



DESCRIPTION OF COURSEWORK

Course Code	CST207
Course Name	Design and Analysis of Algorithms
Lecturer	Dr. Mohammed N. M. Ali
Academic Session	2023/09
Assessment Title	Project (Group)

A. Introduction/ Situation/ Background Information

This project evaluates the student's ability to use and apply different algorithms. Especially the searching and sorting algorithms to solve a problem and explain the solution by analyzing and comparing different algorithms.

B. Course Learning Outcomes (CLO) covered

At the end of this assessment, students are able to:

CLO 3 Demonstrate problem-solving skills via concepts, techniques, and proficiencies in algorithm analysis which leads to potential applications in the self-development stage e.g., further studies.

C. University Policy on Academic Misconduct

1. Academic misconduct is a serious offense in Xiamen University Malaysia. It can be defined as any of the following:
 - i. **Plagiarism** is submitting or presenting someone else's work, words, ideas, data or information as your own intentionally or unintentionally. This includes incorporating published and unpublished material, whether in manuscript, printed or electronic form into your work without acknowledging the source (the person and the work).
 - ii. **Collusion** is two or more people collaborating on a piece of work (in part or whole) which is intended to be wholly individual and passed it off as own individual work.
 - iii. **Cheating** is an act of dishonesty or fraud in order to gain an unfair advantage in an assessment. This includes using or attempting to use, or assisting another to use materials

that are prohibited or inappropriate, commissioning work from a third party, falsifying data, or breaching any examination rules.

2. All the assessment submitted must be the outcome of the student. Any form of academic misconduct is a serious offense which will be penalised by being given a zero mark for the entire assessment in question or part of the assessment in question. If there is more than one guilty party as in the case of collusion, both you and your collusion partner(s) will be subjected to the same penalty.

D. Instruction to Students

This project is **Group work**. Each group should be constructed with **2-5** members (**Maximum 5 members**). You also have the choice to do it individually. Each group should submit a file of the code written in either **C (filename.c)** or **C++ (filename.cpp)** (**Other programming languages not accepted**) and a pdf report:

- (i) “<Student_Name>_<Student_ID>_CST207_Project_Group_202309.pdf”
(Use the name of one student from the Group).
- (ii) “<Student_Name>_<Student_ID>_CST207_Project_Group_202309(.c, or .cpp)”.

Please attach the **Cover page** and **Marking Rubrics** with your Project in a **.pdf** file. The deadline for submission is **18:00, 22nd December 2023**. An overdue penalty will be applicable to the project that is submitted after the deadline.

*** Your codes will be sent to a **Plagiarism** detection system for duplication checking. Please write your codes independently. (Modify your code if you copy some fragment from the Internet because your classmates may copy the same fragment.)**

E. Evaluation Breakdown

No.	Component Title	Percentage (%)
1.	Project	100
	TOTAL	100

F. Task(s)

A hotel management system (HMS) is a comprehensive software solution designed to streamline and automate the day-to-day operations of a hotel or hospitality establishment. It encompasses a wide range of functionalities to facilitate efficient management of various hotel functions, enhance guest experience, and optimize overall operational efficiency. Consider that you are a Software developer working for a software development company. One of the tasks assigned to you is creating a **Hotel Management System** to manage a five-star hotel.

You are required to develop software to manage the hotel operations. In our case, we will customize the hotel system to include three control panels. The first is the **system panel**, the second is the **admin control panel** and the third is the **Reception control panel**.

You are required to complete an algorithm using C, or C++ programming languages to form this system, the formed system should consider the following:

1. The System panel:

- a. Secure the system by creating a login screen to allow only the authorized person to log in. (username: **user**, and password: **123456**).
- b. Create a main screen as a control panel to allow the authorized user to move between the different parts of the system including logging in as admin, log in as a reception staff, or exit from the system.
- c. Allow the authorized user to choose the task from the above three tasks (login as admin, login as reception staff, or exit from the system).

2. The Admin Control Panel:

- a. If the user chooses to log in as an Admin; he should enter a username and password (username: **user**, and the password: **123456**).
- b. After logging into the system successfully, the admin control panel will appear.
- c. The admin control panel include the following items:

- i. Add Employee (ID, Name, age, gender, job, salary, phone, address, email)
- ii. Add Room (Room number, status (available or occupied), cleaning status (clean or dirty), price, bed type (single or double), discount percentage).
- iii. Search Employee (use the Employee ID for the search). Apply the (Binary Search Algorithm).
- iv. Edit Employee data. Use the ID to search for the Employee. Apply (Binary Search Algorithm).
- v. Delete Employee. Use the ID to search for the Employee. Apply (Linear Search Algorithm).
- vi. Edit Room. Use the room number to search for the room. Apply (Binary Search Algorithm).
- vii. Show Rooms (unsorted).
- viii. Show the Rooms sorted by number. Apply the (Quick-Sort Algorithm).
- ix. Show the Rooms sorted by price (Consider the last price of the room after applying a discount (if any)). Apply the (Bubble Sort).
- x. Show the details of all complete booking operations. Allow the admin to search by date (Use Binary Search). Show the total price of all booking operations per day.
- xi. Show the receipts generated by the system for all booking operations (random receipt number, Date, room details (room number, type), and total amount). Sort the receipts by the number. Apply the (Selection Sort Algorithm)
- xii. Go back.

3. The Reception Control Panel:

- a. If the user chooses to log in as a reception staff; he should enter a username and

password (username: [user](#), and the password: [123456](#)).

- b. After logging into the system successfully, the reception control panel will appear.
- c. The Reception control panel should have the following items:
 - i. New customer form (This form includes Customer ID type (passport, IC, etc.), ID number, name, gender, country, allocated room number, check-in date and time, number of staying days, total amount, initial payment, pending amount, and deposit). [Pending amount = total amount – initial payment](#).
 - ii. Show all Rooms (unsorted).
 - iii. Search for the available rooms. ([Apply Binary Search](#)).
 - iv. Show the single bed Rooms sorted by price. Apply the ([Merge Sort Algorithm](#)).
 - v. Show the double bed Rooms sorted by price. Apply the ([Bubble Sort Algorithm](#)).
 - vi. Show the information of all customers.
 - vii. Search for a customer by ID ([Use Linear Search Algorithm](#))
 - viii. Search for a customer by room number ([Use Binary Search Algorithm](#))
 - ix. Update room status. Use the room number to search for the room. Apply ([Binary Search Algorithm](#)).
 - x. Update Customer information. Use ID to search for the customer. Apply ([Binary Search Algorithm](#)).
 - xi. Show min and max price. Apply the [min and max algorithm](#) to show the minimum and maximum price rooms.
 - xii. Delete Customer. Use ID to search for the customer. Apply ([Binary Search Algorithm](#)).
 - xiii. Checkout form. (Customer ID, room number, check-in time, checkout time,

deposit, pending payment). The system should search for the customer by ID, then any saved information must be shown in the checkout form. (Use Binary Search Algorithm). At the end of this operation, the room status should be changed from occupied to available.

- xiv. Show an invoice for the customer, including the details (**customer ID, customer name, room number, initial payment, pending payment, and deposit**). The system should search for the customer by ID, then all customer information must be shown. (Use Linear Search Algorithm).
- xv. On the same screen, **below the invoice**, show a message to ask for payment (If any pending payment), the system will collect the customer payment (**Example: Enter the amount tendered: (amount)**); then will display the change. (**For example**, if a customer paid **200 RM** and the total amount of the invoice is **50 RM**, the money will be given back to him (**Change**) is 150 RM). If the customer paid an amount less than the pending amount of the invoice, show a message saying “**Money not enough, enter sufficient amount**”. If the amount is enough; the system will show a **receipt** ready for printing. **This receipt contains** the customer's details (ID and name), the total price, the customer's initial payment entered into the system, and the change. The receipt must have a unique random number, date, and hotel name (**Choose any name**). If the same customer booked more than one room; the rooms in the receipt must be sorted in **nondecreasing** order by **price**. (Use the selection sort algorithm). (**Note: The number of the receipt should be generated randomly. You may use a random function to generate this ID**).
- xvi. Go back. (The system will bring him back to the main control panel).

After building and running your algorithm, you are required to complete a report and do the following: (The handwriting is not allowed in this part).

1. Present your work through a short video that explains what you have done and how to run your program. The length of the video should **not exceed 7 minutes** as a maximum and the size does **not exceed 100 MB**.
2. Compare the searching algorithms you used in your system and show the difficulties you faced when applying them (if any).
3. In your opinion which searching algorithm is best and easiest to implement (**regardless of the time and space complexity of this algorithm**); justify your answer.
4. Compare the sorting algorithms you used in your system and show the difficulties you faced when you applied them (if any).
5. In your opinion which sorting algorithm is best and easiest to implement (**regardless of the time and space complexity of this algorithm**); justify your answer.
6. Elaborate your answer using resources from the internet, when possible, support your answer and cite the references at the end of your report.
7. Explain in brief your algorithm (The system you created) and paste screenshots from the system in your report.
8. At the end of your report, copy the code of the algorithm in the appendix. (You can take screenshots from your code).
9. The report must not exceed 10 pages. (**Excluding the appendix and the references**)
10. **To assess the personal skills** of each student in the group; you should write in your report each student's contribution. (**Each student in the group must participate in creating the system and writing the final report**).
11. Be sure your report format is well written without grammar and spelling mistakes.

12. Plagiarism should be less than 20%.

APPENDIX 1

MARKING RUBRICS

Component Title						Percentage (%)		
Criteria	Score and Descriptors					Weight (%)	Marks	
	Excellent (90-100)	Good (80-89)	Average (60-79)	Need Improvement (50-59)	Poor (0-49)			
Quality of implementation	The program runs efficiently without errors. Adheres to good programming practices.	Provide more than 50% correct code. Somewhat adheres to good programming practices.	Provide up to 50% correct code. Room to improve on good programming practices.	Attempted to complete code but with many logic errors. Some application of good programming practices.	Unable to provide any correct code. Poor/no application of good programming practices.	30		
The implementation and justification of the search algorithms	Implement the algorithms correctly with convenient justification. Solutions are properly organized, with accurate and great details.	Implement the algorithms correctly with convenient justification. Solutions are properly organized and with sufficient details.	Implement the algorithms and justify. Some are correct	Inaccurate implementation and justification.	Use different algorithms, very little work, or no justification.	15		
The implementation and justification of the sorting algorithms	Implement the algorithms correctly with convenient justification. Solutions are	Implement the algorithms correctly with convenient justification. Solutions are	Implement the algorithms and justify. Some are correct	Inaccurate implementation and justification.	Use different algorithms, very little work, or no justification.	15		

	properly organized, with accurate and great details.	properly organized and with sufficient details.					
Report Format	The report is well written without grammar and spelling mistakes. Good presentation format. All sections are present. A good set of references.	Report format adhered to. Grammar is mostly sound. All sections are present. Some references.	Some grammatical mistakes. The formatting needs improvement. Minimal references.	Poor format. Plenty of grammatical mistakes. Minimal or no references.	The format was completely ignored. Major inconsistencies and/or no references.	20	
The assessment of each student's personal skills in each group by measuring the student's contribution to the project.	The highest contribution student	The next highest contribution student	The medium contribution	The lowest contribution	The student did not have any contribution	20	
TOTAL						100	

Note to students: Please print out and attach this appendix together with the submission of coursework