ASSESSMENT RESULT SHEET

Centre: **Computing** Course: **Foundation Degree in Engineering in Software Engineering**

Assessment Set By: Alan Stewart Date: 3rdSeptember2021

Assessment Validated By: Stephanie Allen Date: 27thSeptember 2021

|  |  |  |  |
| --- | --- | --- | --- |
| Name: | | Percentage Awarded | Weighting Within Module:  60% |
| Module Title:  Programming 1 |  | Date Issued:  10thMarch 2022 | Dates Due:  31stMarch 2022 |
|
| Assessment No: P1A2 |

|  |  |  |  |
| --- | --- | --- | --- |
| Outcomes – No. & Title |  | | |
| A1 Be able to understand problem specifications, plan, and implement appropriate solutions.  A2 Be able to understand the importance of algorithms in designing solutions to a range of problems.  B1 Design and implement software solutions using a standard methodology.  B2 Design and implement testing measures to ensure the functionality of a software solution.  C1 Document and test solutions in a high-level language based on a problem specification.  C2 Implement programming solutions based on an algorithmic solution. | | | |
|  | | Percentage Awarded | Percentage Available |
| **Main area for assessment:** | |  |  |
|  | |  |  |
| 1. Use of appropriate data types, variables and expressions | |  | 10% |
| 2. construction of a menu-based system | |  | 10% |
| 3.option1: Use of input and output with validation  4. option 2: print calendar  5. option 3: print date information  6.Document and test program using valid and invalid data  7. Use of procedures | |  | 20%  20%  10%  20%  10% |
|  | |  |  |
| Total | |  | 100% |

Signature of Assessor\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date

Cross Assessor Signature\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

IM Signature\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Date\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |
| --- |
| ASSESSMENT FEEDBACK |

# Assessment 2016 - 2017

|  |
| --- |
| Minimum StandardsPresentation skills are considered essential for students. If your assignment document does not conform to the following minimum standards the marks available will automatically be **reduced by 25%.**   * All assignments must have the appropriate front sheets attached. If these are not attached the assignment **will not be accepted**. * Documents must be word processed * Documents must have a Contents page, a Bibliography and a Reference section * Headers and footers to include your name, the module, the assignment title and a page number  Originality All work submitted must be original. Any material duplicated, either in published articles or other students’ assignments will automatically be disqualified. When two or more assignments are found to have similar portions, all parties carry the same penalty. |

## BMC Calendar Application

In this assignment you are required to design, implement and test a Pythonprogram which manages and displays formatted calendar information.

The application manages the following date information:

Day : an integer in the range 1 to 31 (depending on month)

Month : an integer in the range 1 to 12

Year : an integer >= 2000

Day of Week : an integer in the range 1 to 7 (with Sunday having value 1)

Through a menu system, the application will allow the user to specify the current date, display the date via formatted output and display the calendar for the current month.

When the system starts the customer will be presented with the following:

**BMC Calendar Application**

**Please choose from the following options:**

**1. Set Date**

**2. Display calendar for this month**

**3. Display date information**

**4. Quit**

**Option:**

Should the user select option 2 or 3 before the date has been set (via option 1) the following output should appear:

**BMC Calendar Application**

**Please choose from the following options:**

**1. Set Date**

**2. Display calendar for this month**

**3. Display date information**

**4. Quit**

**Option:3**

**Please set the date first**

**Menu details**

**Option Details**

**1** When this option is chosen,the user is asked to provide each piece of information in the following order: year, month, day and day of week, as follows (user input underlined:

|  |
| --- |
| **BMC Year Planner**  **Please choose from the following options:**  **1. Set Date**  **2. Display calendar for this month**  **3. Display date information**  **4. Quit**  **Option:1**  **Enter Year after 1999:2016**  **Enter Month (1 to 12):9**  **Enter Day:26**  **Enter Day of week (1-7):2** |

If any of the date components are not valid, a suitable error message should be given. The following shows the sequence for entry of a valid year but not a valid month:

|  |
| --- |
| **Option:1**  **Enter Year after 1999:2016**  **Enter Month (1 to 12):14**  **Problem with month entry:14**  **BMC Year Planner**  **Please choose from the following options:**  **1. Set Date**  **2. Display calendar for this month**  **3. Display date information**  **4. Quit**  **Option:** |

**2** When the user selects option 2, the calendar for the current month (depending on date entered) should be displayed. For example, if the date is defined as 26/9/2016, day of week 2 (for Monday), the following output should appear:

|  |
| --- |
| **Option:2**  **Calendar for September 2016**  **Su Mo Tu We Th Fr Sa**  **1 2 3**  **4 5 6 7 8 9 10**  **11 12 13 14 15 16 17**  **18 19 20 21 22 23 24**  **25 \* 27 28 29 30** |

The current month and year should be displayed on a single line, followed by the calendar information. The calendar should be organised with headings for each week day, starting with Sunday, with subsequent lines showing the list of days appropriate for the current month. If a month starts part-way through a week, previous days should be prefilled with a space. Similarly, if thecurrent month ends part-way through a week, remaining days of that week should be filled with a space. The current day should be displayed with an asterisk.

**3** When this option is chosen, the current date (if defined) should be displayed. If the current date is defined as 26/9/2016, day of week 2 (for Monday), the following output should appear:

|  |
| --- |
| **Option:3**  **\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***  **Monday, 26 September 2016**  **2016 is a leap year.**  **\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*** |

The output should indicate whether or not the current year is a leap year (as in the example above).

**4** When this option is chosen, the program should display the following message before exiting the application.

|  |
| --- |
| Thank you for visiting the BMC Calendar application |

**Program supplementary notes**

* Appropriate validation of input data should be applied throughout.
* Options 1,2& 3when completed should bring the user back to the main menu.
* Option 4 should quit the program with the confirmation message above.
* The output generated by your program should be exactly as described above.
* Your program ***must not*** make use of any ***date-based library classes***.

**Submission**

Submission is via CANVAS. You must attach the following files in your submission.

* **A test table with associated test runs (linked to the test plan). Ensure that one test uses the sample data from the problem specification.**
* **A copy of a fully documented program listing (ideally as a notepad file).**