University of Missouri – Kansas City

Project 1

Jasmine Naraine

Computer Science 303 – Data Structure

Ahmed Alkinani

April 23, 2021

Introduction

In project 1, number 3, we are creating a program for a client that is capable of using the Stack data type for a bank teller. In the program, it has to have to decide that the

* last customer to arrive will always be the first to be served
* create a class that will show the transaction of the customer.
* the customer can see their current balance and see their transaction, like to deposit or withdrawal money.
* show the transaction of five customers and the names of the five customers who will be served.

Implementation

This program will be using stack method and will also use string. To start off the project, first thing it will need are two classes. The first class will be the transaction. For a transaction it will need to know the amount of money the customer will deposit or withdrawal. There was no need to make anything private in the class. The program worked out fine with it being public. The customers amount and balance was public in the class.

For the second class, it will be the customers. The program will need the customer’s name, balance and transaction. As my first class, there was no need to put anything private in the class. Everything was made public, including the customer’s name, balance, and transaction. In the customer class, the customers will be able to deposit or withdrawal their money. If a customer doesn’t have enough money to withdrawal a certain amount, then the money will be failed to withdrawal. It will also show that if a customer wants to deposit money into their account, then it will show the customer the amount of money they will have in their account once they deposit their money.

As shown in the UML Diagram at the bottom, the two classes are public because of the “+” is public. The transaction is aggregation because the transaction class is not a strong class on it own.

UML DiagramDiagram

Description automatically generated

In the main program, there are five customers. Each customer has the ability to deposit and withdrawal their money. If the customer does not have enough money to withdrawal, then that customer will be failed to withdrawal their money. By using the Stack, push method, the customer that came in last will be the first to be taken care of. In the code, sine Klay is last name as the customer, then he will be the first to taken care of. After Klay then it is Luka, and so on after that. After all of the customers are severed, then it will show who will be processed next.

Big-O Notation

Since the Stack is using the push method, the Big-O notation is **O(1)** because it is a simply push method. The method didn’t involve any difficult mathematic for the Big-O notation to be something else.

Reference:

E, Sadique Ali. “UML Class Diagram Explained With C++ Samples.” *CPP Code Tips*, 19 July 2018, [www.cppcodetips.wordpress.com/2013/12/23/uml-class-diagram-explained-with-c-samples/](http://www.cppcodetips.wordpress.com/2013/12/23/uml-class-diagram-explained-with-c-samples/).

“Stack Push() and Pop() in C++ STL.” *GeeksforGeeks*, 19 Sept. 2018, [www.geeksforgeeks.org/stack-push-and-pop-in-c-stl/](http://www.geeksforgeeks.org/stack-push-and-pop-in-c-stl/).