|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | **ASSIGNMENT COVER PAGE** | | |  |
| **Programme** | | | **Course Code and Title** | | |
| Diploma in Computer Studies  Diploma in Information Technology | | | DDA1224 Data Structures and Algorithms  DDA1224N Data Structures and Algorithms | | |
| **Student’s name / student’s id** | | | **Lecturer’s name** | | |
|  | | | Rosmah Ismail | | |
| **Date issued** | **Submission Deadline** | | | **Indicative Weighting** | |
| Week 2 -10/6/2021 | Week 6- 8/7/2021 | | | 20% | |
| **Assignment [1]** | Equipment Hiring Information System (EHIS)  Assignment 1 –group of 4 members | | | | |
| This assessment assesses the following course learning outcomes | | | | | |
| **# as in Course Guide** | **UOWM KDU Penang University College Learning Outcome** | | | | |
| **CLO1** | Implement basic data structures | | | | |
| **Student’s declaration** | | | | | |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.  Student’s signature: Submission Date: | | | | | |

|  |  |
| --- | --- |
| **Dates and Mechanisms for Assessment Submission and Feedback** | |
| **Mechanism for handout to students** | Canvas LMS |
| **Mechanism for submission of work by student** | *Softcopy online submission via Canvas* |
| **Date by which work, feedback and marks will be returned to students** | 22 July 2021 |
| **Mechanism for return of assignment work, feedback and marks to students** | Feedback will be provided by a marking template. This will be available to students via Canvas. The discussions at the walkthroughs will also provide informal feedback |

# COURSEWORK SUBMISSION GENERAL INFORMATION

# Academic Integrity Statement

You must adhere to the university college regulations on academic conduct. Formal inquiry proceedings will be instigated if there is any suspicion of plagiarism or any other form of misconduct in your work. Students must **NOT** collude with other groups of students or plagiarize their work.

# Nature of the submission required

A softcopy of your assignment in **PDF version** should be submitted to lecturer, no later than the date and time stipulated on the cover sheet.

Diagrams may be used where they are helpful to support your arguments or description. If they are not your own work, the source must be referenced. Please help us to handle and mark your work efficiently.

Please take note for group submission, only **one submission per group**. This will contain both the group and individual elements. The individual element must be clearly labelled to indicate which group member completed the task.

# Documentation guidelines

Student is required to submit a **SOFTCOPY** of the report and ensure that it use the following formatted styles: 1) Font type: **ARIAL**, 2) Font size: **11** **pt**., 3) Line spacing: **Single spacing** and 4) Page layouts: **Justify**. Please make sure you have proper format alignment for all paragraphs, following standard writing style and use **HARVARD CITATION STYLE** for citation. Please include a **HEADER** with the following information: **Student ID, Student name, Course code and Assignment type**. Please also include a proper cover page for your submission which contains information about the students, assignment, course, and department with KDU logos on top. Also include page number and list of references, which is shown in the last page.

# Penalties for Late Submission

For late submission of this Assignment, a penalty of a reduction by 10% of the maximum mark may be applicable for each Calendar Day or part thereof that the submission is late. An Assignment submitted more than **TEN** Calendar Days after the deadline will have a mark of zero recorded for this Assignment.

# Submission arrangement

1. Cover page
2. Table of Content
3. Main Report
4. Reference List or Bibliography List (whichever applicable)
5. Marking Rubric (in landscape orientation)

# Assignment instructions/Background

You have been hired on a short-term contract by a small software development company. The company is developing a program for use by PC users to manage information. Recently, the company obtained an offer from a small family business company to develop a system to keep track of their business records system. Below is a brief information of the system.

Greenhill Hire firm is a small family business that stocks general equipment such as garden tools, car maintenance tools etc. for hiring to its customers. Equipment is classified under various categories. Equipment is of different brands within each category. Brands vary in their prices and are obtained from various suppliers. The known categories are:

a) Gardening Equipment

b) Building Equipment

c) Decorating Equipment

d) Car Maintenance

e) Miscellaneous

The company wants you to use C++ to develop the program; its working name is Equipment Hiring Information System (EHIS). It will eventually be integrated with other programs the company is working on, for instance to allow the equipment obtained from various suppliers, but at the moment you are just to work on a standalone program.

**Requirements**

The computer is to be used when a customer visits the company to hire equipment. At this time the following processing is necessary for each transaction:

• The customer's details are recorded.

• The equipment to be hired, with expected return date, is recorded.

1. The initial Requirements Analysis for EHIS has been completed and documented in the form of a Use Case diagram. Where information is not available, or if there are any ambiguities in the requirement you are to make and document sensible assumptions.

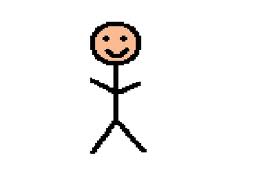


Figure 1: Use Case diagram for the EHIS Program

1. There is no need to store the record in a file or a database. You may store the records in an array. You may use structure and array.
2. For readability and maintainability purposes, you are required to use functions to implement the cases shown in the Figure 1 above.

**Deliverables**

1. A system with the initial requirement specified above.
2. A report in Word format. The report should consist of the following item:
   * Introduction – a brief description on the system that you have developed.
   * Functionality of the system.
   * Screen shots of the system, consisting screen shots of your program in operation to demonstrate the functionality you have achieved.
   * Strengths and weaknesses of the program.
   * Individual self-reflection. The reflection must include the summarized understanding of the system implemented, the challenges, and the soft skills that you have gained or improved in completing the coursework.

The word length of your report should be in between 1000 and 1500 words.

Note: marks awarded based on the following criteria:

* Functionality
* Code quality
* Use of appropriate data structures
* Modularity of the program (use of functions, parameters)
* Readability of the programs (indentation, meaningful variables names, comments, etc)
* Report

Submission Instructions:

Upload the following items through Canvas:

* + Program code (C++ program)
  + Report in softcopy format

**Introduction**

The Equipment Hiring Information System that have been developed is a system that handles equipment hiring made by customers. The system has a fixed equipment inventory that can only be modified by changing the source code. Each equipment has their own attributes such as category, item description, brand, quantity and price per item. Hired equipment have 2 additional attributes for its hiring start and return date. With this system, customers will be able to search for tools that they are looking for easily because they are organized appropriately in their respective categories. Customers can also check the details of their hiring as well to make sure that they did not hire the wrong equipment using the functions that are provided with the system.

**Functionality of the system.**

Before interacting with the system, A welcome message is displayed, and the system will prompt a user input to let users enter their name, email and phone number. Then, a main menu is displayed. In the main menu, there are a few functions available for the users to choose, namely creating a hiring, listing equipment returns on a specific date, counting the number of hiring in each category and listing all equipment by their category. The usage of each function is explained below: - //intro + intro to func = 208

* **Creating a hiring**

This function allows users to create a hiring for the equipment available in the store’s inventory. Upon selecting this function, an equipment category submenu is displayed, and the system prompts the user to enter the category that they want to browse in. Based on the chosen category, a list of tools is displayed in a menu, and another input is prompted for the user to select the tool that they want to hire. A list of brands available for the selected tool is displayed after that with their price and quantity, and the user is asked again to input their choice. Every input that the user enters must be a valid data type, otherwise the system will repeat its process until it gets one.

After identifying the tool that the user wants to hire, the system asks the user to enter the quantity of the tool to hire, the hiring’s start date and return date. Then, the system displays the information of the tool hiring that were gathered previously and let users confirm whether they want to hire the tool. If the user enters ‘Y’, the system records the details of the hiring, and displays that the process was successful. Conversely, if the user enters ‘N’, the system returns the user back to the main menu and displays that the process was unsuccessful. The user will be returned to the main menu regardless of both inputs, and the confirmation process repeats if the user enters neither inputs.

This function is only usable if the hiring limit has not been reached. The system will notify the user that they reached the hiring limit if they do after creating a hiring. //279

* **Listing equipment returns on a specific date**
* **// assume 150 words**
* **Counting the number of hiring**

This function displays the number of hiring made by the user for each equipment category. Upon selecting this function, the system displays the number of hiring that the user had made for each equipment category. The system simply displays 0 as result if a category has no tool hiring. Finally, the system returns the user back to the main menu. //60

* **Listing all equipment by their category**

This function lists all equipment that have been hired based on their category. Upon selecting this function, the system displays 5 tables representing each equipment categories, and each table displays the details of hired tools that were categorized under it. If no hired tools can be found in a table, the system displays that there are no hiring records for that table. If the user has not hired any tools, the system displays that the user has not made any hiring record. Finally, the system returns the user back to the main menu. //93

**Screenshots**

**Strengths and weaknesses of the program.**

One of the strengths of this program is the user can clearly identify the tools that are available for hiring. The program also mimics a graphical user interface that is advantageous when compared to text-based console applications. The system provides guidance on what kind of input should users enter too.

Though, there are some flaws of the program. One of it is that the program still records some property values like date values regardless of whether they are in the correct format. Another flaw is, the program gets cramped after a certain amount of use and the screen does not clear past outputs. //103

**Individual self-reflection**

* Lim Zhe Yuan

In the beginning of the assignment, we discussed about the concepts of our program using language that was too complicated. This leads us to little understanding of our program concepts and the way we write our code. To overcome this, we had to spend more time rewriting our code and examining parts of the system to determine the logic behind each operation and see whether they are relevant to the assignment questions. Thinking back, we simply just need to communicate more with ourselves, determining a concrete program concept first before giving in-depth walkthrough on how we should write the program. Doing this ensures that we will not have ideas that conflict with the main concept we have decided on when we write our program.

* Thor Wen Zheng
* Tan Peng Heng
* Adam John Simpson

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **DDA1224 Data Structures and Algorithms**  **MARKING RUBRIC**  **ASSIGNMENT 1**  **(20%)**  **Student1(S1): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Student2(S2): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Student3(S3): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Student4(S4): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_** | | | | | | | | | | |
| **Coding(70%)** | | | | | | | | | | |
| **Learning Outcome** | **MARKING CRITERIA** | **SCALE** | | | | | | | | |
| **Not Achieved (0-4.5)** | **Just Achieved (5.0 - 5.5)** | **Proficient (6.0 - 6.5)** | **Very Good (7.0 - 7.5)** | **Excellent (8.0-10.0)** | **YOUR MARKS/COMMENTS** | | | |
|  | | | |
| CLO1: Implement basic data structures | 1. Code Quality  (15%) | Very poor attempt. Unstructured, no comment/remarks, non meaningful variables used. | A poor attempt; which may be several problems with structure, or very little use  has been made of comments, or the  naming of classes, methods and variables is unsatisfactory in a  significant number of cases. | quality with several omissions of naming  and use of comments. | of comments, and where the majority of  classes, variables and methods have  been appropriately named. However  there may be several omissions of  Javadoc comments, and the code. | Good use of commenting throughout,  including Javadoc comments for the  vast majority of classes, methods and  variables. |  | | | |
| 2. Modularity (use of  classes/functions, with  parameters and returned  value/ collection)  (10%) | Non modular program. No function used. | Demonstrate some limited use of  classes/ functions/ collection. | Demonstrates appropriate use of  classes/ functions/ collection. | Demonstrates proficiency in use of  classes/ functions/ collection. | Demonstrates mastery in the use of  classes/ functions/ collection. |  | | | |
| 3.Use of appropriate data structure (15%) | No data structure used | Demonstrates limited use of data  structure and algorithm. | Demonstrates reasonable use of data  structure and algorithm, but with a few  shortcomings. | Demonstrates proficiency in use of data  structure and algorithm. | Demonstrates complete and proper use  of data structure and algorithm. |  | | | |
| 4. Functionality (Program execution and  output quality)  (20%) | Non executable program | Program executed with runtime error but  achieve partial program requirements. | Program executed error free with  limitations to achieve minimum program  requirements. | Program executed error free with  correct output and achieve all program  requirements. | Program executed error free with  excellent output with appropriate  validation. |  | | | |
| 5. Readability of the program  (10%) | Non readable program | readability with poor indentation, variable names, and remarks | readability with poor indentation, but meaningful variable names, and remarks | readability with good indentation, meaningful variable names, but less remarks | Excellent readability with good indentation, meaningful variable names, and remarks |  | | | |
| **Total: 70%** | | | | | |  | | | |
| **Report (30 %)** | | | | | | | | | | |
| **Learning Outcome** | **MARKING CRITERIA** | **SCALE** | | | | | | | | |
| **Not Achieved (0-4.5)** | **Just Achieved (5.0 - 5.5)** | **Proficient (6.0 - 6.5)** | **Very Good (7.0 - 7.5)** | **Excellent (8.0-10.0)** | **YOUR MARKS/COMMENTS** | | | |
|  | | | |
| CLO1: Implement basic data structures | 1. **Introduction**   **(5%)** | Little or insufficient definition of a data structure | Basic definition of a data structure.One or two topics is/are left out. | Able to define a data structure and its services but some parts are without adequate explanation/support evidence. | Able to define mobile a data structure with adequate explanation /support but there are only minor parts need further clarification. | Author is able to define a data structure with adequate support of journal papers. |  | | | |
| **2. Quality of information**  **(10%)** | Information not clearly relates to the main  Topic. | Information clearly relates to the main  topic. Points are insufficiently  developed. Analysis is weak | Information clearly relates to the main  topic. Points are made, but analysis is  minimal | Information clearly relates to the main  topic. Points and analysis are made and  related to the topic. | Information clearly relates to the main  topic. Points are clearly made. Analysis  is sophisticated |  | | | |
| **5.Style of writing**  **(5%)** | The report writing does not meet the criteria for the assignment (too short or incomplete, too long, and/or completely off-topic). Reference section is missing. | Many ideas require clarification and/or are off-topic or have marginal relevance to the assignment. Many grammatical and/or spellings errors throughout the paper. Improper reference section | Ideas are stated clearly and are related to the topic, with only adequate grammatical and/or spelling errors. Reference section with minor flaws | Most ideas are stated clearly and are related to the topic, with only minor grammatical and/or spelling errors. Reference section is in minimal | Writing is clear and relevant, with no grammatical and/or spelling errors – polished and professional. Reference section properly formatted. |  | | | |
|  | **6. Individual self-reflection**  **(10%)** | Student unable to demonstrates accurate, extensive, and deep understanding of the topics, where connection to a wide range of context, and reflection are not evident. | Student strive to demonstrates accurate, extensive, and deep understanding of the topics, where connection to a wide range of context, and reflection are partly evident. | Student strive to demonstrates accurate, extensive, and deep understanding of the topics, where connection to a wide range of context, and reflection are evident. | Student demonstrates accurate, extensive, and deep understanding of the topics, where connection to a wide range of context, and reflection are evident. | Student demonstrates very accurate, extensive, and deep understanding of the topics, where connection to a wide range of context, and reflection are clearly evident. | **S1** | **S2** | **S3** | **S4** |
|  |  |  |  |
| **Total 30%**  **:** | | | | | | |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Overall score (100%)** |  |  |  |  |