BLM3021 Algorithm Analysis
Semester Project
Book Recommendation System

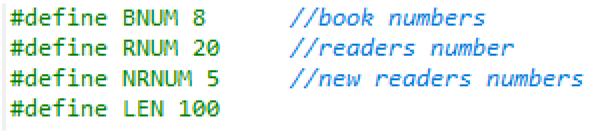
Group 2

Instructor: Assist. Prof. M. Amaç GÜVENSAN

Mounes Zawal – 18011906

General definitions:

Some explanation about the variables used in the code:



```
#define BNUM 8      //book numbers
#define RNUM 20     //readers number
#define NRNUM 5     //new readers numbers
#define LEN 100
```

winBook (int): **will store the final recommended book id.**

target (string): **Stores reader's name the user will enter.**

targetIndex (int): **gets the readers id, in other words, the row it is in.**

users (string): **Stores the users (readers) names.**

nUsers (string): **Stores the new users' names.**

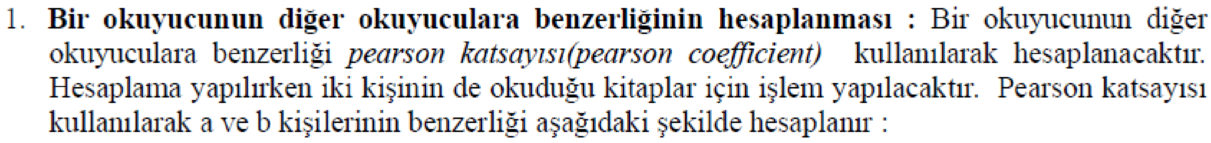
books (string): **Stores the books name.**

rates (int **): **Stores the rating given by x user to y book in a matrix.**

nuRates (int **): **Stores the rating given by x new user to y book in a matrix.**

Project aspects:

1.STEP

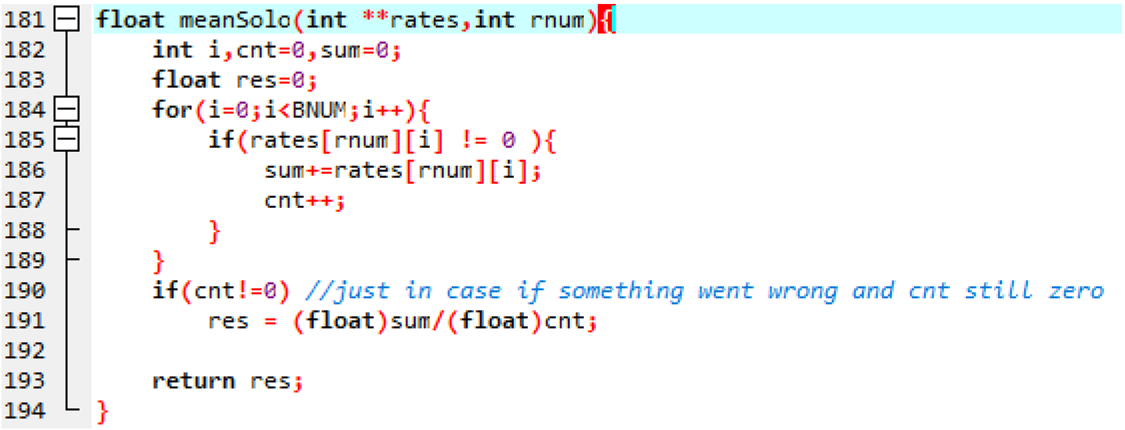


1. **Bir okuyucunun diğer okuyuculara benzerliğinin hesaplanması :** Bir okuyucunun diğer okuyuculara benzerliği *pearson katsayısı(pearson coefficient)* kullanılarak hesaplanacaktır. Hesaplama yapılırken iki kişinin de okuduğu kitaplar için işlem yapılacaktır. Pearson katsayısı kullanılarak a ve b kişilerinin benzerliği aşağıdaki şekilde hesaplanır :

rnum: **Row number.**

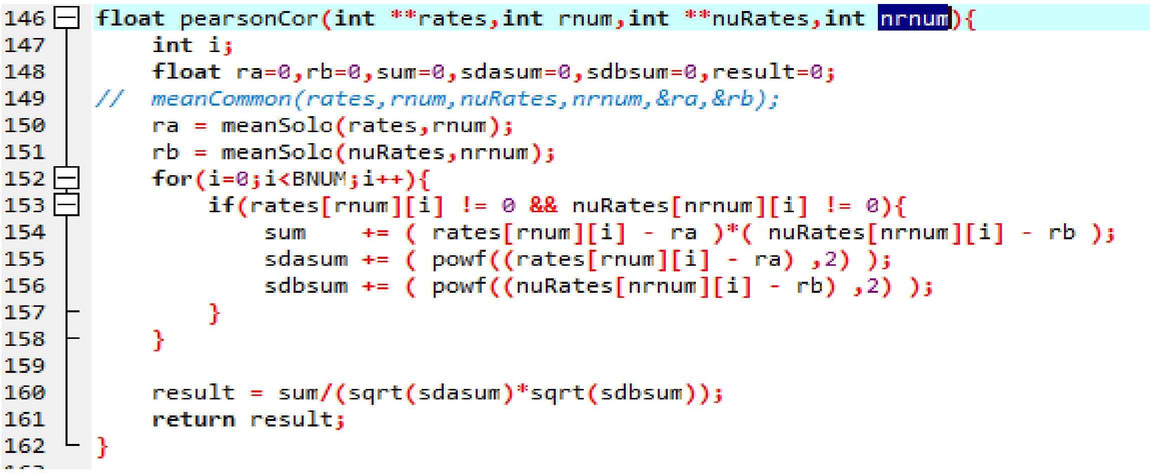
nrnum: **New user row number.**

This function is used to calculate the mean of read books ratings of a user.



```
float meanSolo(int **rates,int rnum){
    int i,cnt=0,sum=0;
    float res=0;
    for(i=0;i<BNUM;i++){
        if(rates[rnum][i] != 0 ){
            sum+=rates[rnum][i];
            cnt++;
        }
    }
    if(cnt!=0) //just in case if something went wrong and cnt still zero
        res = (float)sum/(float)cnt;

    return res;
}
```

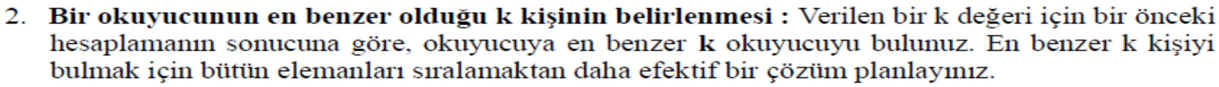
Calculating the Pearson Coefficient between two users.



```
float pearsonCor(int **rates,int rnum,int **nuRates,int nrnum){
    int i;
    float ra=0,rb=0,sum=0,sdasum=0,sdbsum=0,result=0;
//  meanCommon(rates,rnum,nuRates,nrnum,&ra,&rb);
    ra = meanSolo(rates,rnum);
    rb = meanSolo(nuRates,nrnum);
    for(i=0;i<BNUM;i++){
        if(rates[rnum][i] != 0 && nuRates[nrnum][i] != 0){
            sum    += ( rates[rnum][i] - ra )*( nuRates[nrnum][i] - rb );
            sdasum += ( powf((rates[rnum][i] - ra) ,2) );
            sdbsum += ( powf((nuRates[nrnum][i] - rb) ,2) );
        }
    }

    result = sum/(sqrt(sdasum)*sqrt(sdbsum));
    return result;
}
```

Combining these two functions we can achieve the first step of the recommendation process which is to calculate similarity.

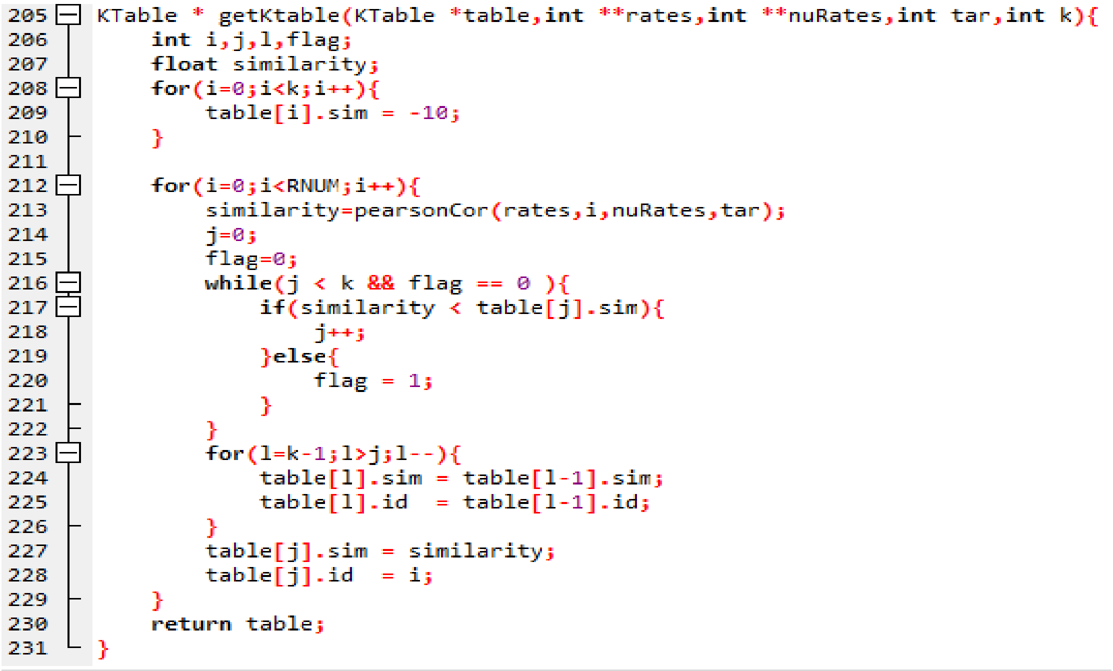
2.STEP



2. **Bir okuyucunun en benzer olduğu k kişinin belirlenmesi :** Verilen bir k değeri için bir önceki hesaplamanın sonucuna göre, okuyucuya en benzer **k** okuyucuyu bulunuz. En benzer k kişiyi bulmak için bütün elemanları sıralamaktan daha efektif bir çözüm planlayınız.

we calculate the similarity between new reader and each reader then inserts the similarity values with their row number in the correct cell, when a bigger similarity value comes, first it slides the table array (k length) in a way that deletes the last element from the table and insert the new value associated with its id in the right cell.

time complexity: worst case - O(k).



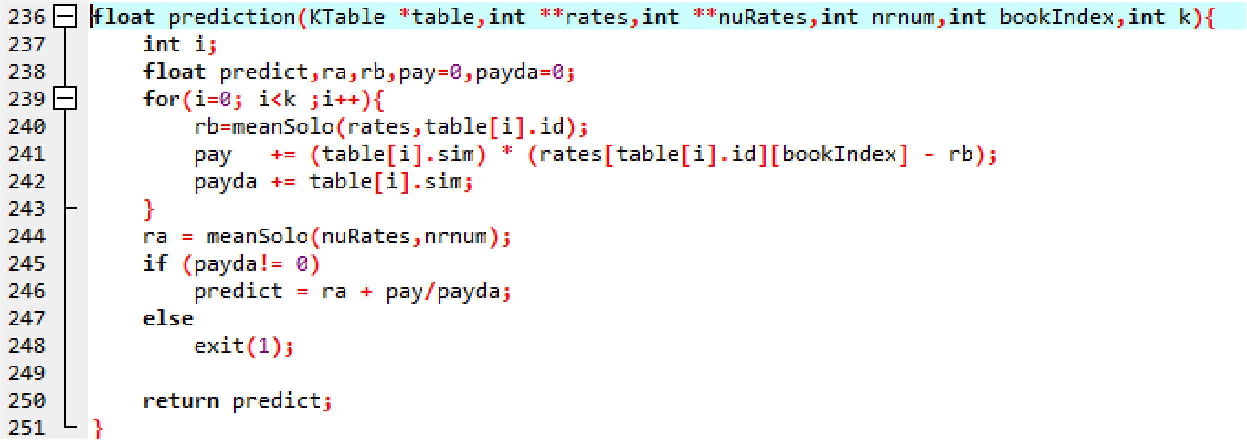
```
KTable * getKtable(KTable *table,int **rates,int **nuRates,int tar,int k){
    int i,j,l,flag;
    float similarity;
    for(i=0;i<k;i++){
        table[i].sim = -10;
    }

    for(i=0;i<RNUM;i++){
        similarity=pearsonCor(rates,i,nuRates,tar);
        j=0;
        flag=0;
        while(j < k && flag == 0 ){
            if(similarity < table[j].sim){
                j++;
            }else{
                flag = 1;
            }
        }
        for(l=k-1;l>j;l--){
            table[l].sim = table[l-1].sim;
            table[l].id  = table[l-1].id;
        }
        table[j].sim = similarity;
        table[j].id  = i;
    }
    return table;
}
```

3.STEP

3. **Okuyucuya yeni kitap önerme :** Okuyucuya kitap önermek için en benzer k kişi için aşağıdaki eşitlik ile hesaplama yapınız . Bu eşitlikte **N** değeri kullanıcının en çok benzediği **k** kişiyi göstermektedir Hesaplanan *pred(a,p)* değeri kişinin okumadığı bir kitaba vereceği puanın tahmini değeridir.

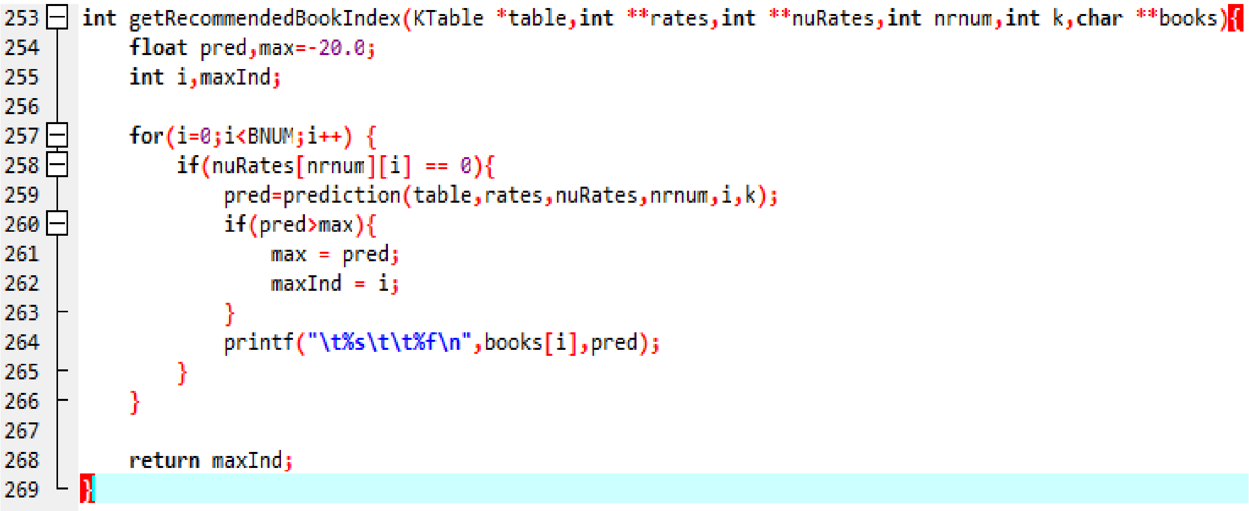
Calculates rating prediction for the unread book with (bookIndex) id.

Instead of recalculating the similarities I am just using the K similarity table.



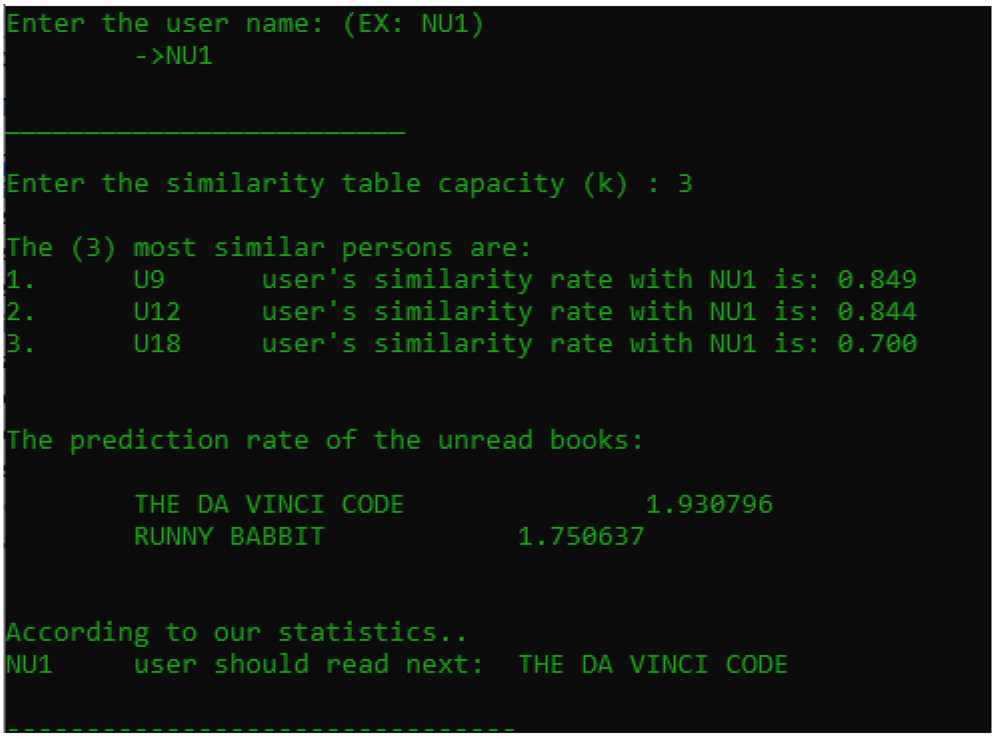
```
float prediction(KTable *table,int **rates,int **nuRates,int nrnum,int bookIndex,int k){
    int i;
    float predict,ra,rb,pay=0,payda=0;
    for(i=0; i<k ;i++){
        rb=meanSolo(rates,table[i].id);
        pay   += (table[i].sim) * (rates[table[i].id][bookIndex] - rb);
        payda += table[i].sim;
    }
    ra = meanSolo(nuRates,nrnum);
    if (payda!= 0)
        predict = ra + pay/payda;
    else
        exit(1);

    return predict;
}
```

And In this function, we calculate the predicted rate for each unread book of the new user and returns the winning book id.

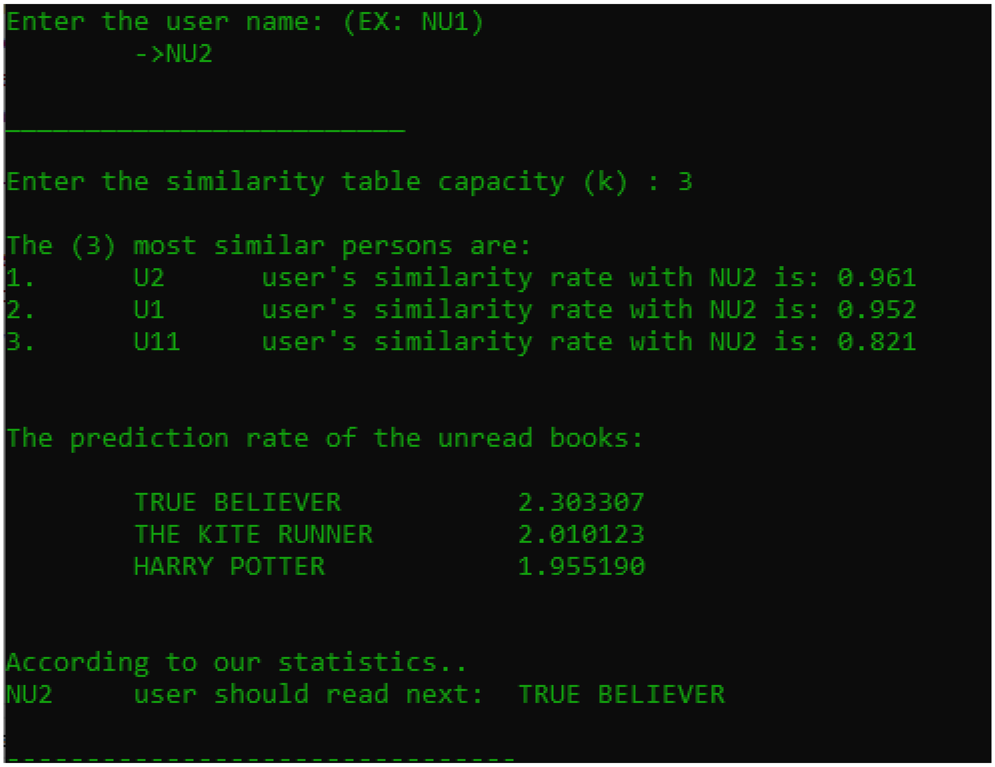


```
int getRecommendedBookIndex(KTable *table,int **rates,int **nuRates,int nrnum,int k,char **books){
    float pred,max=-20.0;
    int i,maxInd;

    for(i=0;i<BNUM;i++) {
        if(nuRates[nrnum][i] == 0){
            pred=prediction(table,rates,nuRates,nrnum,i,k);
            if(pred>max){
                max = pred;
                maxInd = i;
            }
            printf("\t%s\t\t%f\n",books[i],pred);
        }
    }

    return maxInd;
}
```

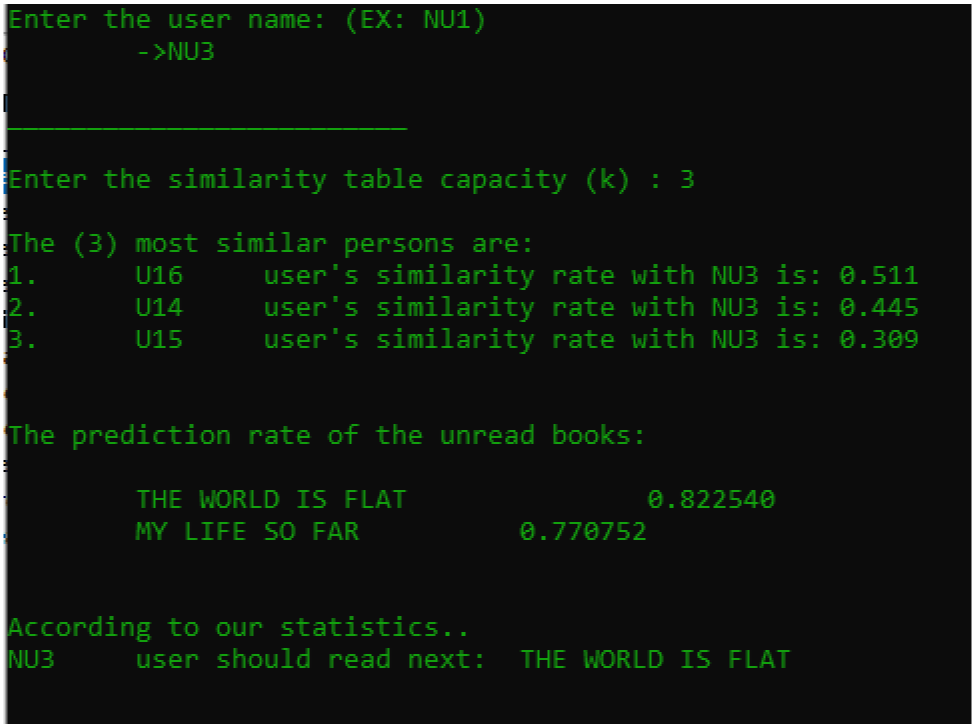
Screenshots:



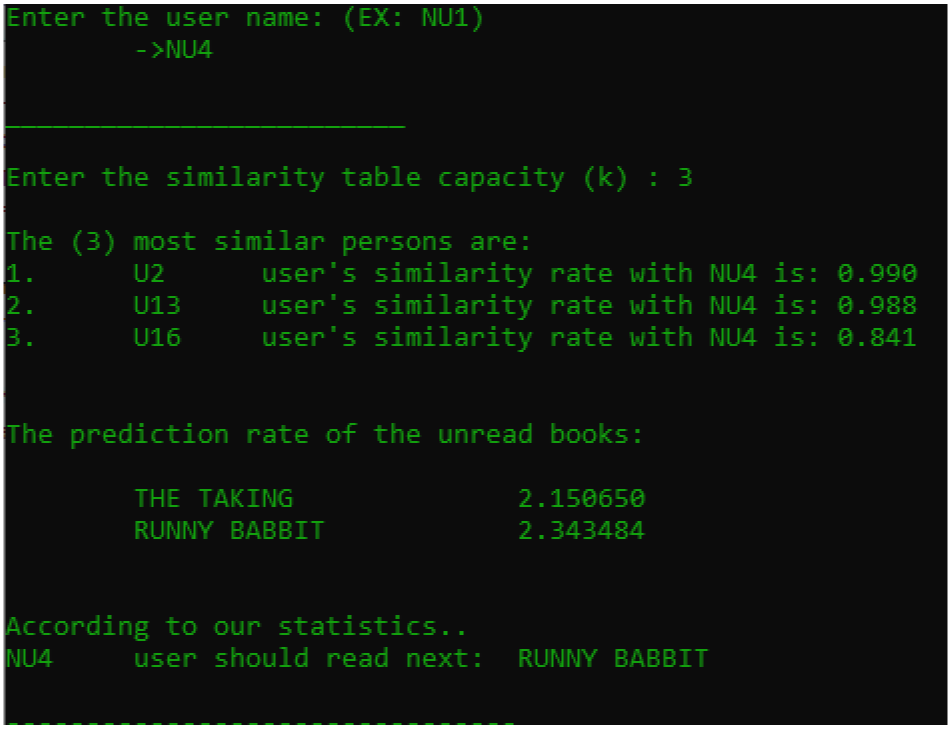
```
Enter the user name: (EX: NU1)
        ->NU1


_________________________

Enter the similarity table capacity (k) : 3

The (3) most similar persons are:
1.      U9      user's similarity rate with NU1 is: 0.849
2.      U12     user's similarity rate with NU1 is: 0.844
3.      U18     user's similarity rate with NU1 is: 0.700


The prediction rate of the unread books:

        THE DA VINCI CODE               1.930796
        RUNNY BABBIT            1.750637


According to our statistics..
NU1     user should read next:  THE DA VINCI CODE

--------------------------------
```



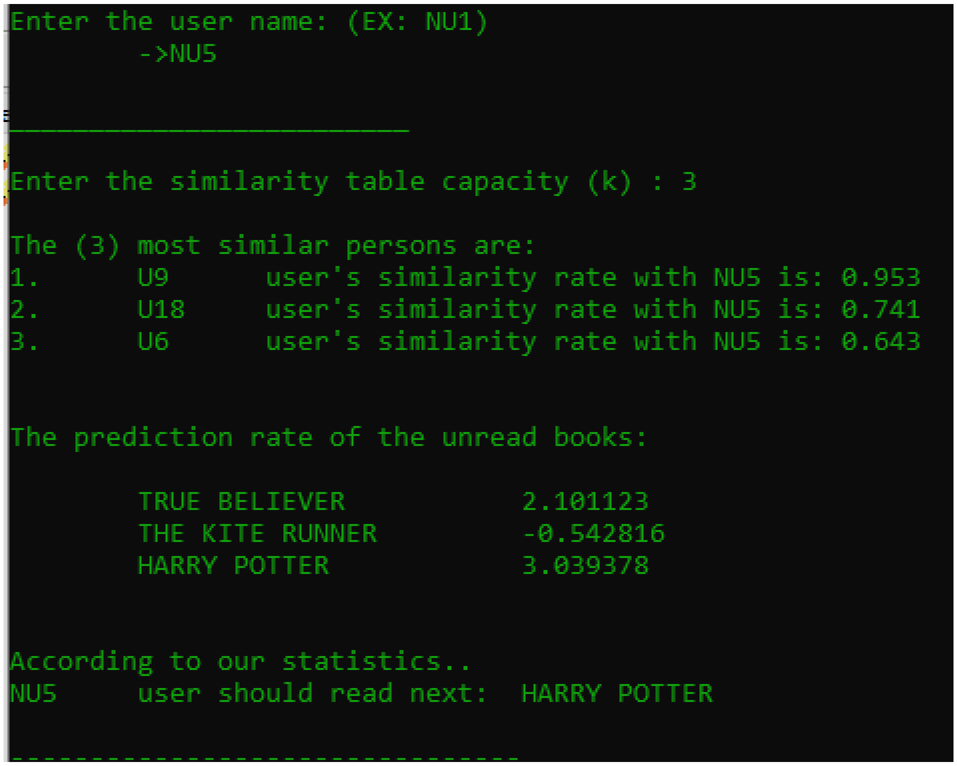
```
Enter the user name: (EX: NU1)
        ->NU2


_________________________

Enter the similarity table capacity (k) : 3

The (3) most similar persons are:
1.      U2      user's similarity rate with NU2 is: 0.961
2.      U1      user's similarity rate with NU2 is: 0.952
3.      U11     user's similarity rate with NU2 is: 0.821


The prediction rate of the unread books:

        TRUE BELIEVER           2.303307
        THE KITE RUNNER         2.010123
        HARRY POTTER            1.955190


According to our statistics..
NU2     user should read next:  TRUE BELIEVER

--------------------------------
```



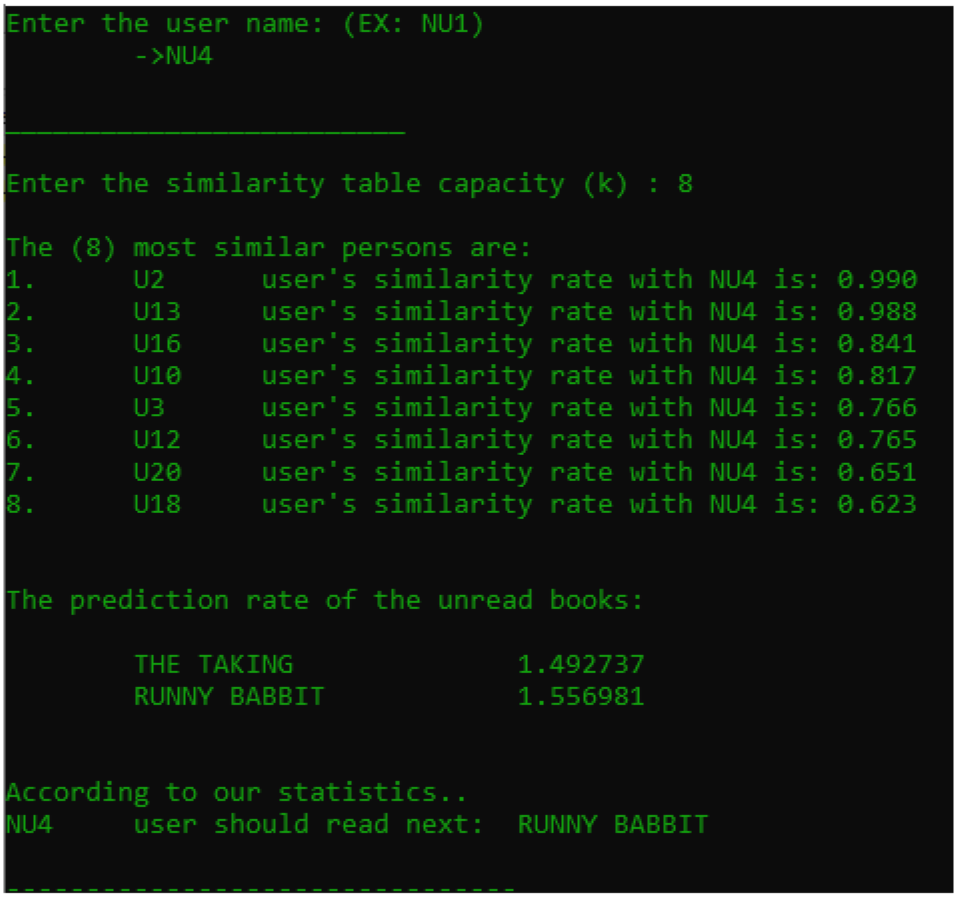
```
Enter the user name: (EX: NU1)
        ->NU3


_________________________

Enter the similarity table capacity (k) : 3

The (3) most similar persons are:
1.      U16     user's similarity rate with NU3 is: 0.511
2.      U14     user's similarity rate with NU3 is: 0.445
3.      U15     user's similarity rate with NU3 is: 0.309


The prediction rate of the unread books:

        THE WORLD IS FLAT               0.822540
        MY LIFE SO FAR          0.770752


According to our statistics..
NU3     user should read next:  THE WORLD IS FLAT
```



```
Enter the user name: (EX: NU1)
        ->NU4


_________________________

Enter the similarity table capacity (k) : 3

The (3) most similar persons are:
1.      U2      user's similarity rate with NU4 is: 0.990
2.      U13     user's similarity rate with NU4 is: 0.988
3.      U16     user's similarity rate with NU4 is: 0.841


The prediction rate of the unread books:

        THE TAKING              2.150650
        RUNNY BABBIT            2.343484


According to our statistics..
NU4     user should read next:  RUNNY BABBIT

--------------------------------
```



```
Enter the user name: (EX: NU1)
        ->NU5

_________________________

Enter the similarity table capacity (k) : 3

The (3) most similar persons are:
1.      U9      user's similarity rate with NU5 is: 0.953
2.      U18     user's similarity rate with NU5 is: 0.741
3.      U6      user's similarity rate with NU5 is: 0.643


The prediction rate of the unread books:

        TRUE BELIEVER           2.101123
        THE KITE RUNNER         -0.542816
        HARRY POTTER            3.039378


According to our statistics..
NU5     user should read next:  HARRY POTTER

--------------------------------
```



```
Enter the user name: (EX: NU1)
        ->NU4

_________________________

Enter the similarity table capacity (k) : 8

The (8) most similar persons are:
1.      U2      user's similarity rate with NU4 is: 0.990
2.      U13     user's similarity rate with NU4 is: 0.988
3.      U16     user's similarity rate with NU4 is: 0.841
4.      U10     user's similarity rate with NU4 is: 0.817
5.      U3      user's similarity rate with NU4 is: 0.766
6.      U12     user's similarity rate with NU4 is: 0.765
7.      U20     user's similarity rate with NU4 is: 0.651
8.      U18     user's similarity rate with NU4 is: 0.623


The prediction rate of the unread books:

        THE TAKING              1.492737
        RUNNY BABBIT            1.556981


According to our statistics..
NU4     user should read next:  RUNNY BABBIT

--------------------------------
```