I enumerated Customer ids according to input file order.

Structures
BankAccount: represents a bank account for a company. It has fields for the company name, balance, and mutex.
TicketVendingMachine: represents a ticket vending machine. It has fields for the thread, mutex, in-process status, customer ID, and ID.
Customer: represents a customer. It has fields for the thread, ID, sleep time, ticket vending machine ID, bank account ID, and payment amount.

Functions
output_func: a thread that writes to a log file any output string that is not empty.
vmachine_func: a thread that represents a ticket vending machine. It waits until a customer arrives, and when it does, it takes the customer's prepayment amount and adds it to the balance of the customer's bank account, and updates the in-process status to indicate that the machine is no longer in use.
customer_func: a thread that represents a customer. It sleeps for a certain duration, then goes to a ticket vending machine and prepays for a ticket.
main: creates and initializes the bank accounts, ticket vending machines, and customers, and creates and joins the threads.

Global Variables
accounts: an array of BankAccount structures, representing the bank accounts.
vmachines: an array of TicketVendingMachine structures, representing the ticket vending machines.
customers: an array of Customer structures, representing the customers.
finish: an integer representing whether the ticket vending machine thread should continue running or not.
finish_out: an integer representing whether the output thread should continue running or not.
outfile: a string representing the file to log output to.
output: a string representing the current output to be logged.

NUM_COMPANIES: an integer representing the number of companies.
COMPANY_NAMES: an array of strings representing the names of the companies.
NUM_VMACHINES: an integer representing the number of ticket vending machines.

Main Function
Initialize the bank accounts:
- Create an array of BankAccount structures, representing the bank accounts.
- Initialize the company name and balance fields for each bank account using the COMPANY_NAMES array.
- Initialize the mutex field for each bank account using the pthread_mutex_init function.
Initialize the ticket vending machines:
- Create an array of TicketVendingMachine structures, representing the ticket vending machines.
- Initialize the in_process and id fields for each ticket vending machine.
- Initialize the mutex field for each ticket vending machine using the pthread_mutex_init function.
Initialize the customers:
- Create an array of Customer structures, representing the customers.
- Initialize the fields for each customer using input values.
Create the output thread:
- Create a new thread using the pthread_create function and the output_func function.

Create a new thread for each ticket vending machine using the pthread_create function and the vmachine_func function.
Create a new thread for each customer using the pthread_create function and the customer_func function.
Join each customer thread using the pthread_join function.
Join each ticket vending machine thread using the pthread_join function.
Join the output thread using the pthread_join function.

Output Function (output_func())
Open the log file specified by the outfile global variable using an ofstream object.
While the finish_out global variable is greater than 0:
If the output global variable is not empty:
- Write the output global variable to the log file.
- Clear the output global variable.
- Unlock the output_mutex global mutex.
Close the log file.
Exit the thread.

Ticket Vending Machine Function (vmachine_func()) :
Get the ID of the ticket vending machine from the function parameter.
Initialize amount local variables.
While the finish global variable is greater than 0 or the in_process field of the ticket vending machine is 1:
If the in_process field of the ticket vending machine is 1:
Get the customer ID of the customer using the ticket vending machine from

the customer_id field of the ticket vending machine.
Get the customer structure for the customer using the customer ID.
Get the bank account ID and prepayment amount for the customer.
Lock the mutex of the customer's bank account using the pthread_mutex_lock function.
Add the prepayment amount to the balance of the customer's bank account.
Unlock the mutex of the customer's bank account using the pthread_mutex_unlock function.
Set the in_process field of the ticket vending machine to 0.
Lock the output_mutex.
Set the output variable to a string containing the customer ID, prepayment amount, and company name.
Unlock the mutex of the ticket vending machine using the pthread_mutex_unlock function.

Exit the thread.

Customer Function (customer_func())
Get the ID of the customer from the function parameter.
Get the customer structure for the customer using the customer ID.
Sleep for the duration specified by the sleep_time field of the customer using the this_thread::sleep_for function.
Get the ID of the ticket vending machine to use from the tvm_id field of the customer.
Lock the mutex of the ticket vending machine using the pthread_mutex_lock function.
Set the customer_id field of the ticket vending machine to the customer ID.
Set the in_process field of the ticket vending machine to 1.
Unlock the mutex of the ticket vending machine using the pthread_mutex_unlock function.
Exit the thread.