SYSTEM PROGRAMMING ASSIGNMENT

1 i) # prompt user for employee details

read -p "Enter employee name: "name

read -p "Enter hours worked: "hours\_worked

read -p "enter rate per hour." rate\_per\_hour

#calculate basic pay

basic\_pay=$(echo "Shours\_worked $rate\_per\_hour" | bc)

#calculate tax

if (($(echo "Sbasic\_pay > 70000" | bc-1))); then

tax=$(echo "0.25 \*Sbasic\_pay" | bc)

elif(($(echo "Sbasic\_pay >=15000 && $basic\_pay <= 70000" |bc-1))); then tax=$(echo "0.15 Sbasic\_pay" | bc) \*

else

tax=0

be

#display results

echo "Employee name: $name"

echo "Basic pay: $basic\_pay"

echo "tax: $tax"

ii) # calculate basic pay

basic\_pay=$(echo "$hours\_worked \*$rate\_per\_hour" | bc)

#Display basic pay

echo "Employee name: $name"

echo "Basic pay: $basic\_pay"

iii) #function to calculate tax

calculate\_tax() {

basic\_pay==$1

if [ $basic\_pay-gt 70000 ]; then tax=$(echo "0.25\* $basic\_pay" | bc)

elif [ $basic\_pay -ge 15000 ]; then tax=$(echo "0.15 \*$basic\_pay" | bc)

else

tax=0

be

echo $tax

}

iv) #calculate net pay

net\_pay=$(echo "$basic\_pay $tax" | bc)

2.

#include <stdio.h> // include standard i/o library for file operations

inrt main() {

FILE \*file\_ptr; //declare a pointer to a file object

if (file\_ptr == NULL) {

printf("Error opening file!\n");

exit(1); //exit the program with an error code

}

// write "Hello World"to the file

fprintf(file\_ptr, "hello world\n");

//close the file

fclose(file\_ptr);

//open the file again, this time in read mode

file\_ptr=fopen("example.txt", "r");

return 0; //exit the program successively

}

3. i) #Prompt the user to enter CustomerID, CustomerName, and UnitConsumed

read -p "Enter Customer ID: " customer\_id

read -p "Enter Customer Name: "customer\_name

read -p "Enter Units Consumed: "units\_consumed

# Display the entered information

echo "Customer ID: $customer\_id"

echo "Customer Name: Scustomer\_name"

echo "Units Consumed: $units\_consumed"

ii) #Prompt the user to enter CustomerID, CustomerName, and UnitConsumed

read -p "Enter Customer ID: " customer\_id

read -p "Enter Customer Name: "customer\_name

read -p "Enter Units Consumed: " units\_consumed

# Initialize the bill amount

bill\_amount=0

# Calculate the bill amount based on the units consumed

if [ $units\_consumed -It 200 ]; then bill\_amount=$(($units\_consumed \* 120))

elif [ $units\_consumed -ge 200 ] && [ $units\_consumed -lt 400 ]; then bill\_amount=$(($units\_consumed \* 150))

elif [ $units\_consumed -ge 400] && [ $units\_consumed -lt 600 ]; then bill\_amount=$(($units\_consumed \* 180))

else

bill\_amount=$(($units\_consumed \* 200 ))

be

# Display the bill amount

echo "Customer ID: $customer\_id"

echo "Customer Name: $customer\_name"

echo "Units Consumed: $units\_consumed"

echo "Electricity Bill: Ksh $bill\_amount"

iii) # Function to calculate the total bill amount

calculate\_bill() {

local units\_consumed=$1

local rate=0

# Determine the rate based on the units consumed

if [ $units\_consumed -It 200 ]; then

rate=120

elif [ $units\_consumed -ge 200 ] && [ $units\_consumed -It 400 ]; then rate=150

elif [ $units\_consumed -ge 400] && [ $units\_consumed -It 600 ]; then

rate=180

else

rate=200

be

# Calculate the total bill amount

total\_bill=$(($units\_consumed \* $rate ))

echo $total\_bill

}

# Main program

read -p "Enter Customer ID: " customer\_id

read -p "Enter Customer Name: " customer\_name

read -p "Enter Units Consumed: " units\_consumed

# Calculate total bill

total\_bill=$(calculate\_bill Sunits\_consumed)

# Display the total bill amount

echo "Customer ID: Scustomer\_id"

echo "Customer Name: Scustomer\_name"

echo "Units Consumed: $units\_consumed"

echo "Total Bill Amount: Ksh $total\_bill"