

## Semaphore

TellerSem      2

LoanSem        1

<Below are metrixs>

LoanInLine     []0

LeavesLoan     []0

TellerInLine   []0

LeavesTeller   []0

Trans           []0

## Thread

Cust            5

Teller          2

Loaner          1

Int Account    5 of

Int Loan        5 of

Int Max\_Visits 15 max

**Linked\_List      item saved (#of customer<amount> todo)**

LoanLine       let Loan officer know the request and customer

TellerLine      let teller know the request and customer

TellerDone      let customer know who did the job

Item saved(# teller/loan officer)

LoanWindow

TellerWindow

## Function

Void customer

{

    Cust    #of tag

    Int amount = rand(1,5)

    amount = amount\*100

    Int todo = rand(1,3)

    For(int=3)

        if todo ==1 // loan

            Loan\_line.add([#,amount,todo])              //get in loan line

            print (Customer [#] is in Loan line ready for [amount] [Loan] )

            signal (LoanInLine)                              //let loaner know "imhere"

            print (Customer now waits for and available Loan Officer)

            wait (LoanSem)                                      // wait for available loaner

            print (Customer now waiting for Loan Office to finish request)

            signal(LeavesLoan)                              //leaves

            print (Customer has left)

```

        Max_visits++
    if todo == 2// deposit
        Teller_line.add([#,amount,todo])    //get in teller line
        print (Customer [#] is in Teller line ready for [amount] [Deposit])
        signal (TellerInLine)                //let Teller know "imhere"
        print (Customer [#] now waits for and available Teller)
        wait (TellerSem)                      //wait for available teller
        print (Customer now waiting for Teller to finish)
        signal(LeavesTeller)                  // leaves
        print (Customer has left)
        Max_visits++
    if todo == 3// Withdrawal
        Teller_line.add([#,amount,todo])    //get in teller line
        print (Customer [#] is in Teller line ready for [amount] [Withdrawal])
        signal (TellerInLine)                //let Teller know "imhere"
        print (Customer [#] now waits for and available Teller)
        wait (TellerSem)                      //wait for available teller
        print (Customer now waiting for Teller to finish)
        signal(LeavesTeller)                  // leaves
        print (Customer has left)
        Max_visits++
}

Void Loan officer ()
{
    print (Loan officer created)
    while (true)
        wait(LoanInLine)                    // Loaner waits for Customer
        Loan_line.pop([#,amount,todo])      // gets customer from queue line
        print(Loan Officer now serving Customer [#])
        print(Approves a loan of [amount])
        Loan# = Loan# + amount              //applys loan amount
        Signal (TellerSem)
        print (Loan Officer has finished serving customer [#])
        wait(LeavesLoan)
        print (Loan Officer now ready to serve new customer)
        if (Max_Visits == 15)
            BreaksAll() //function to end the program
}

Void Teller()
{
    print (Teller created)
    int teller #
    while(true)

```

```

wait(TellerInLine)
print(Teller [#] now serving Customer [#])
print(Approves a loan of [amount])
Teller_Line.pop([#,amount,todo])
print(Teller [#] now serving Customer [#])
signal(TellerSem)
if todo==2 // deposit
    account# = account# + amount
    print(Teller # approves a deposit of [amount])
if todo==3 // withdrawal
    account# = account# - amount
    print(Teller # approves a withdrawal of [amount])
print(Teller [#] finished serving customer [#])
wait(Leaves_Teller)
print(Teller [#] now ready to serve new customer)
if (Max_Visits == 15)
    BreaksAll() //function to end the program
}

Main
{
    Customer_tread(5)
    Teller(2)
    Loaner(1)

}
BreaksAll
{
    daemon to Teller&Customer
}

```

Customer	Teller	Loan Officer
<b>Task: Deposit</b> Queue in teller line Request deposit Get Receipt	Process deposit	[x]
<b>Task: Withdrawal</b> Queue in teller line Request withdrawal Get Receipt	Process withdrawal	[x]

<b>Task: Loan</b> Queue in loan line Request loan Get Receipt	[x]	Process loan
--	-----	--------------