

Lab Final

Data Structure and Algorithm lab, Time: 35 min

Problem 1: Managing Library Book Borrowing System

Scenario:

The Department of Library Management is developing a software module to manage the queue of students waiting to borrow newly arrived books. Since books with high demand often attract many students, the system maintains a queue to ensure that the students are served in a fair, first-come first-served manner.

You have been asked to create a simple queue-based program for this purpose. The system should allow students to join the queue, allow the librarian to serve (remove) the next student in line, and also let the librarian view the current first student waiting.

A partially completed Python code for queue implementation using a list is already provided. You must use the given class Queue to perform the required operations.

Task:

Write a Python program using the given queue implementation to simulate the book-borrowing queue system.

Your program must:

1. Create an empty queue. Accept the number of students who will initially join the queue.
2. Accept each student's name and insert them into the queue using `enqueue()` and display the current queue
3. Allow the librarian to serve the next student by performing a `dequeue()` and Display the name of the student served.
4. Print the current student at the front of the queue using `front()`.
5. Display the final state of the queue.

Sample input

Enter number of students in queue: 3
Enter student 1 name: Arif
Enter student 2 name: Sumaiya
Enter student 3 name: Rahim

Sample output

Queue: ['Arif', 'Sumaiya', 'Rahim']
Served student: Arif
Next in queue: Sumaiya ✓
Updated Queue: ['Sumaiya', 'Rahim']

Problem 2 : Sorting Product Prices**Scenario:**

A retail shop is upgrading its billing software to organize product prices more efficiently. The system receives various product prices that may include positive, negative, or zero values due to discounts, returned items, or promotional adjustments. To prepare the price list for further processing—such as generating reports or displaying sorted prices to customers—the shopkeeper wants the software to sort all product prices in ascending order using a simple sorting technique. Your task is to write a Python program using the given Selection Sort implementation to help the shopkeeper organize the product prices.

Task:

1. Accept the number of product prices to be entered.
2. Accept each product price from the user and store them in a list.
3. Use the provided selectionSort() algorithm to sort the prices in ascending order.
4. Display the list after each sorting step (as shown in the given output).
5. Finally, print the fully sorted list of product prices.

Sample input

Enter number of product prices: 5
Enter price 1: -2
Enter price 2: 45
Enter price 3: 0
Enter price 4: 11
Enter price 5: -9

Sample output

Step : 0
[-9, 45, 0, 11, -2]
Step: 1
[-9, -2, 0, 11, 45]
Step : 2
[-9, -2, 0, 11, 45]
Step : 3
[-9, -2, 0, 11, 45]
Step: 4
[-9, -2, 0, 11, 45]
Sorted product prices in ascending order:
[-9, -2, 0, 11, 45]