

Caribbean Maritime University

Faculty of Engineering& Applied Technology

B.Eng. In Mechatronics

Computer Programming

Final Project

Names and ID Number: Vaughn Davis 20190952

Alexander Pinnock

Leonard Smith

Troy Hamilton 20190886

Group Names: Mechatronics Stream2 Year 2

**Table of Content**

1. Abstract
2. General Terms
3. Introduction
4. Designing Function
5. Algorithm
6. Output From Test Cases
7. Conclusion & Future Scope
8. References

**Abstract**

Nowadays, people are racing to achieve a comfortable standard of living. They work hard to support their families and also to the satisfaction of his own life. It is not surprising if we see many who can afford to buy houses and school with full payment for the common interest. However not all are able to do so, those who have large families are more likely to pay in monthly installments. For example for loan calculator. Therefore our group has developed an application that will facilitate their affairs that cannot afford for the full installment for the anything. We developed an application called “Loan Calculator” to reduce their problem. The purpose of this application is to help people who want to make purchases by using monthly payments. With this application also, people will find out if they can afford school, getting a dream house etc by using this user friendly loan calculator. The user just need to insert total amount of needed, make down payments, interest rate and loan period then the application will calculate to the user how much user need to pay monthly within a certain period of time.

**Introduction**

When you are planning a purchase, but you realize that you cannot afford it from your savings or income (alternatively you are about to take an [investment](https://www.omnicalculator.com/finance/investment) decision requiring financial support), the most straightforward way to obtain the necessary money is to turn to a bank for a loan. If you are in such a situation, probably one of the first things that comes to your mind is whether you will be able to make the required payments (or in other word, installments) from your income. To solve this puzzle, you need to answer the crucial question: what will my loan payment be? With our **loan payment calculator**, **you can quickly compute the amount of money you need to devote for loan repayment over a payment period**. This device will also give you the total amount you need to pay back during the whole loan term. Alternatively, you can use it to compute the loan amount from the loan payments.

**Designing Function**

The program only contains two(2) functions:

I. amortize\_table() - The purpose of this function is to generate an amortization table based on the loan information that was entered by the user.

The function takes three(3) arguments:

a. amount: This is the loan amount borrowed

b. rate: The monthly interest rate

c. payment\_years: This is the loan duration in years

II. menu\_check() - This functions’ purpose is to ensure that the user is only able to enter a valid response option from the menu list that is given at the start of the program.

The function requires no arguments.

**Algorithm**

* Variable initializations

**Psuedocode**

# Declaring and initalizing variables

loan\_amount = 0.0

interest\_rate = 0.0

loan\_duration = 0.0

monthly\_payment = 0.0

total\_loan\_repaid = 0.0

compound\_interest = 0.0

number\_of\_payments = 0.0

def amortize\_table(amount, rate, payment\_years):

# Declaring and Initializing function variables

num\_of\_months = 0

total\_payments = 0.0

interest = 0.0

principal = 0.0

total\_payments = payment\_years\*12

payment = (amount\*(rate\*((1 + rate)\*\*total\_payments))) / ((1 + rate)\*\*total\_payments)-1)

table\_headers = ['PAYMENT NO.', 'LOAN VALUE', 'INTEREST', 'PRINCIPAL', 'PAYMENT', 'REMAINING BALANCE’]

while amount > 0:

num\_of\_months = num\_of\_months + 1

interest = amount\*rate

principal = payment - interest

if (amount - payment) < 0:

principal = amount

table\_rows = [num\_of\_months, amount, interest, principal, payment, (amount-principal)]

amount = amount - principal

print(table)

def menu\_check():

# Declaring and initalizing function variables

user\_input = 0

while True:

try:

user\_input = int(input('Pick a number from list'))

except ValueError:

print('Choose a valid response')

continue

if user\_input <= 0 or user\_input > 4:

print('Valid responses only')

continue

else:

return user\_input

while True:

#Menu System

print("Choose a valid response from the list given below: ")

print('1. Exits the program')

print('2. Enter loan information')

print('3. View amortization table based on the entered loan information')

print('4. View amortization table based on the entered loan information and automatically exits the program')

choice = menu\_check()

if choice == 1:

break

elif choice == 2:

while True:

try:

# Gathering loan information from the user

loan\_amount = float(input('Enter the amount being loaned'))

interest\_rate = float(input('Enter the interest rate amount'))/100

interest\_rate = interest\_rate/12

loan\_duration = int(input('Enter the number of years to repay the loan'))

number\_of\_payments = loan\_duration\*12

except ValueError:

print('ERROR: Please choose a valid response')

continue

# Calculation Section

monthly\_payment = (loan\_amount \* (interest\_rate \* (1 + interest\_rate)\*\*number\_of\_payments)) \

/ (1 + interest\_rate)\*\*number\_of\_payments) - 1))

total\_loan\_repaid = monthly\_payment\*loan\_duration\*12

compound\_interest = loan\_amount \* (1 + ((interest\_rate \* 12)/ 1)) \*\* (1 \* loan\_duration)

# Displaying the information

print(f"For the loan amount ${loan\_amount}.")

print(f"The total amount to be repaid is ${total\_loan\_repaid}.")

print(f"The total amount to be paid back monthly is ${monthly\_payment}.")

print(f"The compound interest is ${compound\_interest}")

break

elif choice == 3:

amortize\_table(loan\_amount, interest\_rate, loan\_duration)

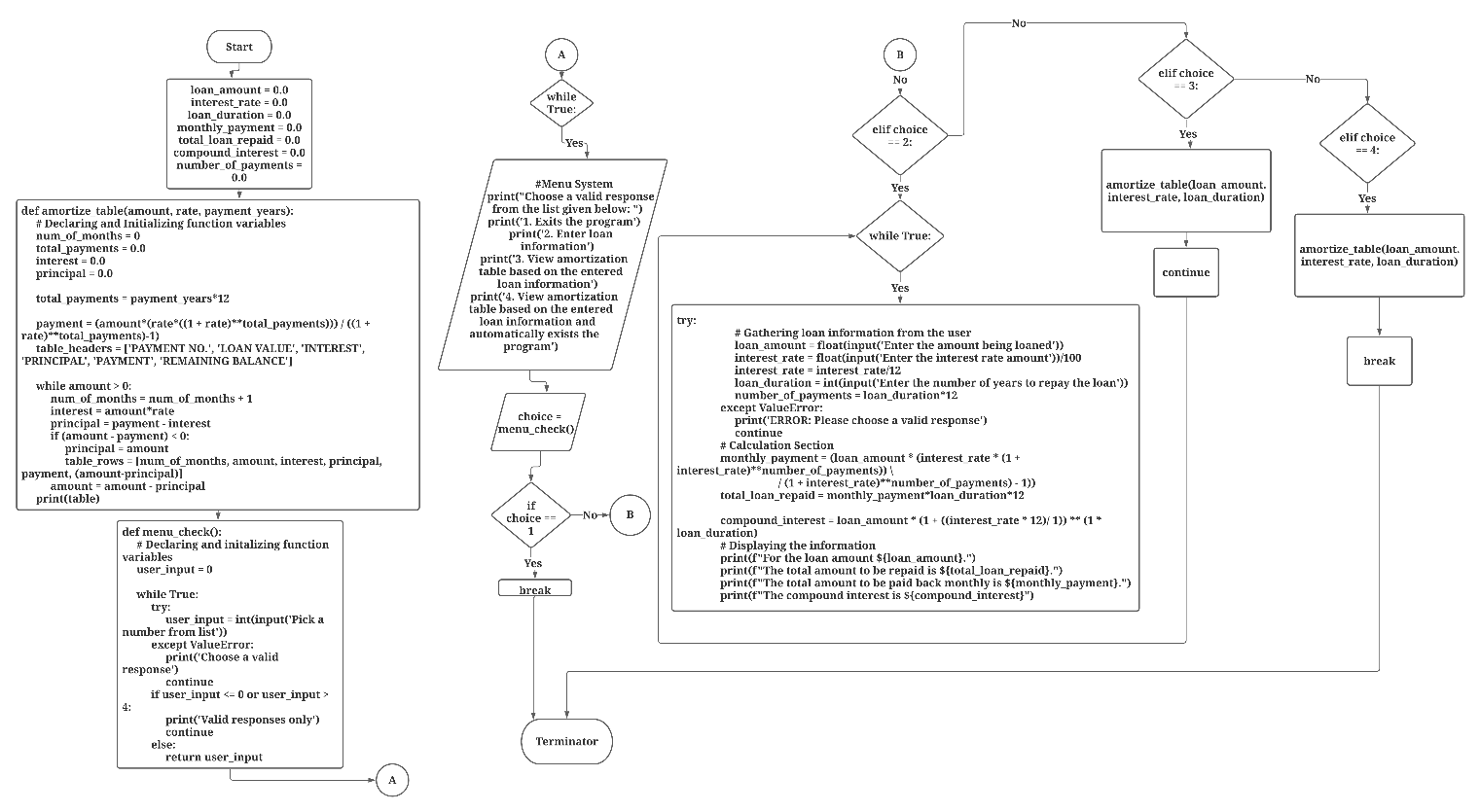
continue

elif choice == 4:

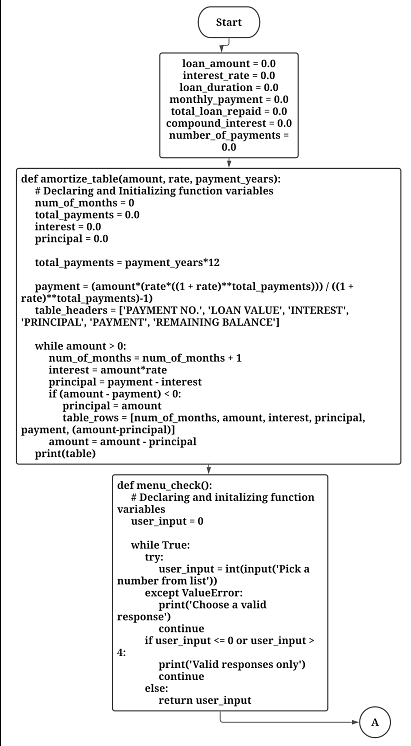
amortize\_table(loan\_amount, interest\_rate, loan\_duration)

break

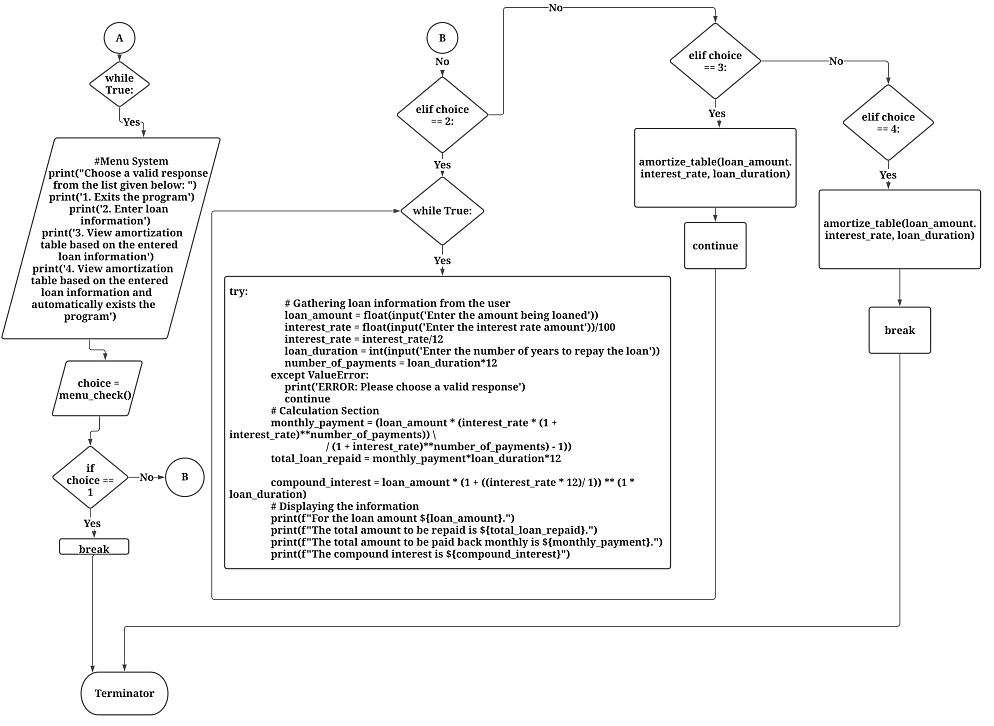
**Flow Chart**



*Figure* 1: Diagram showing overview of the entire flowchart.



*Figure* 2: Diagram showing the first section of the flowchart (the left side of the overview).

  
*Figure* 3: Diagram showing second and third sections of the flowchart (the portion in the middle and to the right).

**Output From Test Cases**

**Conclusion & Future Scope**

**References**