# A car with its interior and engine Description automatically generated with medium confidence

Mastering Embedded System Online Diploma

www.learn-in-depth.com

First Term (Final Project 2)

Eng. Habeba Ahmed Elbaghdady

Table of Contents

1. Problem Statement
2. Approach
3. Idea
4. Main.c
5. Queue.h
6. Queue.c

* Create Queue.
* Enqueue.
* Dequeue.
* Is Empty.
* Is Full.
* Print Queue.

1. App.h
2. App.c

* Choice.
* Add Students by File.
* Add students Manually.
* Find by Roll Number.
* Find by First Name.
* Find by Last Name.
* Find by Course Id.
* Find Total Number of Students.
* Update Student by Roll Number.
* Delete Student by Roll Number.
* Show All Students.
* Check Roll Number.
* Exit Function.

Problem Statement

A Simple Software for Student Management System Which Can perform the following operations:

1. Store the First Name of the Student.
2. Store the Last Name of the Student.
3. Store the Unique Roll Number for every Student.
4. Store the GPA of each Student.
5. Store the Courses registered by the student.

Approach:

The idea is to form an individual functions for every operation.

All the functions are unified to form the software:

* Add Student Details from File.
* Add student Details Manually.
* Find the Student by the given Roll Number
* Find the Student by the given First Name.
* Find the Student by the given Last Name.
* Find the Student registered in a course.
* Count of Students.
* Delete a Student.
* Update Student.
* Checking Roll Number in the Data
* Exit

Idea:

The software consists of 5 Files Main.c Queue.h Queue.c App.c App.h.

The main contains the calling functions.

The Queue.h contains the declarations of queue functions.

The Queue.c contains the implementations of queue functions.

The App.h contains the declarations of Student Management System functions.

The App.c contains the implementations of Student Management System functions.

Main.c

* The queue is initialized using create\_queue function.
* A welcoming message is shown to the user.
* The choice function is called infinitely.



Queue.h

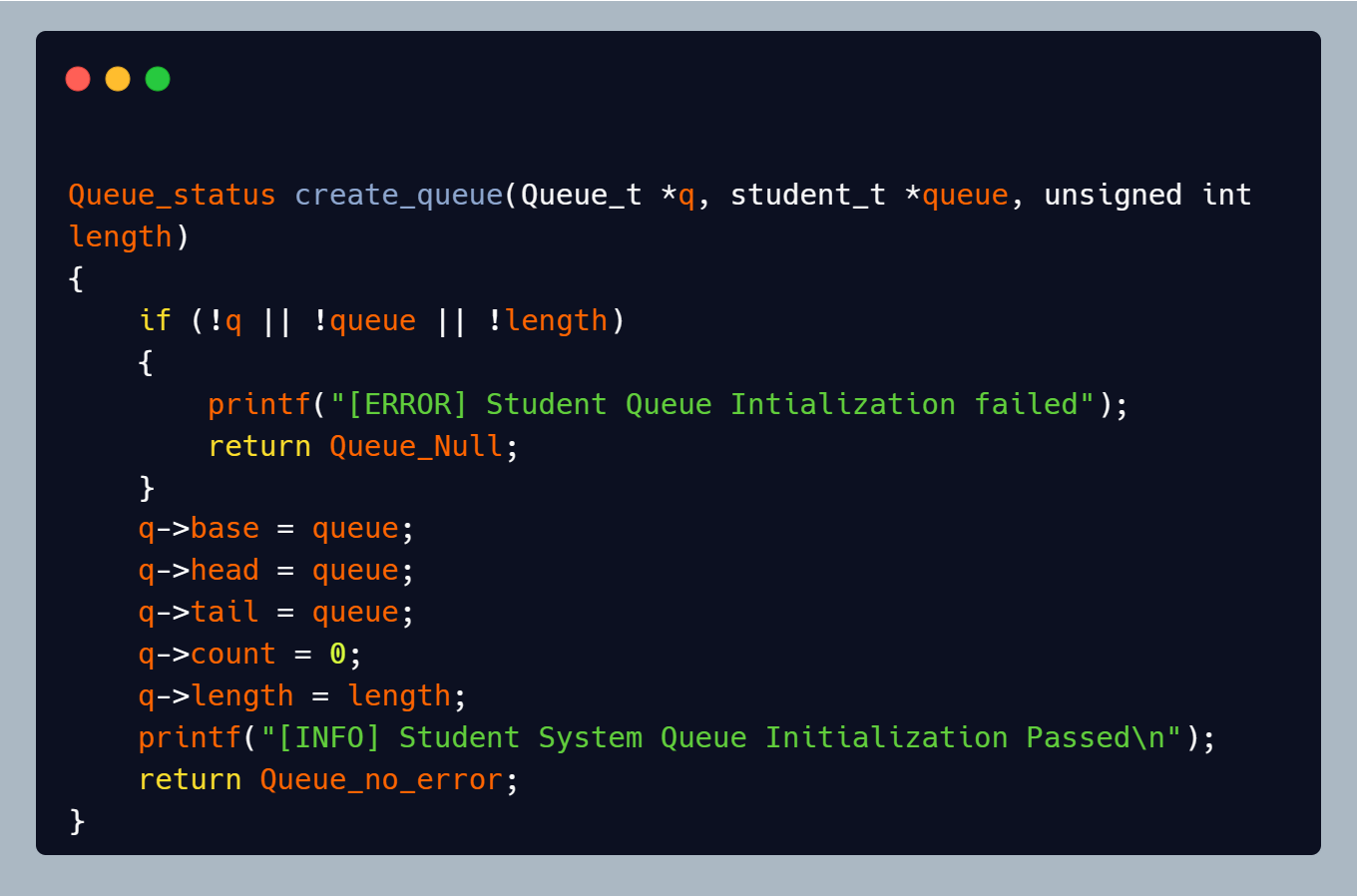
* It includes the libraries needed for the program to run.
* A struct of type student holds the variables which will have the data of each student in the system.
* A struct of type queue.
* An enum of type status that includes the status of each operation done in the queue.
* Functions declarations.



Queue.c

* create\_queue

The function is used to initialize the queue.



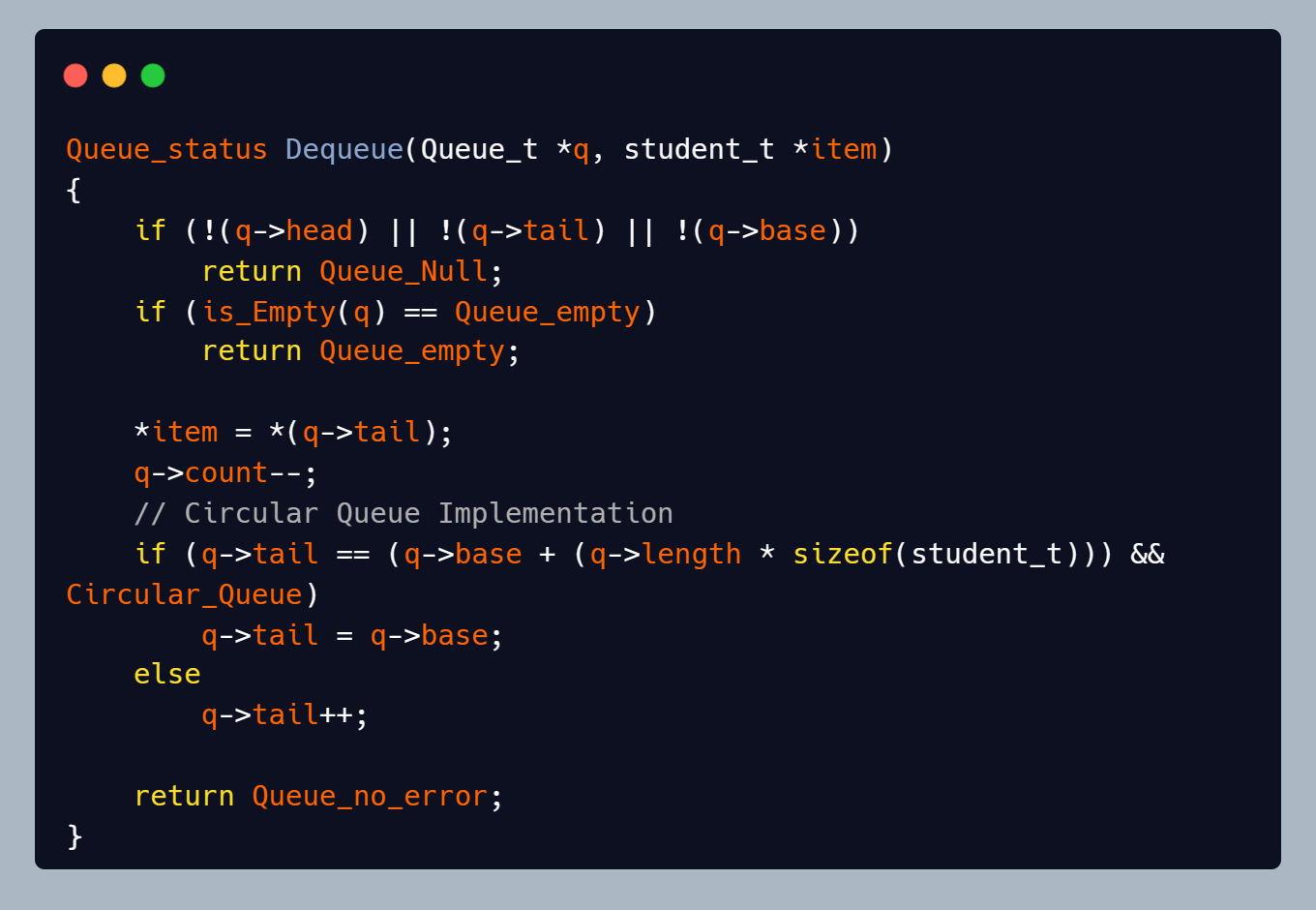
* Enqueue

The function is used to add a value in the queue.



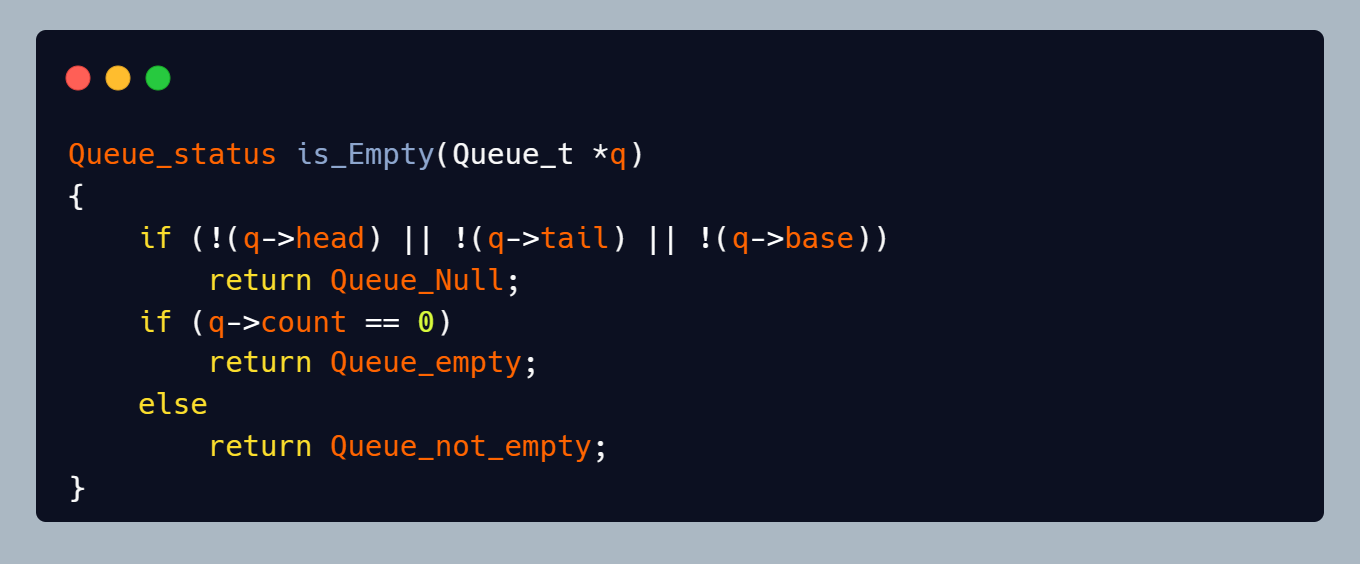
* Dequeue

The function is used to remove the value from the queue.



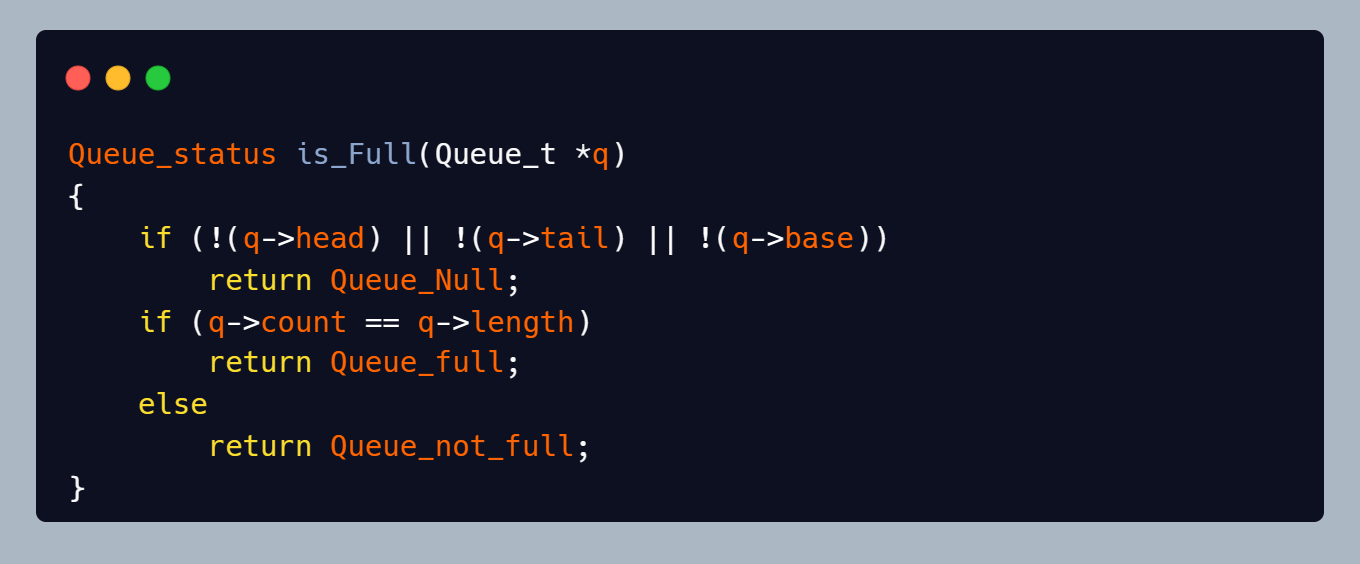
* is\_Empty

The function is used to check if the queue is empty or not.



* is\_Full

The function is used to check if the queue is full or not.



* printQueue

The function is used to print the values of the Queue.



App.h

* It includes the Queue.h File.
* It has the creation of Queue\_List and student.
* Functions Declarations.

A screenshot of a computer program

Description automatically generated

App.c

* Choice

The Function is used to show the user all the functions in the system and asks him which function he wants to use.



* Add\_student\_file

The function is used to add students into the system using a text file that contains the details of the students.



* Add\_student\_manually

The function is used to add students to the system by the user Manually.



* Find\_by\_roll\_number

The function is used to search for a student record by his roll number.



* Find\_by\_first\_name

The function is used to search for a student record by his First Name.



* Find\_by\_last\_name

The function is used to search for a student record by his Last Name.

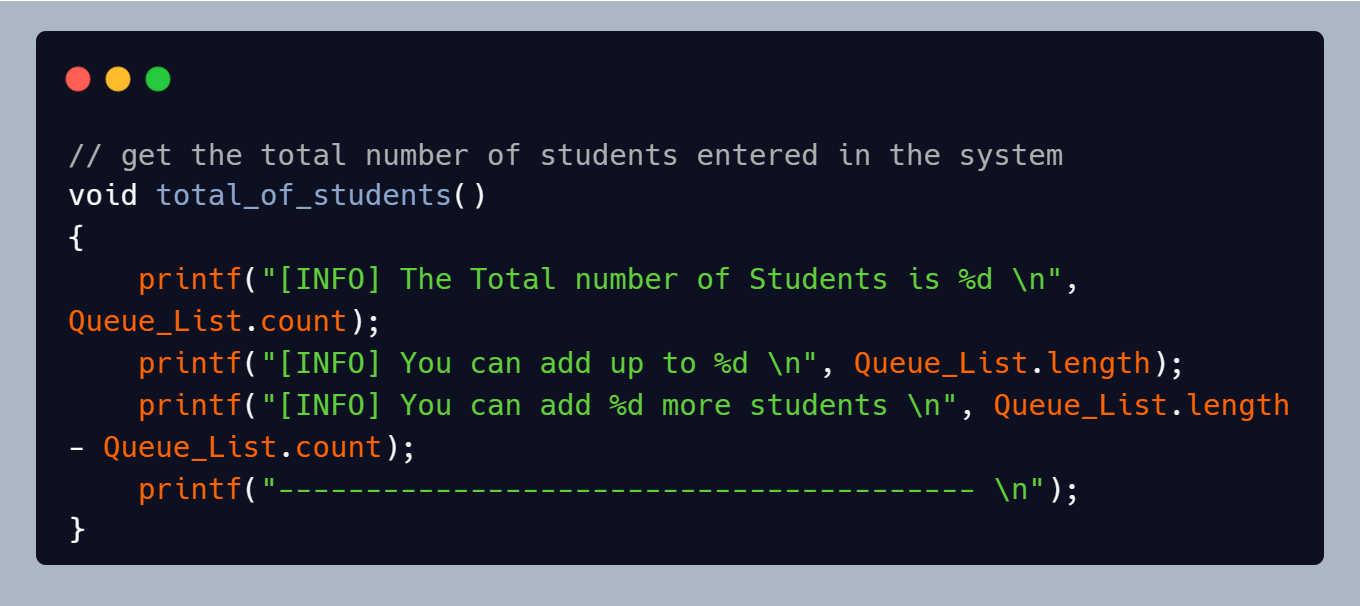


* Find\_students\_by\_courses

The function is used to search for students record by the course id they are in.



* Total\_of\_students

The function is used to get the total number of students in the system.

* Delete\_student

The function is used to delete a student record from the system.



* Update\_student

The function is used to update any detail in the student record, the roll number can be changed if the new number does not match any roll number in the system.



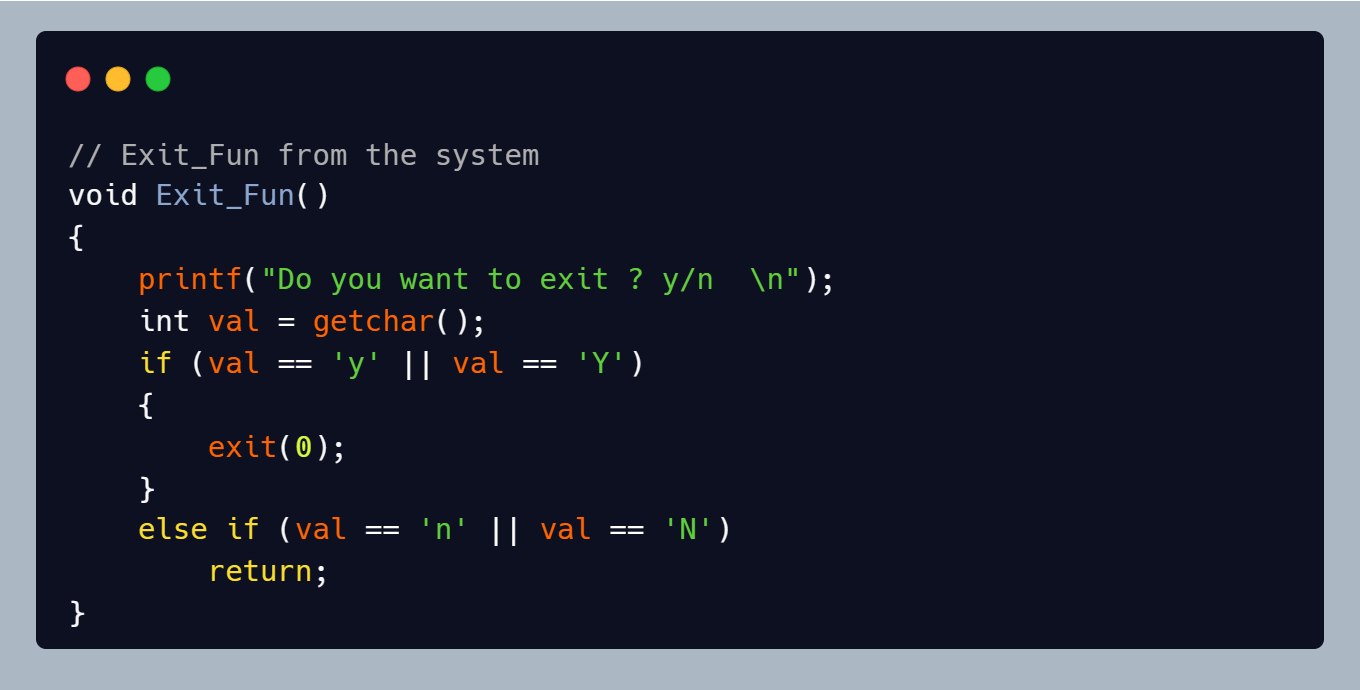
* Show\_all\_students

The function is used to show all students recorded in the system.



* Exit\_Fun

The Function is used to exit from the system if the user wants to.



* Roll\_Check

The function is used to check if the roll number exists in the system or not.

It is used in adding and updating records, so the user won’t add an already recorded roll number.

