Data Structures PROJECT
2023

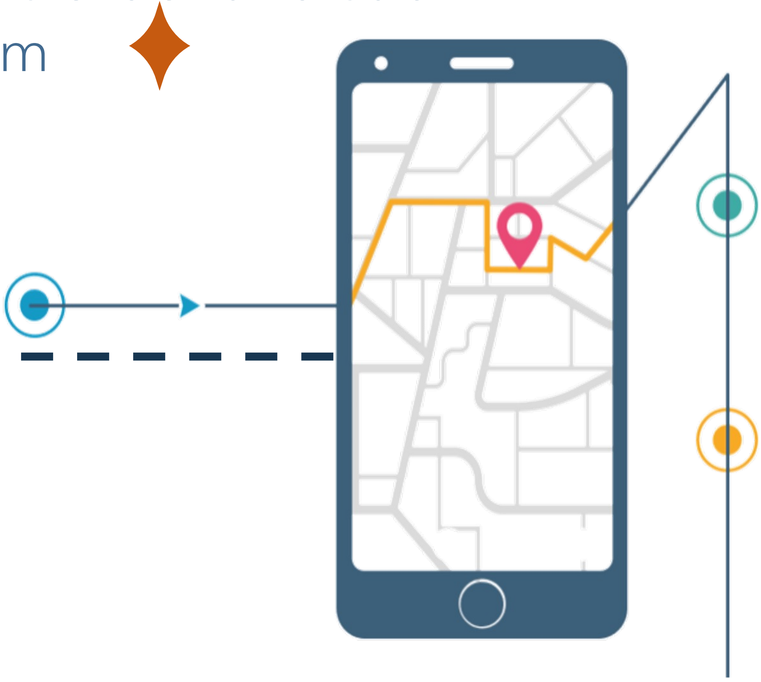
Order delivery system

Instructor : Manal Alharbi
class 4

Introduction to the project :

The main objective of the project :
programming a delivery system that serves the user and provides him with many good drivers to deliver their orders, with the addition of many special delivery offers

A simplified idea of our delivery system:
A simplified idea of our delivery system: a system that allows the user to enter his personal Information and his location data , then displays many stores for him to choose from, then he chooses what he wants from the orders and they are saved for him in the cart, he will be shown many drivers present in his area to deliver his order and submit offers especially for him



classes of the system

Class for storing user data

Class Name : Customer -------> to store his personal information

Class Name : location -------> to store his location information

Class for storing stores and their products

Class Name : Store

Class Name : Item

Classes for orders and the user's product cart

Class Name : OrderService

Class Name : ShoppingCart

Class Name : DATETIME

Class Name : LinkedList

Classes for store all drivers and their information

Class Name : DriverInformation

Class Name : ListOfInformation

Classes for storing order invoice and discounts on it

Class Name : Invoice
+class Arwa

Data structures used in the system

Data Structures Name	Implemented within a class	The reason for its choice - the purpose of its implementation
Array	Store Stores in an array	• To be able to do search method in array more easily and smoothly. • The complexity time for the search method in array will be less compared to the search method in list.
	Store Products for each store in an array	• To separate each product and consider it as an “item” with certain characteristics. • For easy access to the characteristics of a particular product.
	Store Locations for each store in an array	• To be able to do search method in array more easily and quickly.

Data structures used in the system

Data Structures Name	Implemented within a class	The reason for its choice - the purpose of its implementation
Array	Shopping cart	to speed up the process of accessing and displaying information, as the array is faster in the process of retrieving and accessing information than linked list
	Invoice	To take the product of the customer's basket, display it in the invoice, and calculate its price, taking into account the quantity of the product, Because if it reaches this point, all products have a fixed size.

Data structures used in the system

Data Structures Name	Implemented within a class	The reason for its choice - the purpose of its implementation
Linked List	Store the price of the products in a singly linked list.	• To be able to do sort method in list more easily and smoothly. • Time complixty is good.
	Store the storted stores in a singly linked list.	• Use “Linked List” instead of “array” to avoid booking positions without using it or scrolling to it. • To be able to add the element at the beginning or end of the list.
	Shopping Cart	because most of the cart’s operations are adding and deleting, I used linked list for all operations of dealing with the cart, because it is faster in the process of adding and deleting.

Data structures used in the system

Data Structures Name	Implemented within a class	The reason for its choice - the purpose of its implementation
Tree	Report	each object of OrderService class will be stored inside tree structure , because tree is the best data structure for dealing with big data.
Stack		

Algorithms used in the system

Algorithm name	Executed on the data Structure	complexity time	Big-O
sortList()	singly linked list<Node>	consnant	F(n) = O(1)
sortStores(Store[] store, String userArea)	singly linked list<Store>	consnant	F(n) = O(1)
sortProduct(Store[] store)	singly linked list<double>	consnant	F(n) = O(1)
searchExactLocation(String userArea, Store store)	array	linear	F(n) = O(n)
BinarySearch()	Array	Log n	F(n)=O(log n)
Merge Sort	Array	Nlog n	F(n)=O(nlog n)
indexOf(E e)	LinkedList	linear	F(n)= O(n)

Algorithms used in the system

Algorithm name	Executed on the data Structure	complexity time	Big-O
Search(E e)	LinkedList	linear	F(n)=O(n)
RemoveFirst()	LnkedList	constant	F(n)=O(1)
AddFirst(E e)	LinkedList	Constant	F(n)=O(1)
AddLast(E e)	Linked List	Constant	F(n)=O(1)
Clear()	LinkedList	linear	F(n)=O(n)
AddNode(E e)	BinaryTree	Log n	F(n)=O(log n)
searchNode (E e)	BinaryTree	Log n	F(n)=O(log n)

Algorithms used in the system

Algorithm name	Executed on the data Structure	complexity time	Big-O
deleteNode (E e)	BinaryTree	Log n	F(n)=O(log n)

Project benefits for us

○ Think well before choosing the appropriate algorithm

○ Reducing the time it takes to implement the program

○ Being able to deal with more than one data structure

○ Distinguish between types of data structures in an applied way

○ The ability to implement one algorithm on more than one data structure

○ Increase team collaboration to produce the best delivery system

The distribution of the work

Murooj Al-zahrani

ID : 443008435

Lamis Mohammed

ID : 443007788

I implemented tree data structure, linked list, Array, and the most important algorithms in it, and used them on the orderservice class, and in the report class, the invoice class, the shopping cart class

The distribution of the work

Arwa Asiri

ID : 443008129

Manar Al-mashi

ID : 443008492

The distribution of the work

Fatme Bin Bisher

ID : 443012084

Detailed explanation

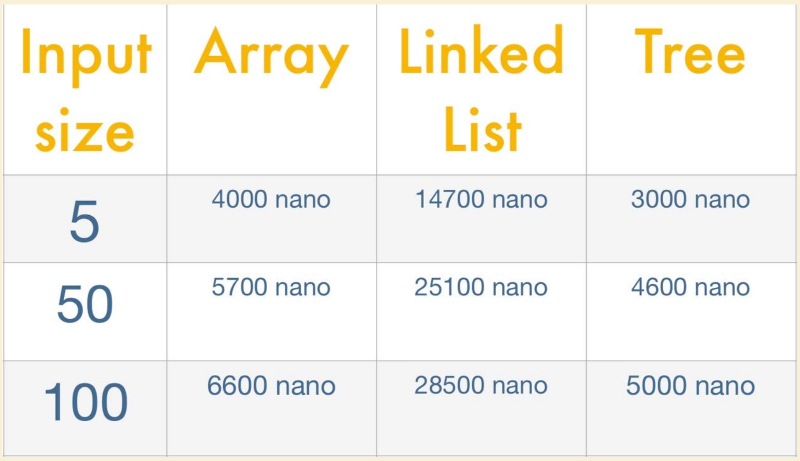
Lamis Mohammed

OrderService class, this class is for all order information, user information, time and date of the order, the cart, and the order number. each object of this class will be stored inside tree structure ,
“Tree structure provides easier and quicker access to data. While linear data structures store data sequentially, tree structures permit data access in different directions. As data sizes increase, access in such structures becomes slower, which can be hugely problematic in today's hyper-digital world. Tree structures can avoid such issues.” (TeachTarget) .
This is why I created a binary tree class, and modified it to fit its purpose. so that each element in the tree is an "order", and it sorts and accesses orders according to the order number.
ShoppingCart class, this class deals with the cart, filling it in, deleting items from it, searching for an item in it, and displaying items. And because most of the cart’s operations are adding and deleting, I used linked list for all operations of dealing with the cart, because it is faster in the process of adding and deleting. upon completion of filling the cart, the linked list will be converted into array, to speed up the process of accessing and displaying information, as the array is faster in the process of retrieving and accessing information than linked list

Detailed explanation

DateTime class, this class is for dealing with the time and date of the request. once the request is confirmed, the time and date will be saved at the same moment, and it will be stored in this class.
Invoice class, this category is concerned with everything related to invoice data and operations. takes order information, including the customers name and phone number. he takes the products and shows them their quantities and the price of each product. then you calculate the price of the existing products, by multiplying the price of each product by the quantity ordered and adding it with the rest of the products. my colleague Arwa added a discount to the invoice products and it will also appear on the invoice.
Item class, This class is for the Items, the item information such as the name, code, price, and the quantity that the user wants from it
Report class, This class I wrote with the help of my colleagues, and the purpose of it at the end of the program will display some of the statistics that occurred in the program.

Experimental Studies of search Algorithm on different data structure:



Input size	Array	Linked List	Tree
5	4000 nano	14700 nano	3000 nano
50	5700 nano	25100 nano	4600 nano
100	6600 nano	28500 nano	5000 nano

Detailed explanation

discount class

This class contains several methods in which each method tests a condition that determines whether the customer will get a discount or not, alongside a method that determines the delivery price depending on the location of the customer and the store.

This class is related directly to the Invoice class and Store class, where the value of the discount and delivery price is going to be implicitly printed within the printInvoice() method.

searchExactLocation() is a method from the Store class that returns a boolean, which was needed to determine the delivery price by testing if the store and customer are in the exact same location.

The linear search algorithm was executed twice, with Item Products[] in quantityDiscount() method, and with String daysList[] in dayDiscount() method, to test the condition for every element in the array.

Detailed explanation

Detailed explanation

Detailed explanation

working team

Murooj Al-zahrani

ID : 443008435

Fatme Bin Bisher

ID : 443012084

Manar Al-mashi

ID : 443008492

Arwa Asiri

ID : 443008129

Lamis Mohammed

ID : 443007788