color: red;
 }
 h2 {
  color: yellow;
 }
 form {
  color: blue;
 }
}
@media print {
 body {
  color: yellow;
 }
}
@media screen and (min-width: 800px){
 p, a, div, h1 {
  color: black;
 }
 h2, h3 {
  color: blue;
 }
}
```

When Jack prints out the webpage, what colour will the contents of the `h1` element on the printed page be?

A. blue

B. red

C. yellow

D. black

Solution: This is a simple responsive design question, you just need to recognise that the print stylesheet applies when printing, regardless of the browser dimensions, and that properties in the H1 would be inherited from the body. The 'print' entry in the default stylesheet would apply to a 'print' element, if one existed.

Q11. In laying out a page, *margin collapse* can be best described as:

A. When two vertically adjacent elements have touching margins, the larger of the two margins remains and the smaller one disappears.

B. The lack of the expected margin effect that occurs when setting left and right margins on `inline` elements.

C. A layout pattern in which element margin values can dynamically shrink in response to the viewport width and height.

D. A failure mode that occurs when dynamically-adjusting margin values leads to margins unintentionally becoming negative.

Solution: Margin collapse is defined and discussed within the assigned reading on normal flow from the second week of CSS content. None of the other options are real.

Q12. When multiple items are added to a grid-based layout, they are automatically placed:

A. In the topmost and leftmost free space available on the grid that is large enough to contain the item.

B. In the earliest large enough free position on the grid according to the document's writing mode.

C. In the first column of the first row on the grid.

D. According to the item's preference rank property, but adjusted to account for whether the item can fit into the current column.

Solution: This question is covered in the assigned grid-based layout reading, and would also be apparent from the grid lab exercise. A is incomplete, this is only the case for English writing order. C would mean all items are in the same location. D is gobbledegook.

Q13. David has written a callback handler for a form submission event, in which he is attempting to make use of value typed into a text input field with the ID 'userinput'.

```
function plusE(){
       const e = 0xE;
       let uinputval = document.getElementById('userinput').value;
       console.log("Now it's "+ (e + uinputval));
}
```

David types '5' into the input field and submits the form. Assuming the callback is executed, what will be logged in the console?

A. "Now it's 0x13"

B. "Now it's 19"

C. "Now it's 20"

D. "Now it's 145"

Solution: The user input value is a string, and adding 14 to it this way produces a string "145". This issue was highlighted in the first JS lecture.

Q14. Which of the following is NOT a keyword in JavaScript?

A. this

B. of

C. not

D. let

Solution: All three of the actual keywords are present in several places in the taught material. 'let' is even present in examples in this exam. 'not' is only expressed as !

Q15. Consider the following JavaScript code.

```
let curval = 20;
if (curval > 20){
       console.log("a")
}
let curval = curval+1;
if (curval > 20){
       console.log("b")
}
```

When this code is executed, which items would appear in the console?

A. Both 'a' and 'b'.

B. Only 'a'.

C. Only 'b'.

D. Neither 'a' nor 'b'.

Solution: The 'a' value would not be printed because 20 is not greater than 20. The line that looks like it is incrementing curval to 21 would actually produce an error, because it's an attempt to redeclare the same value, so curval is still 20 at the second check, and 'b' would not be logged. This behaviour of 'let' was in the lecture and the assigned reading on variables.

Q16. Gordon uses `fetch` to retrieve a web page in an asynchronous request, but he has made a typographical mistake in the *path* component of the URL. After he gets a 404 response, the state of the fetch `Promise` is:

(cont.)

A. 'pending'

B. 'fulfilled'

C. 'rejected'

D. undefined

Solution: As the question says, a response has been received, so the Promise is fulfilled – `fetch` has delivered what it promised. The fact that the response itself indicates an error state is immaterial. The state would only be 'rejected' if there was a network error preventing a response from being obtained (which could happen for a typo in the protocol or domain).

Q17. A callback is attached to multiple JavaScript Promises like so:

```
Promise.all([promise1,promise2,promise3]).then(commonHandler)
```

`commonHandler` will be invoked:

A. Up to three times, when any of the three promises are fulfilled.

B. At most once, when every promise has been fulfilled.

C. Exactly three times, every time one of the promises is resolved, whether they are fulfilled or rejected.

D. Exactly once, after every promise has been resolved, whether they are fulfilled or rejected.

Solution: This invocation and its behaviour was in the assigned reading for the second week of Javascript. 'All' means all promises are fulfilled.

Q18. Bob needs to protect his web application's database from overlength passwords. It's most important for him to use:

A. A polite notice to users about the password length restriction.

B. The `maxlength` property in HTML form validation.

C. JavaScript form validation, using an event listener to check the length of the input field's value upon the form submission event.

D. Server-side input validation.

Solution: This question was prefigured by a question in a past exam paper, and was also the subject of a set reading in the second week of Javascript.

Q19. When using `wget` with the `-nc` option, what will happen when attempting to download a file twice into the same directory?

A. The first version of the file will be overwritten with the second version.

B. The first version of the file will be preserved and the second version will be given an extension '.1'.

C. The first version of the file will be preserved and `wget` will refuse to download the second version of the file.

D. The behaviour will depend on the local and remote timestamps for the two versions. If the second version is newer than the first version, then the first version will be overwritten.

Solution: Reading about the '-nc' option was an assigned exercise for the scraping lab. These are all behaviours that wget can engage in under different conditions. A would be clobbering, B is default behaviour, D is behaviour under '-N'.

Q20. Guidelines to follow when web-scraping DO NOT include:

A. Check for an alternative API endpoint first.

B. Respect the `robots.txt` file.

C. Be aware of limitations that apply to republishing even public web content.

D. Remember that while you can access web pages freely, downloading copies is copyright infringement.

Solution: D is legalistic nonsense, there is no technical difference between 'accessing' and 'downloading a copy' of a web page. The guidelines for web scraping were covered in the lecture and reiterated in the lab exercises.

Q21. Consider the HTML document below:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset='utf-8' />
    <link rel='stylesheet' href='style.css'/>
    <title>A Title</title>
  </head>
  <body>
    <header>
      <h1>A Title</h1>
    </header>
    <main>
      <p class='info'>
          This page links to some other pages.
      </p>
      <div class='container'>
     <div id="l1">
       <a href='./page1.html'>Page 1</a>
     </div>
     <div id="l2">
       <a href='./page2.html'>Page 2</a>
     </div>
     <div id="l3">
       <a href='./page3.html'>Page 3</a>
     </div>
     <div id="l4">
       <a href='./page4.html'>Page 4</a>
     </div>
     <div id="l5">
       <a href='./page5.html'>Page 5</a>
     </div>
      </div>
    </main>
    <footer>
     -- by A. Page Author.
    </footer>
  </body>
</html>
```

Assume a BeautifulSoup object named 'soup' has been created to represent this document. Which of the following lines of Python code would resolve to the string "Page 1"?

A. `soup.find_all('a')[0]`

B. `soup.find('l1').a.text`

C. `soup.main.div.div.a.text`

D. `soup.find('footer').parent.find('div').find_all('a')[1-1].contents`

Solution: This relies mostly on understanding how find calls can be used. A finds the anchor element that contains the required string, B will error out (l1 does not resolve to an element), D is closer than it might seem but retrieves a (single-member) array of contents.

Q22. The protocol HTTP operates at the 7th, *Application layer* of the OSI model. At which layer does the TCP protocol operate?

A. 6, Presentation layer

B. 5, Session layer

C. 4, Transport layer

D. 2, Data Link layer

Solution: The OSI model, including TCP's place in it, was covered in the first lecture of this TB.

Q23. Which statement best describes Public Key Cryptography?

A. One key is created which is encrypted with a password.

B. Two keys are created, and both are kept secret.

C. Two keys are created, one that's is public, one that is kept secret.

D. One key is created that is publicly shared.

Q24. Consider the following certificate chain. Who issued it and when does the certificate expire?

```
Certificate chain
 0 s:CN=kingdomofloathing.com
   i:C=US, O=Amazon, CN=Amazon RSA 2048 M02
   a:PKEY: rsaEncryption, 2048 (bit); sigalg: RSA-SHA256
   v:NotBefore: Oct 14 00:00:00 2024 GMT; NotAfter: Nov 12 23:59:59 2025 GMT
 1 s:C=US, O=Amazon, CN=Amazon RSA 2048 M02
   i:C=US, O=Amazon, CN=Amazon Root CA 1
   a:PKEY: rsaEncryption, 2048 (bit); sigalg: RSA-SHA256
   v:NotBefore: Aug 23 22:25:30 2022 GMT; NotAfter: Aug 23 22:25:30 2030 GMT
 2 s:C=US, O=Amazon, CN=Amazon Root CA 1
   i:C=US, ST=Arizona, L=Scottsdale, O=Starfield Technologies, Inc., CN=Starfield
       Services Root Certificate Authority - G2
   a:PKEY: rsaEncryption, 2048 (bit); sigalg: RSA-SHA256
   v:NotBefore: May 25 12:00:00 2015 GMT; NotAfter: Dec 31 01:00:00 2037 GMT
```

A. kingdomofloathing.com, November 13 0:00:00 2025 GMT

B. Amazon, August 23 22:25:31 2030 GMT

C. Amazon, November 13: 00:00:00 2025 GMT

D. kingdomofloathing.com, October 14: 00:00:00 2024 GMT

Q25. Why might you want to sign and encrypt an email?

A. To ensure that it will not be flagged as spam.

B. To ensure that the message is delivered to the correct recipient.

C. To ensure that the message cannot be read by others, and to ensure that a strong encryption algorithm was used.

D. To ensure that the message cannot be read by others, and so that the recipient can verify that you sent it.

‘

Q26. What does RFC stand for in relation to internet standards?

A. Request For Comments

B. Request For Changes

C. Recent Future Code

D. Read For Code

‘

Q27. What is a Berkeley Socket?

A. A degree program at a prestigious university.

B. A mechanism for resolving a domain name into an IP address.

C. A file descriptor that can be `read` and `writen` to.

D. The endpoint of a connection over a network.

‘

Q28. What port does finger run on?

A. 80

B. 79

C. 70

D. 22

‘

Q29. What is fuzz testing?

A. A series of unit tests that verify the code performs as expected.

B. A mechanism for finding bugs by testing with random inputs until a crash is detected.

C. A technique for testing surface rendering in computer graphics.

D. A formal methods approach for proving code is correct.

‘

Q30. What is the advantage of property-based test frameworks like *QuickCheck* over traditional unit testing?

A. They can be written without knowing how to write code.

B. They are faster and more efficient at finding bugs.

C. They can generate a large number of test cases without having to specify each individually.

D. They prove definitively that the code is correct for all inputs..

Q31. Brian wants to test their sorting function works. Which of the following properties might make good candidates to test with QuickCheck?

1. Sorting a sorted list gives the same list back.
2. A sorted list has the same number of elements as an unsorted list.
3. A sorted list is only equal to a reversed sorted list if it contains one or less unique elements.
4. Sorting an empty list gives you an empty list

A. 4 only.

B. All of them.

C. None of them.

D. 1, 2, and 3.

‘

Q32. Who can't encrypt their email?

A. Alice

B. Johnny

C. Bob

D. François

‘

(cont.)

—Blank page—

Qu. continues ...

—Blank page—

(cont.)

—Blank page—

Qu. continues ...

—Blank page—

(cont.)

—Blank page—

END OF PAPER