```python
import numpy as np

salary = 48000
raise_mean = .027
raise_std = .004
age_current = 24
age_retire = 60
emp_contribution = .12
match = .5

return_wghts = { 'a': .5,
                 'b': .25,
                 'c': .25
                 }
return_means = { 'a': .0663,
                 'b': .0989,
                 'c': .0855
                 }
return_stds  = { 'a': .1346,
                 'b': .1528,
                 'c': .1690
                 }
fund_values  = { 'a': 0,
                 'b': 0,
                 'c': 0
                 }

salaries = []

def calc_contribution_return(fund_values, contribution, fund):
    contribution_part = return_wghts[fund] * contribution/12
    fund_return = np.random.normal(loc=return_means[fund], scale=return_stds[fund])
    print('\t\tfund {} return: {}'.format(fund, fund_return))
    annuitized = contribution_part*(((1 + (fund_return/12))**(12))-1)/(fund_return/12)
    return fund_values[fund]*(1+fund_return) + annuitized



for i in range(age_current, age_retire):
    print('age: {}'.format(i))
    year_contribution = salary * (emp_contribution * (1+match))

    for fund in fund_values.keys():
        fund_values[fund] = calc_contribution_return(fund_values, year_contribution,
        fund)


    raise_pct = np.random.normal(loc=raise_mean, scale=raise_std)
    salary = salary * (1+raise_pct)
    salaries.append(salary)

    print('\tsalary: {}\n\tcontribution: {}\n\tnew act value: {}'.format(salary,
    year_contribution, sum(fund_values.values())))


```