



Universität St.Gallen

**Mortgages Calculator & Comparator
project using Python**

-

Programming with Advanced Computer Languages

-

Scott Lee Waclaw BRODA
18-419-077

Table of Contents

<i>Ideation of the project.....</i>	<i>3</i>
<i>Overview of the project</i>	<i>3</i>
<i>Overview of the three types of mortgages.....</i>	<i>3</i>
<i>Input needed from the user.....</i>	<i>4</i>
<i>Project Dependencies.....</i>	<i>4</i>
<i>How the model works.....</i>	<i>5</i>
<i>Input – Example.....</i>	<i>6</i>
<i>Output of the code.....</i>	<i>6</i>
<i>Observation and learning outcomes.....</i>	<i>7</i>

Ideation of the project

As a master's student in Accounting & Corporate Finance, I wanted my project to be finance-related. Therefore, I decided to use the knowledge acquired during my Real Estate Finance class with the programming skills I gained on CodingXCamp. Regarding the choice of programming languages, I decided to use Python for two main reasons. First, to apply and develop my programming skills using this language, and secondly, because I am proficient in R, I wanted to explore potential challenges and differences when implementing such a project in Python in comparison to R.

Overview of the project

The objective of this project is to aid users in understanding and comparing different mortgage options. The user is prompted multiple times to input the necessary information for calculating each type of mortgage. Based on their input, the project outputs the three most common mortgage prices along with a visual representation of the mortgage payment and outstanding loan balance over the mortgage's duration.

Overview of the three types of mortgages

Constant Payment Mortgages – CPM :

CPM is perhaps one of the most common types of mortgages. As the name implies, the borrower makes fixed monthly payments throughout the loan duration. The monthly payment amount remains constant ($\text{payment} = \text{principal} + \text{interest}$). As the loan progresses, the proportion of the payment allocated to interest decreases, while the portion allocated to the principal increases.

Constant Amortizing Mortgages – CAM :

CAM is a type of mortgage in which the principal payment remains constant throughout the loan term. The borrower pays a fixed amount of principal each month, plus the interest payment. The monthly payment decreases over the loan period, as the interest payment decrease, and the principal payment remain constant.

Adjustable-Rate Mortgage – ARM :

An Adjustable-Rate Mortgage (ARM) is a type of mortgage with an interest rate that can fluctuate over time. Typically, the ARM begins with an initial fixed rate period, during which the interest rate remains constant. After the initial period, the interest rate adjusts periodically based on changes in a specified financial index. As a result, the monthly payment amount can change, causing the loan balance and repayment schedule to adjust accordingly.

For simplification of the project, we will include only one interest rate adjustment

Input needed from the user

1. **First prompt asking if you want to calculate :** Yes or No
2. **Principal Amount (\$):** The initial loan amount or principal.
3. **Annual Interest Rate (%):** The annual interest rate charged on the loan.
4. **Number of Years:** The duration of the loan in years.
5. **Maximum Monthly Payment (\$):** The maximum monthly payment the user can afford to see if the project is affordable (if not warning message printed).
6. **Initial Rate Period for ARM (in years):** The initial fixed rate period for the Adjustable-Rate Mortgage (ARM).
7. **Initial Interest Rate for ARM (%):** The initial interest rate for the Adjustable-Rate Mortgage (ARM).
8. **Full amortization :** asking if the user want to fully repay the loan at maturity or not
9. **If not :** how much of outstanding balance the user wants ?

Project Dependencies

This mortgage calculator project requires the following libraries to be installed:

- **NumPy:**
 - Installation Command: `pip install numpy`
- **Matplotlib:**
 - Installation Command: `pip install matplotlib`
- **Pandas:**
 - Installation Command: `pip install pandas`
- **numpy_financial:**
 - Installation Command: `pip install numpy-financial`

How the model works

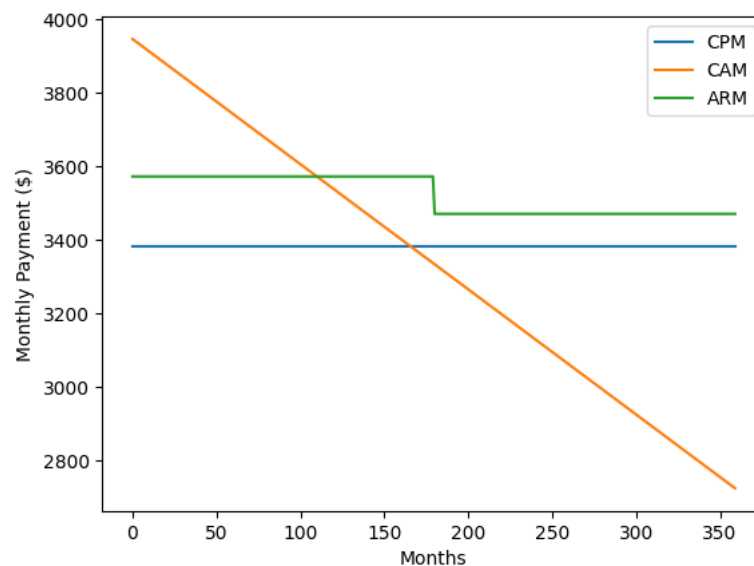
1. The code defines three functions to calculate payments and loan balances for each type of mortgage: `calculate_payments_cpm`, `calculate_payments_cam`, and `calculate_payments_arm`.
2. The `calculate_payments_cpm` function uses the provided principal, interest rate and loan duration, to calculate the monthly payment using the formula for a constant payment mortgage. It then calculates the outstanding loan balance for each month using a loop and stores the results in an array.
3. The `calculate_payments_cam` function calculates the monthly payment and outstanding loan balance for each month using the constant amortization method. It starts with the principal amount and iteratively calculates the principal and interest payments for each month.
4. The `calculate_payments_arm` function calculates the monthly payment and outstanding loan balance for each month for an adjustable rate mortgage. It calculates the initial monthly payment based on the initial rate period and interest rate, and then calculates the remaining loan balance and monthly payments for the remaining months.
5. Two plotting functions, `plot_payments` and `plot_loan_balance`, are defined to create visualizations of the mortgage payment and outstanding loan balance over the loan duration for the three mortgage types.
6. The `mortgage_calculator` function serves as the main entry point for the program. It prompts the user to input the necessary mortgage information, calls the respective calculation functions, and plots the results.
7. Finally, a loop allows the user to calculate multiple mortgages if desired, by repeating the mortgage calculation process based on user input.

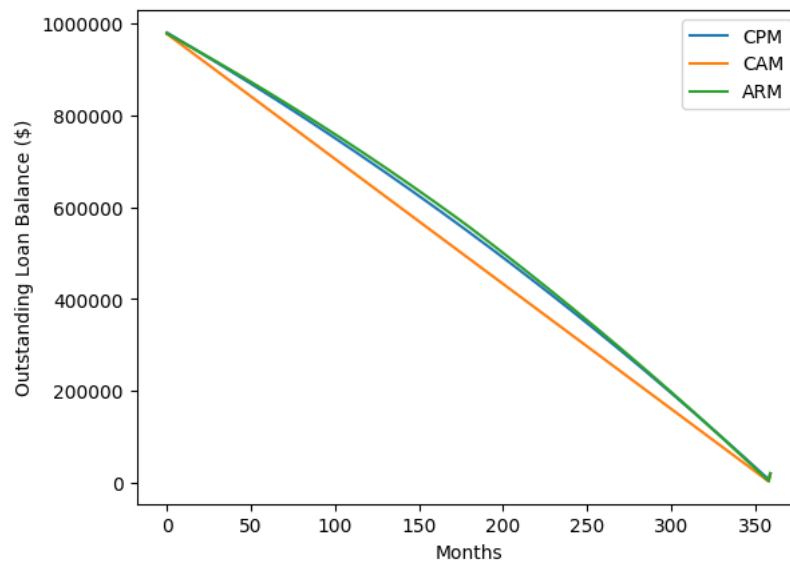
Input – Example

```
Do you want to calculate a mortgage? (yes/no): yes
Enter the principal amount ($): 1000000
Enter the annual interest rate (%): 1.5
Enter the number of years: 30
Enter the maximum monthly payment you can afford ($): 2000
Enter the initial rate period for ARM (in years): 15
Enter the initial interest rate for ARM (%): 1.9
Do you want to fully repay your loan at maturity? (yes/no): no
Enter the outstanding loan balance you want at maturity ($): 20000
```

Output of the code

Warning! The monthly payment for CPM is above your budget.
Warning! The monthly payment for CAM at some point goes above your budget.
Warning! The monthly payment for ARM at some point goes above your budget.





	Mortgage Type	Avg. Monthly Payment	Total Payment	Interest Paid
0	CPM	\$3,382.18	\$1,217,584.10	\$217,584.10
1	CAM	\$3,336.42	\$1,201,112.50	\$201,112.50
2	ARM	\$3,522.64	\$1,268,152.09	\$268,152.09

Do you want to calculate a mortgage? (yes/no): no

Observation and learning outcomes

After completion, the three main observations. The first one, perhaps the most frustrating one, was dealing with data visualization in Python. I found it much easier to visualize data in R, especially being used to the plethora of libraries design for data visualization. Secondly, trying to incorporate all the elements inherent to the different mortgages types was challenging as each new features impacted the overall code. Lastly, the difference in the syntax and working environment between R and Python was an issue at the beginning (as I had R “reflexes”). Despite these challenges, the project and the class refreshed my prior knowledge in Python, and gave me an incentive to keep practicing coding.