컴퓨터 알고리즘과 실습 1주차 보고서

2016112158 김희수

문제1번

<1-1> 유사코드 작성

accumulate\_average(array):

    sum = 0

    for i = 0 to Length(array)-1:

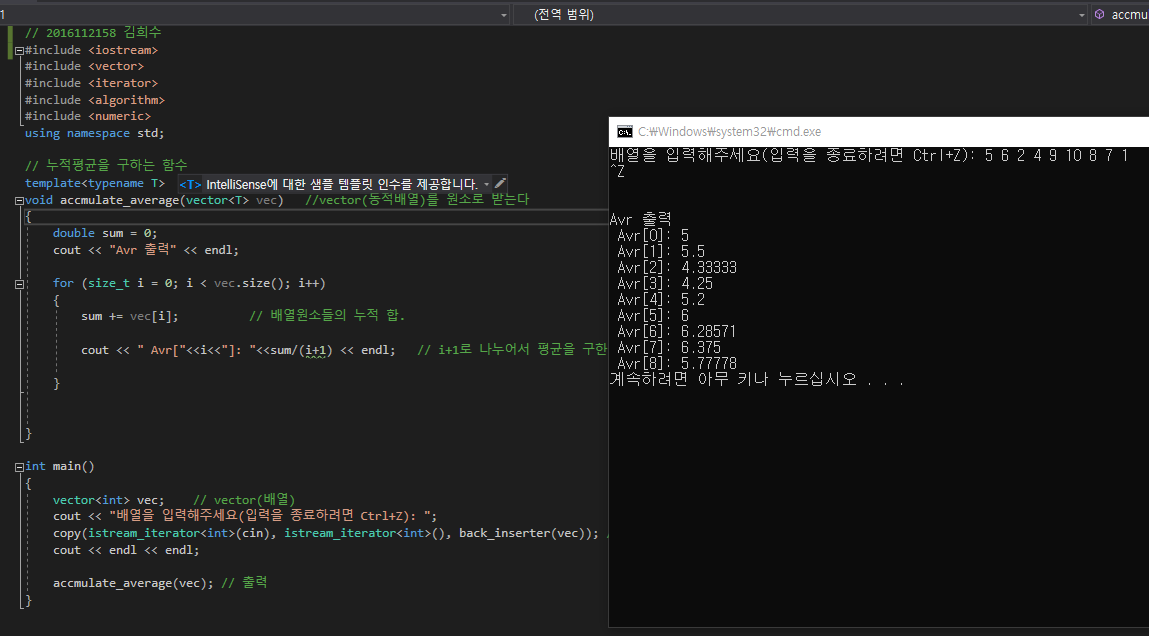
        sum+=array[i]

        print(sum/(i+1))

배열을 입력받아서 각 누적합을 구한 후, 누적합을 각 단계까지의 원소개수로 나누어주면 누적평균이 된다.

<1-2> 소스코드 첨부했음

<1-3실행화면>



문제2번

<2-1 GreedyBetterChange 유사코드>

GreedBetterChange(Money, coin, d):

    for i := d-1 to 1

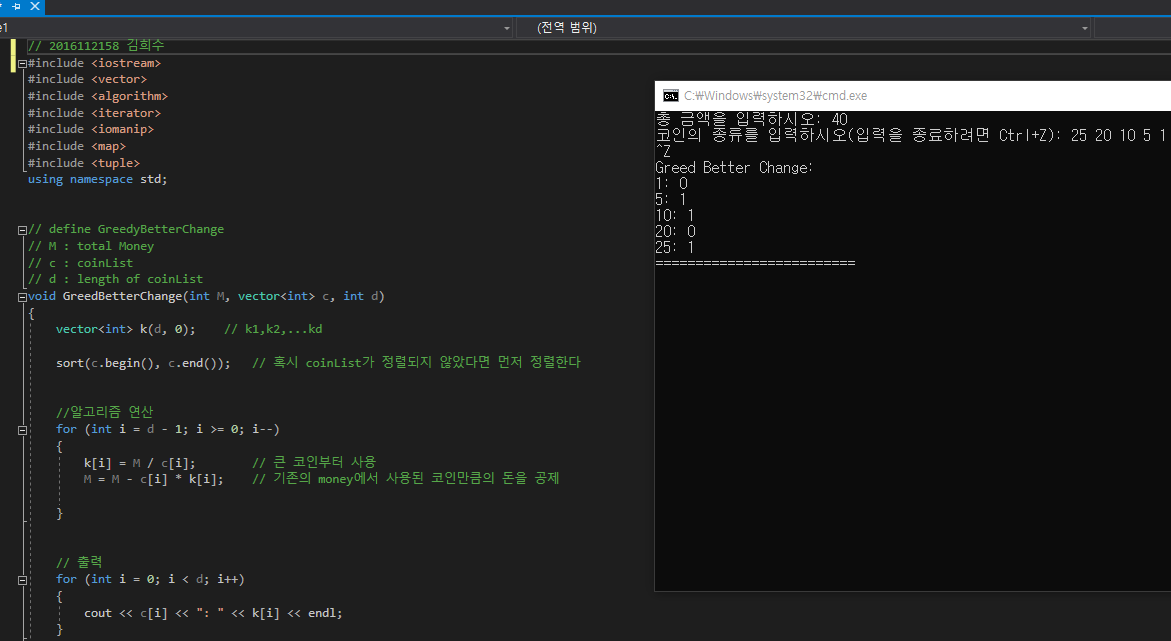
        k[i] := Money/coin[i]  //k is array. k1, k2, ... , kd

        Money := Money - coin[i] \* k[i]

    return k

d는 코인종류의 개수. d부터 시작하는게 의미하는 바는 “큰 수”부터 시작함을 의미한다. 임시 배열 k에 Money를 해당 인덱스의 coin으로 나눈 몫을 저장하고 coin \* k(== coin \* 몫)을 제외한 값을 Money로 다시 정의한다. 예를 들어 Money를 40이라 하고, coin의 (d-1)번째 원소를 25라고 하면 k[d-1]에는 1이 저장되고 Money에는 15가 저장된다. 이를 반복하면 배열 k에는 40을 25,20,10,5,1로 나눈 몫들이 저장될 것이다.

<2-2>



<2-3 BruteForceChange 유사코드>

BruteForceChange(coinValueList, total, numCoins, coinDict, build):

    if build == true:

for i := 0 to length(coinValueList):

         coinDict[coinValueList[i]]

    if total == 0:

        return (numCoins, coinDict)

    bestCoins := -1

    bestDict := make\_dictionary()   // make empty dictionary

    for i := 0 to length(coinValueList):

        dictCopy := make\_dictionary()

        for i:=0 to length(coinValueList):

            dictCopy[coin] = coinDict[coin]

        coin := conValueList[i]

        if coin <= total:

            dictCopy[coin] += 1

            (subCoins, subDict) := BruteForceChange(coinValueList, totla-coin, numCoins+1, dictCopy)

            if bestCoins == -1 && subCoins < bestCoins:

                bestCoins := subCoins

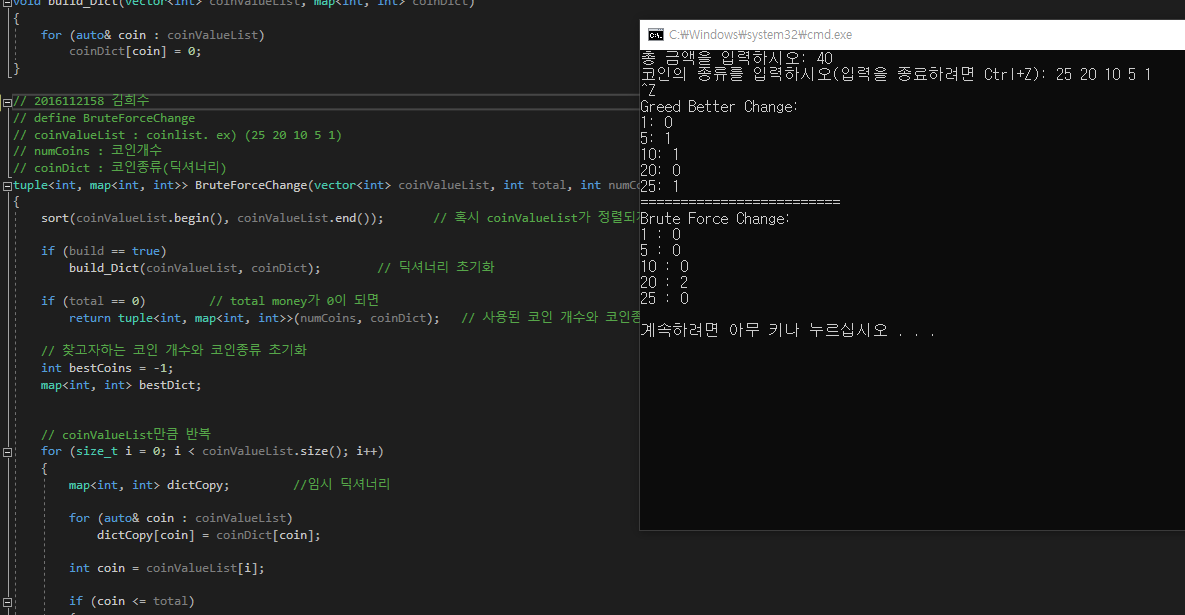
                bestDict := subDict

    return (bestCoins, bestDict)

함수는 int와 dictionary를 반환한다. 매개변수부터 설명하겟다. coinValueList는 코인 종류의 배열이고, total은 바꾸고자 하는 돈, numCoins는 해당코인의 개수, coinDict는 코인 종류의 딕셔너리이다. build는 처음에 0으로 초기화를 할 때를 판단한다. if(total == 0)부분은 재귀의 끝을 리턴하는 부분이다. bestCoins와 bestDict는 최적의 코인의 개수와 코인종류에 초기값을 설정하는 부분이다.

그 후, coinValueList의 크기만큼 다음을 반복한다. 먼저 coinValueList의 원소를 키로 하고 coinDict을 copy한다. 이 복사된 딕셔너리를 dictCopy라 한다. 그 후 coinValueList의 원소에 대해서 이 원소가 total보다 작으면 해당 원소와 같은 값의 키를 가지는 dictCopy의 원소를 1증가시킨다. 그 후 재귀적으로 반복한다.

<2-4>



<2-5 GreedyBetter과 BruteForce가 값이 다른경우>

