University of Central Lancashire
UCLan

CO1404 : Introduction to Programming

2020-2021 Coursework : Vending Machine

1. About

Assessment Method : **Coursework**

- Released on : 30/11/2020
- Submission Deadline : **January 11th 2021 by 23:59:59**
    - Please note that this is the final time you can submit – not the time to submit!
    - Your grade and feedback for this assessment will be made available to you on or before : **01/02/2021**
- This is an Individual coursework project, and no group work is permitted.
- You must implement your solution in Visual Studio using the C# programming language.
- Your solution must also be accompanied by a report.

2. Learning Outcomes

This assessment has been designed to assess the following learning outcomes:

1. Develop a structured solution to a simple problem.
2. Explain the importance of code readability and maintainability.
3. Check the robustness of the code using an appropriate test strategy.

3. Preparation

Before attempting this assessment, it is highly recommended that you revisit the "**Four L's**":

- **Lectures** – This includes the slides, notes and recording.
- **Lecture notes** – Any notes you took during the lectures.
- **Lab worksheets** – Read over all lab worksheets.
- **Lab projects** – Ensure all projects have at least stage one implemented.

Combined these provide all the necessary information for you to successfully complete this assessment. All resources are available on the CO1404 Blackboard area under *Module Materials*.

4. Assessment

This assessment carries a weighting of 100%, meaning the grade awarded is the overall grade you receive for this the module. You will be assessed via a piece of coursework consisting of two components:

- The Implementation aspect will constitute up-to **60%** of your mark.
- The Report aspect will constitute up-to **40%** of your marks.

4.1. Implementation

You have been tasked with developing a new vending machine application for UCLAN Vending Machines LTD latest state-of-the-art vending machine. You are required to implement a proof-of-concept console-based vending machine application to demonstrate the vending machines functionality to investors. The vending machine will dispense the products listed in the table located in the table below:

Product	Cost (credits)
Chocolate Bar	0.80
Soda Can	0.70
Soda Bottle	1.25
Crisps	0.50
Cookies	1.10

4.1.1. Features

As the developer, you have total control regarding the menu, interactions, flow of the application and messages displayed the user. Use the examples included in **Appendix A: Example Outputs** for inspiration. However, you must attempt to implement the logic for following features defined in the table below.

Feature	Description
Add Credits	The application will require a user to input a number of credits into the vending machine application. The credits will represent the currency.
Order Selection	The user should be able to select a product or products from the machine, the application should calculate the total cost of the selected products.
Checkout	If there are sufficient credits available, the user should be able to then purchase the item/s selected deducting the amount from the available credit and return the remaining credits back to the user. If there are insufficient credits available, the user should not be able to purchase the items and will be prompted with a message requesting that additional credits are needed.

4.1.2. Expectations

You are expected to use and implement the programming concepts you have learned throughout this module to complete the implementation aspect of this coursework. You will also be assessed on the readability and appropriate use of programming concepts to implement features of the application defined above. This involves:

- Indenting your code blocks correctly.
- Choosing suitable data types.
- Choosing suitable names for variables
- Choose a suitable name for programming project.
- Appropriate use of programming concepts.
- Appropriate use of comments within your code that describe key or complicated aspects.

4.2 Report

To accompany the application, you will also produce a small report using the coursework template provided on Blackboard. The template will provide all the formatting requirements, such as tables. Using the template populate the following sections described in the table below.

	Description
Front Matter	Replace `Name` and `email` text on the front page with your own full name and UCLAN email address.
Questions & Answers	In your own words, answer the questions included in this section. Take note of the marks associated to each question as this indicates the level of detail required in your answer.
Documentation	In this section of the report, you should produce a user guide document all variables, methods and functions you have implemented into your solution. All documentation should be explicit, meaning descriptions and instructions are clear and unambiguous. You should document the following: **- Variables**: Document the attributes `type`, `name` and `description`. The description should provide details of how it is used in your application. **- Methods**: `method-name`, `description` , any `parameters` (including description) it has and `return-type` should be documented. **- User Guide** - The user guide should provide details on how to use your application. You should include screenshots and text to assist the end-user.
Testing	In this section you should include evidence (screen shots and description) of testing your application against the use-case defined in the coursework template.
Reflection	The final aspect of the report you should include a brief self-reflection. It should include details surrounding your programming accomplishments, some of the programming challenges you encountered as well as any improvements you would make to your application. You should conclude this section with establishing some short-and-long term programming goals that you would like to achieve.

5. Marking Scheme

A summary of the marking criteria for each component is defined below. For a detailed breakdown of the individual criteria please see **Appendix B: Detailed Marking Scheme**

Implementation (60 Marks)	**-**
C.1 Implementation of Application and Features	20
C.2 Use of Programming Concepts	20
C.3 User Interface and Interaction	10
C.4 Application of Good Coding Practices and Code Quality	10
Report (40 Marks)	**-**
C.5 Questions and Answers	10
C.6 Documentation	15
C.7 Testing	5
C.8 Reflection	10
Total Marks	**100**

Please Note

- Your Implementation is worth a maximum of 60 marks and Report is worth a maximum of 40 marks. The combined marks awarded from each component will form your overall grade for this coursework.
- Full details surrounding the grade boundaries can be found in the University's Assessment Handbook located here.

6. Submission Requirements

Please take your time when reading this section, as this contains specific information on how you should submit your coursework. Where you see `<Student-Name>` you should replace this with your full name. Please ensure that you follow the submission requirements defined below.

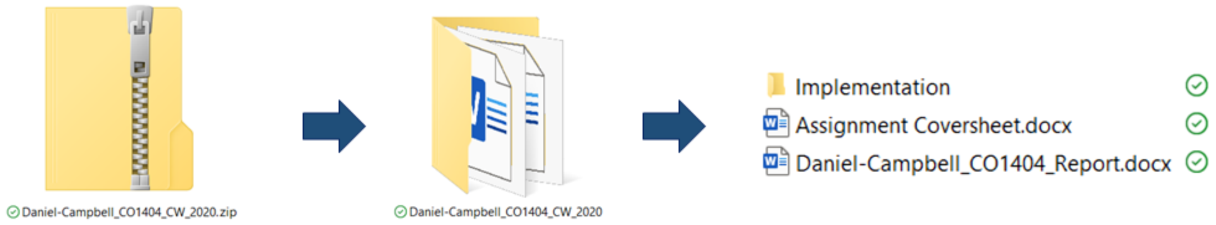
Coursework Constraints

- Your solution must only use only the default libraries provided by the .Net framework.
- You must use C# and developed your solution using Visual Studio.
- Your code must only consist of a single class - use as many methods and functions as you like.
- Only use the products and credit values listed in the products table.
- You must use the report template provided.

File Structure

A `zip` folder should be named `<Student-name>_CO1404_CW_2020.zip` and should contain the following files and folders listed below:

- Folder named `Implementation` containing your project.
- Your Report named `<Student-Name>_CO1404_Report.docx`.
- A completed Assignment coversheet found on blackboard.


Daniel-Campbell_CO1404_CW_2020.zip
Daniel-Campbell_CO1404_CW_2020
Implementation
Assignment Coversheet.docx
Daniel-Campbell_CO1404_Report.docx

Submission

You must submit your coursework submit a `.zip` file online through the appropriate link on Blackboard, by the hand in date stated at the beginning of this document. Please ensure you check the contents of your `zip` folder before uploading!

- **Late submissions**: Except where an extension of the hand-in deadline date has been approved, work that is handed in within 5 working days late will receive a maximum mark of 40%. Work handed in later than this will receive 0%.
- **Academic Malpractice**: The consequences of academic malpractice in assessments are serious. This includes plagiarism, collusion and allowing other students to access your work. This will not be tolerated. Details surrounding the coursework regulations can be found in the University's "Assessment Handbook" located here.

Below are tips that you may find useful when working on this assessment:

- Do not leave this assessment to the last minute.
- If you have any questions regarding this coursework, ask the module leader or module tutors.
- Give your self plenty of time to submit prior to the submission deadline.
- Use pen/cil and paper to work out the flow of your application.

7. Help and Support

- Support will be provided via Microsoft Teams (CO1404 channel), Email and you will also have the opportunity to ask questions during lectures / labs.

- For support with using library resources, please contact our subject librarian Bob Frost RSFrost@uclan.ac.uk. You will find links to lots of useful resources in the My Library tab on Blackboard.
- If you have not yet made the university aware of any disability, specific learning difficulty, long-term health or mental health condition, please complete a Disclosure Form. The Inclusive Support team will then contact to discuss reasonable adjustments and support relating to any disability. For more information, visit the Inclusive Support site.
- To access mental health and wellbeing support, please complete our online referral form. Alternatively, you can email wellbeing@uclan.ac.uk, call 01772 893020 or visit our UCLan Wellbeing Service pages for more information.
- If you have any other query or require further support you can contact The , *The Student Information and Support Centre. Speak with us for advice on accessing all the University services as well as the Library services. Whatever your query, our expert staff will be able to help and support you. For more information, how to contact us and our opening hours visit Student Information and Support Centre.*
- *If you have any valid mitigating circumstances that mean you cannot meet an assessment submission deadline and you wish to request an extension, you will need to apply online prior to the deadline.*

Appendices

Appendix A: Example Outputs

The examples below represent potential interactions and console outputs between the user and vending machine application:

Main Menu Example

```
-------------------------------
UCLAN Vending Machines LTD
-------------------------------
MAIN MENU
  1. Add Credits (current credits = 0.00)
  2. Select product\s
  0. Exit

Please enter a number: 1
```

Product Selection Example

```
PRODUCT SELECTION    [Current Balance = 2.00 credits]

Please choose from the folloing options:

  1. Chocolate Bar  [0.80 credits]
  2. Soda Can     [0.60 credits]
  3. Soda Water   [1.25 credits]
  4. Crisps     [0.50 credits]
  5. Cookies      [1.15 credits]
  0. Return to Main Menu
........................................
Please enter a number : 4
  You have added "Crisps" to your selection. Your current sub total = 0.50
credits. Would you like to add another product?
  Please input 'Y' for yes and 'N' : N
........................................
```

Purchase (sufficient credits) Example

```
........................................
Please enter a number : 4
  You have added "Crisps" to your selection. Your current sub total = 0.50
credits.
  Would you like to add another product?
  Please enter 'Y' for yes and 'N' : N
........................................
Available Balance   = 2.00 credits
Grand total     = 0.50 credits
----------------------------------------
Your new Balance  = 1.50 credits
----------------------------------------
Thank you for your custom!
```

Purchase (insufficient credits) Example

```
........................................
Please enter a number : 4
  You have added "Crisps" to your selection. Your current sub total = 0.50
credits.
  Would you like to add another product?
  Please enter 'Y' for yes and 'N' : N
........................................
Available Balance   = 0.00 credits
Grand total     = 0.50 credits
```

```
----------------------------------------
You have insufficent credits avaiable. You require "0.50" credits.
  Would you like to add more credits?
  Please enter 'Y' for yes and 'N' : Y
.........................................
Please enter how many credits you would like to add to your balance: 0.50
----------------------------------------
Your new Balance  = 1.50 credits
Would you like to continue processing your order?
  Please enter 'Y' for yes and 'N' : Y
........................................
Available Balance   = 0.50 credits
Grand total     = 0.50 credits
----------------------------------------
Your new Balance  = 0.00 credits
----------------------------------------
Thank you for your custom!
```

Appendix B: Detailed Marking Scheme

Below is a detailed breakdown of the marking scheme associated with the Implementation and Report aspects of this coursework.

Implementation Criteria (60%)

C.1 - Implementation of Application and Features (20 Marks)

C.1.1. Features (15 Marks)

You are awarded up to 5 marks for each feature: `Add Credits`, `Order Selection` *and* `Checkout`.

Mark Boundary	
0	No sufficient attempt has made to implement feature
1 - 2	A fair attempt has been made but feature is missing core aspects or contains bugs that prevent it from working
3 - 4	A good attempt at implementing the feature has been made but it contains minor bug/s.
5	Excellent Implementation of application functionality contains no bugs

C.1.2. Application Completeness (5 Marks

You are awarded marks based on the completeness of your application.

Mark Boundary	
0	No Functions Implemented or application fails to run.
1 - 2	One Functionality has been Implemented.
3 - 4	Two Features have been Implemented.
5	All Features have been implemented.

C.2 - Use of programming Concepts (20 marks)

Mark Boundary	
0	No attempt made.
1 - 5	Source code uses basic programming concepts (*variables and If statements*).
6 - 10	A failed attempt has been made to use some programming concepts. However, there are instances were concepts have been used ineffectively.
11 - 15	Very good application of a mixture of programming concepts in the majority of cases throughout the application.
16 - 20	Excellent application of a variety of programming concepts to solve problems logically throughout the entire source code.

C.3 - User Interface and Interactions (10 Marks)

Mark Boundary	
0	No attempt Made
1 - 3	Complicated user Interface, making the application difficult to use.
4 - 6	Creative user interface that prompts the user with various options. Displays some errors.
7 - 10	Creative, consistent and Intuitive user interface throughout the application. User is presented with useful messages to guide them throughout the application.

C.4 - Application of Coding Practices and Code Quality (10 Marks)

Mark Boundary	
0	No attempt Made
1 - 3	Demonstrated a fair attempt at commenting code. Code is untidy and very difficult to read. A naming convention for variables, methods and functions naming convention has been followed in some areas but not all.
4 - 6	Demonstrated good application of coding practices. Variables, methods and functions follow some what strict naming conventions and are clear, succinct. Code is some what readable and use of comments to describe complicated aspects shows a good of understanding of the code and logic.
7 - 8	Demonstrated very good application of coding practices. Most variables, methods and functions follow strict naming conventions and are clear, succinct. Code is readable and use of comments to describe complicated aspects shows a very good of understanding of the code and logic.
9 -10	Demonstrated excellent application of coding practices. All variables, methods and functions follow strict naming convention and are clear, succinct. Code is very easy to readable and use of comments to describe complicated aspects shows a high level of understanding of the code and logic.

Report (40%)

C.5 - Questions and Answers (10 Marks)

Question	Maximum Potential Mark/s
A	1
B	1
C	1
D	2
E	5

C.6 - Documentation (15 Marks)

Mark Boundary	
0	No attempt made.
1 - 4	Documentation is missing or contains several variables and or methods. There are a lot of errors with in the documentation (with respect to the code i.e types, parameters etc.) Descriptions are not clear or confusing. User guide lacks detial on how to use the application
5 - 8	Documentation is mostly complete. However some variable and or method descriptions are not clear or confusing. There are also minor errors making the doucmentation inaccurate. User guide proivdes steps on how to use the application but is not supported with clear screenshots.
9 - 11	Documentation is good. Variable and method descriptions are clear. Documentation is accurate in relatoin to your code. Userguide provides detailed instructions on how to use the application and is supported with clear screenshots
12 - 15	All variables and methods have been thoroughly documented. Variables and Methods have been given explict suitable names, use the correct data type and description clearly indicates it use and is accurate. User guide is excellent. Instructions surrounding Interactions and functionality are succinct and includes screenshot to assist the user. Userguide also incluldes description of error messgaes and how to resolve them.

C.7 - Testing (5 Marks)

Mark Boundary	
0	No attempt made.
1 - 2	Application partially fulfils testing criteria.
3 - 4	Application fulfils testing criteria. However there are some minor discrepancies such as accuracy.
5	Application fulfils all testing criteria. Values returned are accurate.

C.8 - Reflection (10 Marks)

Mark Boundary	
0	No attempt made.
1 - 3	Reflection lacks detail surrounding accomplishments, challenges, improvements.
4 - 6	Reflection in areas has details surrounding accomplishments, challenges and improvements. Some areas could be improved. Set short and long term goals.
7 - 8	Reflection contains details and context surrounding accomplishments, challenges and improvements throughout demonstrating good technical understanding. Long and short terms goals have been specified.
9 - 10	Reflection is clear and succinct. Accomplishments, challenges and improvements are discussed in detail throughout and demonstrates high technical understanding. Long and short term goals have been specified, and justified. Details have also been provided on actions that can be taken towards achieving the goals.

> *Disclaimer: The information provided in this assessment brief is correct at time of publication. In the unlikely event that any changes are deemed necessary, they will be communicated clearly via e-mail and a new version of this assessment brief will be circulated.*

> *Document Version : 1.0.0*

CO1404 Introduction to Programming Coursework
School of Psychology and Computer Science