# WAD Lab Test 1 (1.5 hours) [20 marks]

**General Instructions:**

- You can refer to any offline resources already on your laptop, but you must disable all networking and Bluetooth connections during the test. You must not communicate with anyone via any means during the test.
- Just before the test, you will be given instructions by the invigilator as to how to obtain resource files required for the lab test and how to submit your solutions.
- **No questions** will be entertained during the test. If necessary, **make your own assumptions** and **document them as comments** in the submitted HTML or PHP files**.**
- You are allowed to use only standard PHP classes and functions in your solutions – do not use any third-party libraries.
- Use meaningful names for classes, methods, functions and variables, as well as indent your code correctly. Use 4 spaces for indentation. Otherwise, you may attract penalty of up to **20%** of your score for the corresponding question.
- You **MUST** include your name as author in the comments of all your submitted source files. Failure to do so WILL attract a penalty of up to **20%** of your score for the corresponding question.

  For example, if your registered name is "TAN Haoye" and email ID is tan.haoye.2020, include the following comment at the beginning of each source file you write.

```
<!--
    Name:  TAN Haoye
    Email: tan.haoye.2020
-->
```

- You may wish to comment out the parts in your code which cause errors. But commented code will not be marked.

## DO NOT TURN OVER UNTIL YOU ARE TOLD TO DO SO

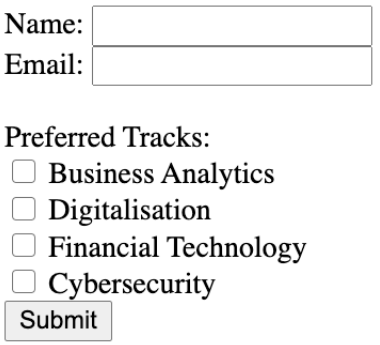
**Question 1 (Difficulty Level: */*)** **[6 marks]**

**Given:**

- q1/
  - q1.html (Modify this file)
  - q1.php (Modify this file)

**Part A [2 marks] – Difficulty Level (*)**

Modify **q1.html** such that it displays a form as shown below:




**Instructions:**

1. The form contains "Name" and "Email" input fields. It also contains checkboxes for selecting "Preferred Tracks" options.

2. Upon clicking the "Submit" button, name, email and the values corresponding to the selected checkboxes are sent to **q1.php** via **HTTP GET**
   a. You must use **relative path**

**Note**: You are free to decide any names for your form components.

**Part B [4 marks] – Difficulty Level (*)**

Modify **`q1.php`** such that it performs form processing as shown below:

<table>
<tr><th><code>q1.html</code></th><th><code>q1.php</code></th></tr>
<tr><td colspan="2">When user enters name and email and click “Submit” button without selecting any option:</td></tr>
<tr><td>Name: SharK<br>Email: shark@smu.edu.sg<br><br>Preferred Tracks:<br>☐ Business Analytics<br>☐ Digitalisation<br>☐ Financial Technology<br>☐ Cybersecurity<br>Submit</td><td>No track selected</td></tr>
<tr><td colspan="2">When user enters name and email, selects “Digitalisation” and “Cybersecurity” options, and click “Submit” button:</td></tr>
<tr><td>Name: SharK<br>Email: shark@smu.edu.sg<br><br>Preferred Tracks:<br>☐ Business Analytics<br>☑ Digitalisation<br>☐ Financial Technology<br>☑ Cybersecurity<br>Submit</td><td><table><tr><th>Name</th><th>Email</th><th>Preferred Tracks</th></tr><tr><td>SharK</td><td>shark@smu.edu.sg</td><td>• Digitalisation<br>• Cybersecurity</td></tr></table></td></tr>
<tr><td colspan="2">When user enters name and email, selects “Business Analytics”, “Digitalisation”, and “Financial Technology” options, and clicks “Submit” button:</td></tr>
<tr><td>Name: SharK<br>Email: shark@smu.edu.sg<br><br>Preferred Tracks:<br>☑ Business Analytics<br>☑ Digitalisation<br>☑ Financial Technology<br>☐ Cybersecurity<br>Submit</td><td><table><tr><th>Name</th><th>Email</th><th>Preferred Tracks</th></tr><tr><td>SharK</td><td>shark@smu.edu.sg</td><td>• Business Analytics<br>• Digitalisation<br>• Financial Technology</td></tr></table></td></tr>
</table>

**Instructions:**

1. `q1.php` validates that at least one checkbox in `q1.html` is selected. Otherwise, it prints the message “No track selected”. You may assume that the user always enters “Name” and “Email”.

2. Use “border=1” for the table. Use the element <th> for table headers.

**Question 2: (Difficulty Level: */** ) [8 marks]**

**Given:**

- q2/
  - images/* (There are 2 JPG image files)
  - q2.html (Modify this file)
  - q2_process.php (Modify this file)

**Part A [2 marks] – Difficulty Level (*)**

Complete "**Options"** table in **`q2.html`** such that it displays:


**Instructions:**

1. The table contains:
   - Checkboxes for upgrade options 1, 2, and 3.
     - The **values** of the three upgrade options are 0, 1, and 2 respectively.
   - A drop down list for Warranty with options: "2 years, free" or "3 years, $168.00".
     - The **values** for the two warranty options are 2 and 3 respectively.
   - A submit button "Calculate"
2. Upon clicking the submit button "Calculate", it submits to **`q2_process.php`** via HTTP **POST**.

**Note**: You are free to decide any names for your form components.

**Part B [6 marks] – Difficulty Level (*/**)**

Complete **`q2_process.php`** such that it performs form processing as shown below:

<table>
<tr><th><code>q2.html</code></th><th><code>q2_process.php</code></th></tr>
<tr><td colspan="2">When user selects upgrade options 1 and 2 and clicks “Calculate” button:</td></tr>
<tr><td>

ThinkPad X1 Carbon Gen 8
<table>
<tr><td>Memory</td><td>8GB RAM</td></tr>
<tr><td>Storage</td><td>256GB SSD</td></tr>
<tr><td>Privacy Filter</td><td>Standard</td></tr>
<tr><td colspan="2">S$ 1,669.00 (w/GST S$ 1,785.83 )<br>with 2 year international & local warranty</td></tr>
<tr><td colspan="2">Options</td></tr>
</table>
<table>
<tr><th>No.</th><th>Description</th><th>Price w/o GST</th></tr>
<tr><td>1 ☑</td><td>Upgrade to 16GB RAM</td><td>$323.00</td></tr>
<tr><td>2 ☑</td><td>Upgrade to 16GB RAM + 512GB SSD</td><td>$433.00</td></tr>
<tr><td>3 ☐</td><td>Upgrade to Premium Privacy Filter</td><td>$48.00</td></tr>
<tr><td>4</td><td colspan="2">Warranty: 2 years, free</td></tr>
<tr><td colspan="3">Calculate</td></tr>
</table>
</td><td>You can only select one upgrade option from 1-2</td></tr>
<tr><td colspan="2">When user directly clicks “Calculate” button:</td></tr>
<tr><td>

ThinkPad X1 Carbon Gen 8
<table>
<tr><td>Memory</td><td>8GB RAM</td></tr>
<tr><td>Storage</td><td>256GB SSD</td></tr>
<tr><td>Privacy Filter</td><td>Standard</td></tr>
<tr><td colspan="2">S$ 1,669.00 (w/GST S$ 1,785.83 )<br>with 2 year international & local warranty</td></tr>
<tr><td colspan="2">Options</td></tr>
</table>
<table>
<tr><th>No.</th><th>Description</th><th>Price w/o GST</th></tr>
<tr><td>1 ☐</td><td>Upgrade to 16GB RAM</td><td>$323.00</td></tr>
<tr><td>2 ☐</td><td>Upgrade to 16GB RAM + 512GB SSD</td><td>$433.00</td></tr>
<tr><td>3 ☐</td><td>Upgrade to Premium Privacy Filter</td><td>$48.00</td></tr>
<tr><td>4</td><td colspan="2">Warranty: 2 years, free</td></tr>
<tr><td colspan="3">Calculate</td></tr>
</table>
</td><td>
<h1>Please confirm your selection</h1>
<table>
<tr><th>No.</th><th>Description</th><th>Price w/o GST</th></tr>
<tr><td>1</td><td>ThinkPad X1 Carbon Gen 8</td><td>$1,669.00</td></tr>
<tr><td>2</td><td>2 years warranty</td><td>$0.00</td></tr>
<tr><td colspan="2">Subtotal</td><td>$1,669.00</td></tr>
<tr><td colspan="2">GST</td><td>$116.83</td></tr>
<tr><td colspan="2">Total</td><td>$1,785.83</td></tr>
</table>
</td></tr>
</table>

When user enters valid options and clicks “Calculate” button:

ThinkPad X1 Carbon Gen 8

| | |
|---|---|
| **Memory** | 8GB RAM |
| **Storage** | 256GB SSD |
| **Privacy Filter** | Standard |

**S$ 1,669.00 (w/GST S$ 1,785.83 )**
with 2 year international & local warranty

**Options**

| No. | Description | Price w/o GST |
|---|---|---|
| 1 ☑ | Upgrade to 16GB RAM | $323.00 |
| 2 ☐ | Upgrade to 16GB RAM + 512GB SSD | $433.00 |
| 3 ☑ | Upgrade to Premium Privacy Filter | $48.00 |
| 4 | Warranty: 3 years, $168.00 | |

Calculate

# Please confirm your selection

| No. | Description | Price w/o GST |
|---|---|---|
| 1 | ThinkPad X1 Carbon Gen 8 | $1,669.00 |
| 2 | Upgrade to 16GB RAM | $323.00 |
| 3 | Upgrade to Premium Privacy Filter | $48.00 |
| 4 | 3 years warranty | $168.00 |
| Subtotal | | $2,208.00 |
| GST | | $154.56 |
| Total | | $2,362.56 |

**Instructions:**

1. If a user selects both upgrade options 1 and 2, display: “You can only select one upgrade option from 1-2”.

2. **Otherwise**, print the selected options and their corresponding prices into a table by making use of `$upgrade_info` and `$warranty_info` arrays provided in **`q2_process.php`**.

   The table must also contain subtotal, GST, and total prices. You need to:

   i. Calculate the subtotal of the base price (1669.00) and selected options.
   ii. Calculate the GST amount based on the subtotal. GST is 7% of the subtotal.
   iii. Calculate the total amount.
   iv. Use `number_format()` to format the price to display up to two decimal digits

**Note**:

- Use <h1> for “Please confirm your selection”.
- You can hardcode the description of the laptop (“ThinkPad X1 Carbon Gen 8”) and its base price (“$1,669.00”), but **not** the others (e.g., GST price, etc.).
- If **`q2_process.php`** is visited without going through **`q2.html`**, redirect to **`q2.html`**. Use `header()`

**Question 3 (Difficulty Level **/***)** **[6 marks]**

**Given:**

- `q3/`
  - `images/g1/* (There are 3 JPG image files)`
  - `images/g2/* (There are 3 JPG image files)`
  - `images/g3/* (There is 1 JPG image file)`
  - `sample_test_q3.php`
  - `q3.php` (Modify this file)

1. `q3.php` shows man and woman pairs of images from different groups.
2. In `q3.php`, you are given `$groups`, which is an associative array where:

| Key | Value |
|---|---|
| Group ID<br>( e.g. `"g1"` ) | Indexed array containing a set of person IDs<br>( e.g. `['w1', 'm1', 'm2']` )<br><br>where<br>• a person could be a man or a woman.<br>• a woman's ID always starts with the prefix **w**<br>• a man's ID name always starts with the prefix **m** |

3. In `q3.php`, you are also given `$ratings`, which is an associative array where:

| Key | Value |
|---|---|
| ID of a person<br>( e.g. `"w1"` ) | The rating of the person<br>( e.g. `9` ) |

**Note**

- You may create additional helper functions (i.e. your own additional functions that perform a part of the computation) in `q3.php` as you deem fit.
- Use `$groups` and `$ratings` given in your `q3.php` to test your code against the examples given.
- We will be using a script to automatically test your `q3.php` file. We will use `$groups` and `$ratings` containing data different from the ones given in `q3.php`
- You may assume that the test data we use will have a similar data structure as the ones given in `q3.php`.
- To test your code, you may also use the sample test script `sample_test_q3.php`, which contains five test cases each for Part A and Part B. If you run `sample_test_q3.php` in the browser and it will report whether each of the test cases passes or fails.

**Part A [2 marks] – Difficulty Level (**)**

1. You are to implement the function **`generatePairs($groups)`**.

2. The function returns `$pairs`, which is a multi-dimensional array, an (indexed) array of indexed arrays. It represents **all possible pairs of a man and a woman from different groups, according to the data defined in `$groups`**.

For example, given,

```
$groups = [
            "g1" => ['w1', 'm1', 'm2'],
            "g2" => ['m3'],
            "g3" => ['w4']
          ]
```

The function returns `$pairs` as:

```
[
    ["m3", "w1"],
    ["m1", "w4"],
    ["m2", "w4"],
    ["m3", "w4"]
]
```

Note:

- You may assume that `$groups` data passed to the function is valid. That is, error checking of `$groups` variable is NOT required.
- You may assume that **each person belongs to only one group**.

3. **Each element (indexed array),** e.g. `["m3", "w1"]`, of `$pairs` must satisfy the following conditions:
   a. It contains exactly two elements of String data type; the first element is the ID of a man and the second element is the ID of a woman. Therefore, it represents **a pair of a man and a woman**.

      Note: The first element always represents the man. The second element always represents the woman.

   b. The man and the woman in each pair belong to **different groups defined in `$groups`** (`"g1"`, `"g2"`, etc.). For example, `"m3"` belongs to `"g2"` and `"w1"` belongs to `"g1"`. They cannot be from the same group.

4. The elements of `$pairs` can be **in any order**. For example, both `[['m1', 'w2'], ['m1', 'w3']]` and `[['m1', 'w3'], ['m1', 'w2']]` are accepted.

5. **Duplicate elements** in `$pairs` are not allowed. For example, `[['m1', 'w2'], ['m1', 'w2']]` is incorrect.

See the examples below.

| Given `$groups` | The function returns `$pairs` as: |
|---|---|
| `[`<br>`  "g1" => ['w1', 'm1'],`<br>`  "g2" => ['w2', 'w3']`<br>`]` | `[`<br>`  ['m1', 'w2'], ['m1', 'w3']`<br>`]` |
| `[`<br>`  "g1" => ['w1', 'm1', 'm2'],`<br>`  "g2" => ['w2', 'w3', 'm3']`<br>`]` | `[`<br>`  ['m3', 'w1'], ['m1', 'w2'], ['m1', 'w3'],`<br>`  ['m2', 'w2'], ['m2', 'w3']`<br>`]` |
| `[`<br>`  "g1" => ['m2'],`<br>`  "g2" => ['m3'],`<br>`  "g3" => ['w4']`<br>`]` | `[`<br>`  ['m2', 'w4'], ['m3', 'w4']`<br>`]` |

| | |
|---|---|
| `[`<br>`  "g1" => ['w1', 'm1', 'm2'],`<br>`  "g2" => ['m3'],`<br>`  "g3" => ['w4']`<br>`]` | `[`<br>`  ['m3', 'w1'], ['m1', 'w4'], ['m2', 'w4'],`<br>`  ['m3', 'w4']`<br>`]` |
| `[`<br>`  "g1" => ['w1', 'm1', 'm2'],`<br>`  "g2" => ['w2', 'w3', 'm3'],`<br>`  "g3" => ['w4']`<br>`]` | `[`<br>`  ['m3', 'w1'], ['m1', 'w2'], ['m1', 'w3'],`<br>`  ['m2', 'w2'], ['m2', 'w3'], ['m1', 'w4'],`<br>`  ['m2', 'w4'], ['m3', 'w4']`<br>`]` |

Hint:

- use `array_keys()` to obtain the keys from the associative array
- use `array_merge()` to merge arrays
- You may run `sample_test_q3.php` to test the correctness of your implementation

**Part B [4 marks] – Difficulty Level (***)**

1. You are to implement the function **`sortPairs_addGroup($pairs, $groups, $ratings)`**. `$pairs` is the same data returned by the function `generatePairs`.

2. You may assume that `$pairs`, `$groups`, `$ratings` data passed to the function is valid. Error checking of those variables is NOT required.

3. Firstly, the function sorts the pairs in **descending order**, according to the combined rating of man and woman.

   For example, given

```
$pairs = [ ['m1', 'w2'], ['m1', 'w3'] ]


$ratings = [
                'w2' => 6.5,
                'w3' => 7.5,
                'm1' => 9.5
           ]
```

   The combined rating of the first pair `['m1', 'w2']` is 16.
   The combined rating of the second pair `['m1', 'w3']` is 17.
   Hence, the sorted result is

```
$sortedPairs = [ ['m1', 'w3'], ['m1', 'w2'] ]
```

   Note:
   - You may assume that no pair has the same combined rating value.

4. Next, for each person ID in the sorted pairs, the function **prefixes with the corresponding group ID** in the format of `'groupID/personID'`

   For example, given

```
$groups = [
                "g1" => ['w1', 'm1'],
                "g2" => ['w2', 'w3']
          ]


$sortedPairs = [ ['m1', 'w3'], ['m1', 'w2'] ]
```

   The group IDs are attached as follows:

```
$sortedPairs_addedGroup = [ ['g1/m1', 'g2/w3'], ['g1/m1', 'g2/w2'] ]
```

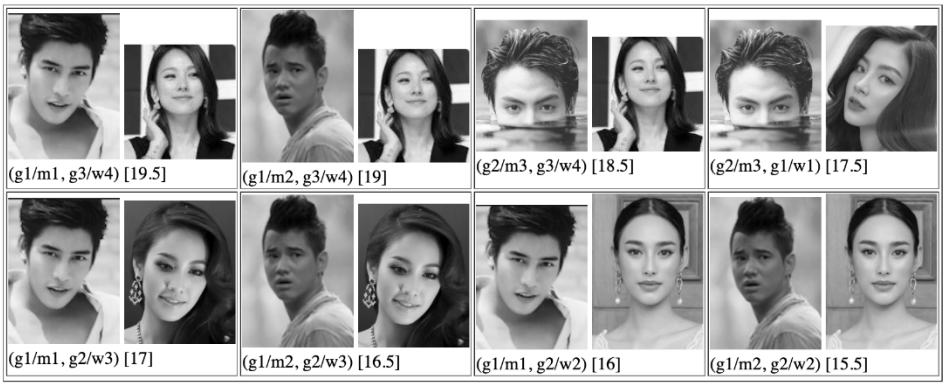
5. The function returns `$sortedPairs_addedGroup`

Hint:

- use `arsort()` to sort the associative array in descending order, according to the value.
- alternatively, use `krsort()` to sort the associative array in descending order, according to the key.
- use `in_array()` to search for a value in the given array.
- You may run `sample_test_q3.php` to test the correctness of your implementation

If the functions `generatePairs` and `sortPairs_addGroup` are implemented correctly, your `q3.php` should display the following in the browser:

**Sorted Celebrity Pairs According to Their Combined Ratings**




**- END -**