2.2

First let me define global parameters for this python program by the following code:

*annual\_risk\_free\_rate = 0.03*

*mortgage\_type = 10 # 10 years 20 years 30 years*

*default\_prob = 0.01*

*month = 12*

*no\_payment = mortgage\_type \* month*

*monthly\_payment = 1000 # you could change it based on your need*

you could fix the mortgage years among 10 ,20 and 30 by changing **mortgage\_type** parameter and monthly payment of the mortgage by changing **monthly\_payment** parameter.

Second, I define a function called default to test whether there is a default to a mortgage.

*def default(): # the function to test whether default*

*if random.random() < 0.01:*

*return 1*

*else: return 0*

if there is a default, when random.random() is less than 0.02, return 1.

Third, I define a function called default\_NPV() to compute the NPV the insurance company have to pay for a default mortgage.

def default\_NPV(default\_period,monthly\_payment): # compute NPV of the mortgage that insurance company have to pay for a random default period!

*NPV = 0.0*

*t = default\_period*

*for i in range(t, no\_payment):*

*NPV = NPV + (monthly\_payment\*exp(-(annual\_risk\_free\_rate/month)\*t))*

*t = t+1*

*return NPV*

If in a random month the buyer is default, get the NPV that the insurance company has to pay, that is the function itself being called by the following function called simulate\_one\_person().

Then, build the simulate\_one\_person(), since it is an independent event for a buyer to be default every month with a possibility of 1%. I may want to simulate 12\*mortgage years times(e.g. if it is a 10-year mortgage, simulate 12\*10 = 120 times) of whether a buyer want to be default. If in a random month the buyer is default, then get the NPV that the insurance company has to pay by calling the default\_NPV() function, the simulate\_one\_person function goes like:

*def default\_one\_person(): # simulate NPV of how much a insurance should pay if one defaulted*

*t = 0*

*sum = 0.0*

*for i in range(no\_payment):*

*if default():*

*sum = default\_NPV(t,monthly\_payment)*

*break*

*else:*

*t = t + 1*

*return sum*

Finally, simulate a situation that the insurance company has so many clients and compute the average cost for those clients by :

*def simulate\_many( n ): # simulate a number of n of persons`s default and take the expected value for those n persons`s insurance cost for the insurance company*

*total = 0.0*

*for i in range(n):*

*total = total + default\_one\_person()*

*return total/n*

As a quick example, let the monthly payment be $1000 and insurance for 10 years, simulate a situation of 10000 persons and 100000 persons. We get the simulating result as following :

*>>> simulate\_many( 10000 )*

*42200.937069654414*

*>>> simulate\_many( 100000 )*

*41911.25743539597*

That means for a 10 -year mortgage with a monthly payment of $1000, the NPV for the insurance is around 42200 . monthly insurance cost for a 10-year mortgage is &407 if there will be no default.