**Source Code**

**cite\_split.c**

#include<stdlib.h>

#include<stdio.h>

long int cnt=0,i;

char ch;

int main()

{

FILE \*fp,\*fp1,\*fp5;

fp=fopen("citations\_3.txt","r");

fp1=fopen("citations\_7.txt","w");

fp5=fopen("citations\_6.txt","w");

ch=fgetc(fp);

while(ch!=EOF)

{

if(ch=='#')

{

if(ch=='\*')

cnt++;

}

ch=fgetc(fp);

}

rewind(fp);

for(i=0;i<150000;i++)

{

if(i<145584)

{

ch=fgetc(fp);

fprintf(fp1,"%c",ch);

while(ch!='\*')

{ch=fgetc(fp);

fprintf(fp1,"%c",ch); }

while(ch!='#')

{ch=fgetc(fp);

fprintf(fp1,"%c",ch); }

while(ch!='@')

{ch=fgetc(fp);

fprintf(fp1,"%c",ch); }

while(ch!='#')

{ch=fgetc(fp);

fprintf(fp1,"%c",ch); }

while(ch!='l')

{ch=fgetc(fp);

fprintf(fp1,"%c",ch); }

while(ch!='#')

{ch=fgetc(fp);

fprintf(fp1,"%c",ch); }

while(ch!='t')

{ch=fgetc(fp);

fprintf(fp1,"%c",ch); }

while(ch!='#')

{ch=fgetc(fp);

fprintf(fp1,"%c",ch); }

while(ch!='c')

{ch=fgetc(fp);

fprintf(fp1,"%c",ch); }

while(ch!='#')

{ch=fgetc(fp);

fprintf(fp1,"%c",ch); }

while(ch!='f')

{ch=fgetc(fp);

fprintf(fp1,"%c",ch); }

while(ch!='#')

{ch=fgetc(fp);

fprintf(fp1,"%c",ch); }

while(ch!='i')

{ch=fgetc(fp);

fprintf(fp1,"%c",ch); }

}

printf("Records checked %ld\n",i+1);

}

}

**Coauthor\_count.c**

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

long int no\_authors;

void count\_authors()

{

FILE \*fp2;

long int no\_lines=0;

char ch1;

fp2=fopen("unique\_authors.txt","r");

if(fp2==NULL)

{printf("Author file not found... exiting\n");

exit(1);}

ch1=fgetc(fp2);

while(ch1!=EOF)

{

if(ch1=='\n')

no\_lines++;

ch1=fgetc(fp2);

}

fclose(fp2);

no\_authors=no\_lines;

}

long int get\_author\_index1(char \*author)

{

FILE \*fp1;

char name[400],ch2;

long int index,id;

fp1=fopen("unique\_authors\_id.txt","r");

do

{

fscanf(fp1,"%ld",&id);

ch2=fgetc(fp1);

ch2=fgetc(fp1);

index=0;

while(ch2!='\n'&&ch2!=EOF)

{

name[index]=ch2;

index++;

ch2=fgetc(fp1);

}

name[index-1]='\0';

name[index]='\0';

if(strcmp(name,author)==0)

break;

if(id==no\_authors-1)

{

id++;

break;

}

}while(ch2!=EOF);

fclose(fp1);

if(id<no\_authors)

return id;

else

{

//printf("\nAuthor %s not found\n",author);

return -1;

}

}

long int get\_author\_index(char \*author)

{

FILE \*fp1;

char name[400],ch2;

long int index,id;

fp1=fopen("unique\_authors\_id.txt","r");

do

{

fscanf(fp1,"%ld",&id);

ch2=fgetc(fp1);

ch2=fgetc(fp1);

index=0;

while(ch2!='\n'&&ch2!=EOF)

{

name[index]=ch2;

index++;

ch2=fgetc(fp1);

}

name[index-1]='\0';

name[index]='\0';

if(strcmp(name,author)==0)

break;

if(id==no\_authors-1)

{

id++;

break;

}

}while(ch2!=EOF);

fclose(fp1);

if(id<no\_authors)

return id;

else

{

printf("\nAuthor %s not found\n",author);

return -1;

}

}

char ch,author[2000],filename1[80],paper\_index[20];

long int i=0,cnt=0,auth\_index[150],id,m;

int ind,j,n,conf\_year,p,start\_year=1980,end\_year=2007,q,r,v,k,l,num1,s,t,check;

struct record

{

long int auth\_id;

long int coauthor\_id[300];

int year\_attend[300][20];

int num;

};

int main()

{

struct record \*auth;

auth=(struct record \*)malloc(501061\*sizeof(struct record));

count\_authors();

FILE \*fp,\*fp3;

for(m=1;m<501061;m++)

{

auth[m].auth\_id=m;

for(n=0;n<300;n++)

{

auth[m].coauthor\_id[n]=-1;

//auth[m].year\_attend[n]=-1;

for(q=0;q<20;q++)

{

auth[m].year\_attend[n][q]=0;

}}

auth[m].num=-1;

}

fp=fopen("citations.txt","r");

ch=fgetc(fp);

while(ch!=EOF)

{

if(ch=='#'){

ch=fgetc(fp);

if(ch=='\*')

cnt++;}

ch=fgetc(fp);}

rewind(fp);

printf("\nTotal papers to be checked %ld\n\n",cnt);

ch=fgetc(fp);

ch=fgetc(fp);

for(i=0;i<cnt;i++)

{

ind=0;

while(ch!='#')

{ch=fgetc(fp);}

while(ch!='@')

{ch=fgetc(fp);}

ch=fgetc(fp);

for(j=0;j<150;j++)

{auth\_index[j]=0;}

j=0;

while(ch!='#')

{

while(ch!='\n'&&ch!=',')

{

author[ind]=ch;

ch=fgetc(fp);

ind++;

}

author[ind]='\0';

auth\_index[j]=get\_author\_index1(author);

if(auth\_index[j]==-1)//if(ch=='\n')

{author[ind-1]='\0';

auth\_index[j]=get\_author\_index(author); }

if(ch==',')

{

ind=0;

j++;

}

ch=fgetc(fp);

}

while(ch!='l')

{ch=fgetc(fp); }

while(ch!='#')

{ch=fgetc(fp);

}

while(ch!='t')

{ch=fgetc(fp); }

fscanf(fp,"%d",&conf\_year);

k=0;

while(auth\_index[k]!=0)

{

id=auth\_index[k];

if(auth[id].num==-1)

{

for(l=0;l<=j;l++)

{

if(id!=auth\_index[l])

{

auth[id].num++;

num1=auth[id].num;

auth[id].coauthor\_id[num1]=auth\_index[l];

auth[id].year\_attend[num1][0]=conf\_year;

}

}

}

else

{

for(l=0;l<=j;l++)

{

check=0;

for(s=0;s<=auth[id].num;s++)

{

if(auth[id].coauthor\_id[s]==auth\_index[l])

{

check=1;

t=0;

while(auth[id].year\_attend[s][t]!=0)

{t++;}

auth[id].year\_attend[s][t]=conf\_year;

break;

}

}

if(check==0&&id!=auth\_index[l])

{

auth[id].num++;

num1=auth[id].num;

auth[id].coauthor\_id[num1]=auth\_index[l];

auth[id].year\_attend[num1][0]=conf\_year;

}

}

}

k++;

}

while(ch!='c')

{ch=fgetc(fp);}

while(ch!='#')

{ch=fgetc(fp);}

ch=fgetc(fp);

while(ch!='#')

{ch=fgetc(fp);}

while(ch!='x')

{ch=fgetc(fp);}

r=0;

ch=fgetc(fp);

while(ch!='\n')

{

paper\_index[r]=ch;

r++;

ch=fgetc(fp);

}

paper\_index[r]='\0';

paper\_index[r-1]='\0';

printf("Records checked %ld paper index %s\n",i+1,paper\_index);

if(i<cnt-1)

{

while(ch!='#')

{ch=fgetc(fp);}

while(ch!='\*')

{ch=fgetc(fp);}

}

}// Total records for loop

fp3=fopen("common\_coauthor.txt","w");

for(m=1;m<501061;m++)

{

fprintf(fp3,"@%ld %d\n",auth[m].auth\_id,auth[m].num+1);

for(n=0;n<=auth[m].num;n++)

{

fprintf(fp3,"\*%ld",auth[m].coauthor\_id[n]);

for(p=0;p!=auth[m].year\_attend[n][p]!=0;p++)

{ }

fprintf(fp3," $%d",p);

for(q=0;q!=auth[m].year\_attend[n][q]!=0;q++)

{

fprintf(fp3," !%d",auth[m].year\_attend[n][q]);

}

fprintf(fp3,"\n");

}

fprintf(fp3,"\n");

fclose(fp3);

}

**Coauthor\_link\_domainname.c**

#include<stdlib.h>

#include<string.h>

#include<stdio.h>

char ch,cont\_name[30],author[2000],conf\_name[6],filename1[80],paper\_index[20];

long int i=0,cnt=0,auth\_index[80];

int ind,j,cont\_id[80],k,m,l,n,conf\_year,p,start\_year=1980,end\_year=2007,q,r;//s=0,t;

void print()

{

if(j>0)

{printf("Matching record found for year %d & index %s\n",conf\_year,paper\_index);

sprintf(filename1,"./art\_int\_coauthor/coauthor\_links\_%d.txt",conf\_year);

fp4=fopen(filename1,"a+");

for(m=0;m<j;m++)

{

for(n=m+1;n<=j;n++)

{

fprintf(fp4,"Author1 @%ld #%d Author2 @%ld #%d paper &%s conf %s year !%d\n",auth\_index[m],cont\_id[m],auth\_index[n],cont\_id[n],paper\_index,conf\_name,conf\_year);

}

}

fclose(fp4);

}

}

int main()

{

FILE \*fp;

fp=fopen("citations1.txt","r");

char \*cont1="north\_america",\*cont2="south\_america",\*cont3="europe",\*cont4="asia",\*cont5="africa";

char \*conf1="AAAI",\*conf2="ECAI",\*conf3="ICDE",\*conf4="ICRA",\*conf5="NIPS",\*conf6="SIGIR";

count\_authors();

ch=fgetc(fp);

while(ch!=EOF)

{

if(ch=='#')

{

ch=fgetc(fp);

if(ch=='\*')

cnt++;

}

ch=fgetc(fp);

}

rewind(fp);

printf("\nTotal papers to be checked %ld\n\n",cnt);

ch=fgetc(fp);

ch=fgetc(fp);

for(i=0;i<cnt;i++)

{

ind=0;

while(ch!='#')

{ch=fgetc(fp);}

while(ch!='@')

{ch=fgetc(fp);}

ch=fgetc(fp);

for(j=0;j<80;j++)

{auth\_index[j]=0;}

j=0;

while(ch!='#')

{

while(ch!='\n'&&ch!=',')

{

author[ind]=ch;

ch=fgetc(fp);

ind++;

}

author[ind]='\0';

auth\_index[j]=get\_author\_index1(author);

if(auth\_index[j]==-1)//if(ch=='\n')

{author[ind-1]='\0';

auth\_index[j]=get\_author\_index(author);}

if(ch==',')

{

ind=0;

j++;

}

ch=fgetc(fp);;

}

k=0;

ch=fgetc(fp);

ch=fgetc(fp);

ch=fgetc(fp);

for(l=0;l<80;l++)

{cont\_id[l]=5;}

l=0;

while(ch!='#')

{

while(ch!='\n'&&ch!=',')

{

cont\_name[k]=ch;

ch=fgetc(fp);

k++;

}

cont\_name[k]='\0';

if(ch=='\n')

cont\_name[k-1]='\0';

if(strcmp(cont\_name,cont1)==0)

cont\_id[l]=2;

else if(strcmp(cont\_name,cont2)==0)

cont\_id[l]=3;

else if(strcmp(cont\_name,cont3)==0)

cont\_id[l]=4;

else if(strcmp(cont\_name,cont4)==0)

cont\_id[l]=0;

else if(strcmp(cont\_name,cont5)==0)

cont\_id[l]=1;

if(ch==',')

{

k=0;

l++;

}

ch=fgetc(fp);

}

fgetc(fp);

fscanf(fp,"%d",&conf\_year);

//t=get\_fp(conf\_year);

//printf("\nt=%d",t);

while(ch!='c')

{ch=fgetc(fp);}

p=0;

ch=fgetc(fp);

while(ch!='\n') //checking conf names

{

conf\_name[p]=ch;

p++;

ch=fgetc(fp);

}

conf\_name[p]='\0';

conf\_name[p-1]='\0';

//fscanf(fp,"%c",&conf\_);

while(ch!='#')

{ch=fgetc(fp);}

ch=fgetc(fp);

while(ch!='#')

{ch=fgetc(fp);}

while(ch!='x')

{ch=fgetc(fp);}

r=0;

ch=fgetc(fp);

while(ch!='\n')

{

paper\_index[r]=ch;

r++;

ch=fgetc(fp);

}

paper\_index[r]='\0';

paper\_index[r-1]='\0';

if((strncmp(conf\_name,conf1,4)==0)&&(conf\_year>=start\_year&&conf\_year<=end\_year))

{print();conf\_name[0]='A';conf\_name[1]='A';conf\_name[2]='A';conf\_name[3]='I';conf\_name[4]='\0';}

else if((strncmp(conf\_name,conf2,4)==0)&&(conf\_year>=start\_year&&conf\_year<=end\_year))

{print();conf\_name[0]='E';conf\_name[1]='C';conf\_name[2]='A';conf\_name[3]='I';conf\_name[4]='\0';}

else if((strncmp(conf\_name,conf3,4)==0)&&(conf\_year>=start\_year&&conf\_year<=end\_year))

{print();conf\_name[0]='I';conf\_name[1]='C';conf\_name[2]='D';conf\_name[3]='E';conf\_name[4]='\0';}

else if((strncmp(conf\_name,conf4,4)==0)&&(conf\_year>=start\_year&&conf\_year<=end\_year))

{print();conf\_name[0]='I';conf\_name[1]='C';conf\_name[2]='R';conf\_name[3]='A';conf\_name[4]='\0';}

else if((strncmp(conf\_name,conf5,4)==0)&&(conf\_year>=start\_year&&conf\_year<=end\_year))

{print();conf\_name[0]='N';conf\_name[1]='I';conf\_name[2]='P';conf\_name[3]='S';conf\_name[4]='\0';}

else

{

if((strncmp(conf\_name,conf6,5)==0)&&(conf\_year>=start\_year&&conf\_year<=end\_year))

{

print();conf\_name[0]='S';conf\_name[1]='I';conf\_name[2]='G';conf\_name[3]='I';conf\_name[4]='R';conf\_name[5]='\0';

}

}

printf("Records checked %ld paper index %s\n",i+1,paper\_index);

if(i<cnt-1)

{

while(ch!='#')

{ch=fgetc(fp);}

while(ch!='\*')

{ch=fgetc(fp);}

}}

}

**Count\_interactn\_1.c**

#include<stdio.h>

#include<stdlib.h>

int main()

{

FILE \*fp,\*fp1,\*fp2;

long int auth[13000][2],id1,id2,auth1\_id,auth2\_id;

int cnt\_inter[13000],i,j,no\_inter,k,conf\_year,year[13000][20],l,m,act\_year[13000],check;

//float p1;

char filename[80],ch1,ch,ch2;

fp2=fopen("random\_author\_pair\_4.txt","r");

for(i=0;i<13000;i++)

{

fscanf(fp2,"%ld",&id1);

fscanf(fp2,"%ld",&id2);

auth[i][0]=id1;

auth[i][1]=id2;

fscanf(fp2,"%d",&act\_year[i]);

for(l=0;l<20;l++)

{year[i][l]=0;}

cnt\_inter[i]=0;

}

fclose(fp2);

for(j=0;j<4;j++)

{

no\_inter=0;

sprintf(filename,"%d.txt",j+1);

fp=fopen(filename,"r");

if(fp==NULL)

{

printf("Interaction list not found\n");

return 0;

}

ch1=fgetc(fp);

while(ch1!=EOF)

{

if(ch1=='\n')

no\_inter++;

ch1=fgetc(fp);

}

rewind(fp);

for(i=0;i<no\_inter;i++)

{

fscanf(fp,"%ld",&auth1\_id);

fscanf(fp,"%ld",&auth2\_id);

ch=fgetc(fp);

ch=fgetc(fp);

while(ch!=' ')

{

ch=fgetc(fp);

}

fscanf(fp,"%d",&conf\_year);

for(k=0;k<13000;k++)

{

if((auth1\_id==auth[k][0]&&auth2\_id==auth[k][1])||(auth1\_id==auth[k][1]&&auth2\_id==auth[k][0]))

{

cnt\_inter[k]++;

l=0;check=0;

while(year[k][l]!=0)

{

if(year[k][l]==conf\_year)

{

check=1;

break;

}

else

l++;

}

if(check!=1)

year[k][l]=conf\_year;

}

}

//printf("Author1 %ld Author2 %ld of year %d interacted %d times\n",auth1\_id,auth2\_id,act\_year,cnt\_inter);

printf("File %d records checked %d out of %d\n",j+1,i+1,no\_inter);

}

fclose(fp);

}

fp1=fopen("interaction\_noc\_unique\_finite.txt","w");

for(i=0;i<13000;i++)

{

if(cnt\_inter[i]>0)

{

fprintf(fp1,"@%ld @%ld of year \*%d interacted #%d times",auth[i][0],auth[i][1],act\_year[i],cnt\_inter[i]);

for(m=0;year[i][m]!=0;m++)

fprintf(fp1," !%d",year[i][m]);

fprintf(fp1,"\n");

}

}

fclose(fp1);

}

**Filter\_publish.c**

#include<stdlib.h>

#include<stdio.h>

#include<string.h>

long int no\_authors,auth[75266][2],auth1,auth2,r=0,p;

int res;

int check\_prev(long int auth1,long int auth2)

{

res=0;

for(p=0;p<=r;p++)

{

if((auth[p][0]==auth1&&auth[p][1]==auth2)||(auth[p][0]==auth2&&auth[p][1]==auth1))

{

res=-1;

break;

}

}

if(res==0)

{

r++;

auth[r][0]=auth1;

auth[r][1]=auth2;

res=1;

}

return res;

}

char ch,ch1,author[2000],conf\_name[10],filename1[80],filename2[80];

long int i=0,cnt=0,auth\_index[150],auth\_id,t,paper\_index;

int ind,j,conf\_year,start\_year=1980,end\_year=2007,q,s,cont\_id1,cont\_id2,year,k,l,m;

int arr[501061];

int main()

{

FILE \*fp;

fp=fopen("citations.txt","r");

count\_authors();

ch=fgetc(fp);

while(ch!=EOF)

{

if(ch=='#')

{

ch=fgetc(fp);

if(ch=='\*')

cnt++;

}

ch=fgetc(fp);

}

rewind(fp);

printf("\nTotal papers to be checked %ld\n\n",cnt);

arr[0]=-1;

for(t=1;t<501061;t++)

{arr[t]=0;}

ch=fgetc(fp);

ch=fgetc(fp);

for(i=0;i<cnt;i++)

{

ind=0;

while(ch!='#')

{ch=fgetc(fp);}

while(ch!='@')

{ch=fgetc(fp);}

ch=fgetc(fp);

for(j=0;j<150;j++)

{auth\_index[j]=0;}

j=0;

while(ch!='#')

{

while(ch!='\n'&&ch!=',')

{

author[ind]=ch;

ch=fgetc(fp);

ind++;

}

author[ind]='\0';

auth\_index[j]=get\_author\_index1(author);

if(auth\_index[j]==-1)//if(ch=='\n')

{author[ind-1]='\0';

auth\_index[j]=get\_author\_index(author);

}

//printf("%s with index %ld\n",author,auth\_index[j]);

if(ch==',')

{

ind=0;

j++;

}

ch=fgetc(fp);

}

ch=fgetc(fp);

ch=fgetc(fp);

while(ch!='#')

{

while(ch!='\n')

{ch=fgetc(fp);}

ch=fgetc(fp);

}

ch=fgetc(fp);

fscanf(fp,"%d",&conf\_year);

s=0;

while(auth\_index[s]!=0)

{

auth\_id=auth\_index[s];

if(auth\_id!=-1&&arr[auth\_id]==0)

arr[auth\_id]=conf\_year;

else if(arr[auth\_id]>conf\_year)

arr[auth\_id]=conf\_year;

s++;

}

while(ch!='c')

{ch=fgetc(fp);}

while(ch!='#')

{ch=fgetc(fp);}

while(ch!='#')

{ch=fgetc(fp);}

while(ch!='\n')

{ch=fgetc(fp);}

printf("Records checked %ld\n",i+1);

if(i<cnt-1)

{

while(ch!='#')

{ch=fgetc(fp);}

while(ch!='\*')

{ch=fgetc(fp);}

}

}

FILE \*fp3,\*fp4;

fp3=fopen("first\_publish.txt","w");

for(t=1;t<501061;t++)

{

fprintf(fp3,"@%ld !%d\n",t,arr[t]);

}

fclose(fp3);

auth[0][0]=0;auth[0][1]=0;

for(q=start\_year;q<=end\_year;q++)

{

sprintf(filename1,"./art\_int\_coauthor/coauthor\_links\_%d.txt",q);

fp3=fopen(filename1,"r");

if(fp3==NULL)

{

printf("Data for year %d doesnt exist\n",q);

}

else

{

sprintf(filename2,"./art\_int\_coauthor\_refined/coauthor\_links\_%d.txt",q);

fp4=fopen(filename2,"w");

l=0;

ch1=fgetc(fp3);

while(ch1!=EOF)

{

if(ch1=='\n')

l++;

ch1=fgetc(fp3);

}

rewind(fp3);

for(m=0;m<l;m++)

{

ch1=fgetc(fp3);

while(ch1!='@')

{ch1=fgetc(fp3);}

fscanf(fp3,"%ld",&auth1);

while(ch1!='#')

{ch1=fgetc(fp3);}

fscanf(fp3,"%d",&cont\_id1);

while(ch1!='@')

{ch1=fgetc(fp3);}

fscanf(fp3,"%ld",&auth2);

while(ch1!='#')

{ch1=fgetc(fp3);}

fscanf(fp3,"%d",&cont\_id2);

while(ch1!='&')

{ch1=fgetc(fp3);}

fscanf(fp3,"%ld",&paper\_index);

while(ch1!='\*')

{ch1=fgetc(fp3);}

ch1=fgetc(fp3);

k=0;

while(ch1!=' ')

{conf\_name[k]=ch1;

k++;

ch1=fgetc(fp3);

}

conf\_name[k]='\0';

while(ch1!='!')

{ch1=fgetc(fp3);}

fscanf(fp3,"%d",&year);

res=check\_prev(auth1,auth2);

if(res==1&&(arr[auth1]>0&&arr[auth1]<year)&&(arr[auth2]>0&&arr[auth2]<year))

{

fprintf(fp4,"Author1 @%ld #%d Author2 @%ld #%d paper &%ld conf \*%s year !%d\n",auth1,cont\_id1,auth2,cont\_id2,paper\_index,conf\_name,year);

}

while(ch1!='\n')

{ch1=fgetc(fp3);}

}

}

fclose(fp3);

fclose(fp4);

printf("Refined coauthor link processed for year %d\n",q);

}return 0;

}

**Prog1.c**

#include<stdio.h>

#include<stdlib.h>

int main()

{

FILE \*fp,\*fp1,\*fp2;

char ch,ch1,ch2,ch3,filename1[80];

long int auth1\_id,auth2\_id,auth3\_id,auth4\_id;

int conf\_year,no\_inter=0,no\_session=0,i,j,cnt=0,k,start\_year=1982,end\_year=2007;

fp=fopen("All\_interactions\_\_artint\_upto2007\_first\_author\_without\_workshops\_unique.txt","r");

fp1=fopen("All\_interactions\_\_artint\_upto2007\_first\_author\_without\_workshops\_unique\_noc.txt","w");

//printf("\n Hello");

if(fp==NULL){

printf("Interaction list not found\n");

return 0;

}

ch1=fgetc(fp);

while(ch1!=EOF)

{

if(ch1=='\n')

no\_inter++;

ch1=fgetc(fp);

}

rewind(fp);

for(i=0;i<no\_inter;i++)

{

fscanf(fp,"%ld",&auth1\_id);

fscanf(fp,"%ld",&auth2\_id);

ch=fgetc(fp);

ch=fgetc(fp);

while(ch!=' ')

{ch=fgetc(fp);

}

fscanf(fp,"%d",&conf\_year);

cnt=0;

if(conf\_year==1980)

{conf\_year=conf\_year+1;}

for(k=conf\_year+1;k<=end\_year;k++)

{

no\_session=0;

sprintf(filename1,"./toplayer/citation\_id\_%d\_new.txt",k);

//printf("%s",filename1);

fp2=fopen(filename1,"r");

ch2=fgetc(fp2);

while(ch2!=EOF)

{

if(ch2=='\n')

no\_session++;

ch2=fgetc(fp2);

}

rewind(fp2);

for(j=0;j<no\_session;j++)

{

ch3=fgetc(fp2);

fscanf(fp2,"%ld",&auth3\_id);

ch3=fgetc(fp2);

while(ch3!='@')

{ch3=fgetc(fp2);

}

fscanf(fp2,"%ld",&auth4\_id);

if((auth1\_id==auth3\_id&&auth2\_id==auth4\_id)||(auth1\_id==auth4\_id)&&(auth2\_id==auth3\_id))

cnt++;

ch3=fgetc(fp2);

while(ch3!='\n')

{ch3=fgetc(fp2);

}

//ch3=fgetc(fp2);

}

printf("\nChecked for year %d, no. of citations in that year %d",k,cnt);

//printf("%d",no\_session);

fclose(fp2);

}

printf("\nSession Author1 @%ld Author2 @%ld after year %d cited each other #%d times",auth1\_id,auth2\_id,conf\_year,cnt);

if(cnt>0)

fprintf(fp1,"Session Author1 @%ld Author2 @%ld after year %d cited each other #%d times\n",auth1\_id,auth2\_id,conf\_year,cnt);

}

fclose(fp);

fclose(fp1);

return 0;

}

**Feature\_citation.c**

#include<stdio.h>

#include<stdlib.h>

int main()

{

FILE \*fp,\*fp1,\*fp2;

char ch,ch1,ch2,ch3,filename1[80];

long int auth1\_id,auth2\_id,auth3\_id,auth4\_id;

int conf\_year[30],conf\_year\_1[30],no\_inter=0,no\_session=0,i,j,cnt1=0,cnt2=0,cnt3=0,cnt4=0,k,start\_year=1982,end\_year=2008,m,year,cnt,n;

fp=fopen("random\_author\_pair\_4.txt","r");

fp1=fopen("citations\_noc\_new.txt","w");

if(fp==NULL){

printf("Interaction list not found\n");

return 0;

}

ch1=fgetc(fp);

while(ch1!=EOF)

{

if(ch1=='\n')

no\_inter++;

ch1=fgetc(fp);

}

rewind(fp);

printf("\nTotal interactions to be checked %d",no\_inter);

for(i=0;i<no\_inter;i++)

{

fscanf(fp,"%ld",&auth1\_id);

fscanf(fp,"%ld",&auth2\_id);

fscanf(fp,"%d",&year);

for(m=0;m<30;m++)

{conf\_year[m]=0;conf\_year\_1[m]=0;}

m=0;cnt1=0;cnt2=0;cnt3=0;cnt4=0;cnt=0;n=0;

for(k=start\_year;k<=end\_year;k++)

{

no\_session=0;

sprintf(filename1,"./toplayer/citation\_id\_%d\_new.txt",k);

fp2=fopen(filename1,"r");

ch2=fgetc(fp2);

while(ch2!=EOF)

{

if(ch2=='\n')

no\_session++;

ch2=fgetc(fp2);

}

rewind(fp2);

for(j=0;j<no\_session;j++)

{

ch3=fgetc(fp2);

fscanf(fp2,"%ld",&auth3\_id);

ch3=fgetc(fp2);

while(ch3!='@')

{ch3=fgetc(fp2);

}

fscanf(fp2,"%ld",&auth4\_id);

if(auth1\_id==auth3\_id&&auth2\_id==auth4\_id&&k<1995)

{cnt1++;cnt++;

if(conf\_year[m-1]!=k||m==0)

{conf\_year[m]=k;

m++;}

}

else if(auth1\_id==auth3\_id&&auth2\_id==auth4\_id&&k>1995)

{cnt2++;cnt++;

if(conf\_year[m-1]!=k||m==0)

{conf\_year[m]=k;

m++;}

}

else if(auth1\_id==auth4\_id&&auth2\_id==auth3\_id&&k<1995)

{cnt3++;cnt++;

if(conf\_year\_1[n-1]!=k||n==0)

{conf\_year\_1[n]=k;

n++;}

}

else if(auth1\_id==auth4\_id&&auth2\_id==auth3\_id&&k>1995)

{cnt4++;cnt++;

if(conf\_year\_1[n-1]!=k||n==0)

{conf\_year\_1[n]=k;

n++;}

}

ch3=fgetc(fp2);

while(ch3!='\n')

{ch3=fgetc(fp2);

}

//ch3=fgetc(fp2);

}

printf("\nChecked for year %d, no. of citations in that year %d",k,cnt);

fclose(fp2);

}

printf("\nInteraction %d Author1 @%ld Author2 @%ld cited each other #%d times",i+1,auth1\_id,auth2\_id,cnt);

if(cnt==0)

fprintf(fp1,"@%ld @%ld of year \*%d cited #%d times\n",auth1\_id,auth2\_id,year,cnt);

else

{

fprintf(fp1,"@%ld @%ld of year \*%d cited #%d <1995 #%d >1995",auth1\_id,auth2\_id,year,cnt1,cnt2);

for(m=0;conf\_year[m]!=0;m++)

{fprintf(fp1," !%d",conf\_year[m]);}

fprintf(fp1,"\n");

fprintf(fp1,"@%ld @%ld of year \*%d cited #%d <1995 #%d >1995",auth2\_id,auth1\_id,year,cnt3,cnt4);

for(m=0;conf\_year\_1[m]!=0;m++)

{fprintf(fp1," !%d",conf\_year\_1[m]);}

fprintf(fp1,"\n");

}

}

fclose(fp);

fclose(fp1);

return 0;

}

**Same\_continent.c**

#include<stdio.h>

#include<stdlib.h>

int main()

{

FILE \*fp,\*fp1,\*fp2;

char ch,ch1,ch2,ch3,filename1[80];

long int auth1\_id,auth2\_id,auth3\_id,auth4\_id;

int conf\_year,no\_inter=0,no\_session=0,i,j,cnt=0,cnt1,cnt2,year;

fp=fopen("random\_author\_pair\_4.txt","r");

fp1=fopen("continent\_noc\_new.txt","w");

if(fp==NULL){

printf("Interaction list not found\n");

return 0;

}

ch1=fgetc(fp);

while(ch1!=EOF)

{

if(ch1=='\n')

no\_inter++;

ch1=fgetc(fp);

}

rewind(fp);

printf("\nTotal interactions to be checked %d",no\_inter);

for(i=0;i<no\_inter;i++)

{

fscanf(fp,"%ld",&auth1\_id);

fscanf(fp,"%ld",&auth2\_id);

fscanf(fp,"%d",&year);

no\_session=0;

sprintf(filename1,"./toplayer/citation\_id\_%d\_new.txt",year);

fp2=fopen(filename1,"r");

ch2=fgetc(fp2);

while(ch2!=EOF)

{

if(ch2=='\n')

no\_session++;

ch2=fgetc(fp2);

}

rewind(fp2);

for(j=0;j<no\_session;j++)

{

ch3=fgetc(fp2);

fscanf(fp2,"%ld",&auth3\_id);

ch3=fgetc(fp2);

ch3=fgetc(fp2);

fscanf(fp2,"%d",&cnt1);

ch3=fgetc(fp2);

while(ch3!='@')

{ch3=fgetc(fp2);

}

fscanf(fp2,"%ld",&auth4\_id);

ch3=fgetc(fp2);

ch3=fgetc(fp2);

fscanf(fp2,"%d",&cnt2);

ch3=fgetc(fp2);

while(ch3!='\n')

{ch3=fgetc(fp2);

}

if((auth1\_id==auth3\_id&&auth2\_id==auth4\_id)||(auth1\_id==auth4\_id&&auth2\_id==auth3\_id))

{

fprintf(fp1,"@%ld #%d @%ld #%d\n",auth1\_id,cnt1,auth2\_id,cnt2);

if(cnt1==cnt2)

cnt++;

break;

}

}

fclose(fp2);

printf("\nInteractions checked %d",i+1);

}

fprintf(fp1,"Total authors with same continent %d out of %d",cnt,no\_inter);

fclose(fp1);

}

**Select\_author.c**

#include<stdio.h>

#include<stdlib.h>

float rand\_num()

{

float temp;

struct timeval tm;

gettimeofday(&tm , NULL);

srand(tm.tv\_sec \* 1000000 + tm.tv\_usec);

temp = (rand()/(RAND\_MAX + 1.0));

return temp;

}

int main()

{

FILE \*fp,\*fp1,\*fp2;

long int auth[2000][2],id1,id2,auth1\_id,auth2\_id;

int cnt\_inter[2000],i,p2,j,no\_inter,k,conf\_year,l1,l=0,year[2000],cnt=0;

float p1;

char filename[80],ch1,ch;

for(j=2008;j<=2008;j++)

{

no\_inter=0;

sprintf(filename,"./toplayer/citation\_id\_%d\_new.txt",j);

fp=fopen(filename,"r");

if(fp==NULL)

{

printf("Citation list not found\n");

return 0;

}

ch1=fgetc(fp);

while(ch1!=EOF)

{

if(ch1=='\n')

no\_inter++;

ch1=fgetc(fp);

}

rewind(fp);

for(i=0;i<2000;i++)

{

p1=rand\_num();

p2=p1\*100000000;

l1=p2%no\_inter;

for(k=0;k<=l1;k++)

{

ch=fgetc(fp);

while(ch!='\n')

{ch=fgetc(fp);}

}

ch=fgetc(fp);

fscanf(fp,"%ld",&id1);

auth[l][0]=id1;

ch=fgetc(fp);

while(ch!='@')

{ch=fgetc(fp);}

fscanf(fp,"%ld",&id2);

auth[l][1]=id2;

year[l]=j;

cnt\_inter[l]=0;

rewind(fp);

cnt++;l++;

printf("\n Done for %d",cnt);

}

fclose(fp);

}

fp2=fopen("random\_author\_pair\_2.txt","a");

for(i=0;i<2000;i++)

{

fprintf(fp2,"%ld %ld %d\n",auth[i][0],auth[i][1],year[i]);

}

fclose(fp2);

}

**Same\_field.c**

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

char ch,author[2000],conf\_name[200],filename1[80],paper\_index[20];

long int i=0,cnt=0,auth\_index[150],id,m;

int ind,j,n,conf\_year,p,start\_year=1980,end\_year=2007,q,r,v,k,l,num1,s,t,check;

struct record

{

long int auth\_id;

char con\_name[50][200];

int year\_attend[50][20];

int num;

};

int main()

{

struct record \*auth;

auth=(struct record \*)malloc(501061\*sizeof(struct record));

count\_authors();

FILE \*fp,\*fp3;

for(m=1;m<501061;m++)

{

auth[m].auth\_id=m;

for(n=0;n<50;n++)

{

auth[m].con\_name[n][0]='\0';

//auth[m].year\_attend[n]=-1;

for(q=0;q<20;q++)

{

auth[m].year\_attend[n][q]=0;

}

}

auth[m].num=-1;

}

fp=fopen("citations.txt","r");

ch=fgetc(fp);

while(ch!=EOF)

{

if(ch=='#')

{

ch=fgetc(fp);

if(ch=='\*')

cnt++;

}

ch=fgetc(fp);

}

rewind(fp);

printf("\nTotal papers to be checked %ld\n\n",cnt);

ch=fgetc(fp);

ch=fgetc(fp);

for(i=0;i<cnt;i++)

{

ind=0;

while(ch!='#')

{ch=fgetc(fp);}

while(ch!='@')

{ch=fgetc(fp);}

ch=fgetc(fp);

for(j=0;j<150;j++)

{auth\_index[j]=0;}

j=0;

while(ch!='#')

{

while(ch!='\n'&&ch!=',')

{

author[ind]=ch;

ch=fgetc(fp);

ind++;

}

author[ind]='\0';

auth\_index[j]=get\_author\_index1(author);

if(auth\_index[j]==-1)//if(ch=='\n')

{author[ind-1]='\0';

auth\_index[j]=get\_author\_index(author); }

if(ch==',')

{

ind=0;

j++;

}

ch=fgetc(fp);

}

while(ch!='l')

{ch=fgetc(fp); }

while(ch!='#')

{ch=fgetc(fp);}

while(ch!='t')

{ch=fgetc(fp); }

fscanf(fp,"%d",&conf\_year);

while(ch!='#')

{ch=fgetc(fp); }

while(ch!='c')

{ch=fgetc(fp);}

while(ch!='#')

{ch=fgetc(fp); }

while(ch!='f')

{ch=fgetc(fp);}

p=0;

ch=fgetc(fp);

while(ch!='\n') //checking conf names

{

conf\_name[p]=ch;

p++;

ch=fgetc(fp);

}

conf\_name[p]='\0';

conf\_name[p-1]='\0';

k=0;

while(auth\_index[k]!=0)

{

l=0;char ch1;

id=auth\_index[k];

if(auth[id].num==-1)

{

auth[id].num++;

num1=auth[id].num;

ch1=conf\_name[l];

while(ch1!='\0')

{

auth[id].con\_name[num1][l]=ch1;

l++;

ch1=conf\_name[l];

}

auth[id].con\_name[num1][l]='\0';

auth[id].year\_attend[num1][0]=conf\_year;

}

else

{

for(s=0;s<=auth[id].num;s++)

{

check=0;

if(strcmp(auth[id].con\_name[s],conf\_name)==0)

{

check=1;

t=0;

while(auth[id].year\_attend[s][t]!=0)

{t++;}

auth[id].year\_attend[s][t]=conf\_year;

break;

}}

if(check==0)

{

auth[id].num++;

num1=auth[id].num;

ch1=conf\_name[l];

while(ch1!='\0')

{

auth[id].con\_name[num1][l]=ch1;

l++;

ch1=conf\_name[l];

}

auth[id].con\_name[num1][l]='\0';

auth[id].year\_attend[num1][0]=conf\_year;

}

}

k++;

}

while(ch!='#')

{ch=fgetc(fp);}

while(ch!='x')

{ch=fgetc(fp);}

r=0;

ch=fgetc(fp);

while(ch!='\n'&&ch!='#')

{

paper\_index[r]=ch;

r++;

ch=fgetc(fp);

}

paper\_index[r]='\0';

paper\_index[r-1]='\0';

printf("Records checked %ld paper index %s\n",i+1,paper\_index);

if(i<cnt-1)

{

while(ch!='#')

{ch=fgetc(fp);}

while(ch!='\*')

{ch=fgetc(fp);}

}

}fp3=fopen("same\_field.txt","w");

for(m=1;m<501061;m++)

{

fprintf(fp3,"@%ld\n",auth[m].auth\_id);

for(n=0;n<=auth[m].num;n++)

{

fprintf(fp3,"\*%s",auth[m].con\_name[n]);

for(q=0;q!=auth[m].year\_attend[n][q]!=0;q++)

{

//auth[m].year\_attend[n][q]=0;

fprintf(fp3," !%d",auth[m].year\_attend[n][q]);

}

fprintf(fp3,"\n");

}

fprintf(fp3,"\n");

}fclose(fp3);

}

**Same\_venue.c**

#include<stdio.h>

#include<stdlib.h>

int main()

{

FILE \*fp,\*fp1,\*fp2,\*fp3;

char ch,ch1,ch2,ch3;

long int auth1\_id,auth2\_id,auth3\_id,auth4\_id;//auth[13000][2];

int conf\_year,no\_inter=0,no\_session=0,i,j,cnt=0,cnt1,cnt2,year1,check1,check2;//year[13000];

fp=fopen("random\_author\_pair\_4.txt","r");

fp1=fopen("venue\_noc.txt","w");

if(fp==NULL){

printf("Interaction list not found\n");

return 0;

}

ch1=fgetc(fp);

while(ch1!=EOF)

{

if(ch1=='\n')

no\_inter++;

ch1=fgetc(fp);

}

rewind(fp);

printf("\nTotal interactions to be checked %d\n",no\_inter);

fp2=fopen("venues\_NW\_1.txt","r");

ch2=fgetc(fp2);

while(ch2!=EOF)

{

if(ch2=='@')

no\_session++;

ch2=fgetc(fp2);

}

rewind(fp2);

printf("Total authors to be checked %d\n",no\_session);

for(i=0;i<no\_inter;i++)

{

fscanf(fp,"%ld",&auth1\_id);

fscanf(fp,"%ld",&auth2\_id);

fscanf(fp,"%d",&year1);

ch2=fgetc(fp2);

while(ch2!=EOF)

{

if(ch2=='@')

{

fscanf(fp2,"%ld",&auth3\_id);

}

if(auth1\_id==auth3\_id)

{

fprintf(fp1,"@%ld",auth1\_id);

ch2=fgetc(fp2);

while(ch2!='@')

{

fprintf(fp1,"%c",ch2);

ch2=fgetc(fp2);

}

fp3=fopen("venues\_NW\_1.txt","r");

ch3=fgetc(fp3);

while(ch3!=EOF)

{

if(ch3=='@')

{

fscanf(fp3,"%ld",&auth4\_id);

}

if(auth2\_id==auth4\_id)

{

fprintf(fp1,"@%ld",auth2\_id);

//check2=1;

ch3=fgetc(fp3);

while(ch3!='@')

{

fprintf(fp1,"%c",ch3);

ch3=fgetc(fp3);

}

break;

}

ch3=fgetc(fp3);

}

fclose(fp3);

break;

}

ch2=fgetc(fp2);

}

rewind(fp2);

printf("Interactions copied %d\n",i+1);

}}

**Same\_venue\_2.c**

#include<stdio.h>

#include<stdlib.h>

int main()

{

FILE \*fp,\*fp1,\*fp2,\*fp3;

char ch,ch1,ch2,ch3,num1,num2;

long int auth1\_id,auth2\_id,auth3\_id,auth4\_id;//auth[13000][2];

int conf\_year,no\_inter=0,no\_session=0,i,j,cnt=0,cnt1,cnt2,year1,check1,check2;//year[13000];

fp=fopen("random\_author\_pair\_4.txt","r");

fp1=fopen("venue\_noc\_new.txt","w");

if(fp==NULL){

printf("Interaction list not found\n");

return 0;

}

ch1=fgetc(fp);

while(ch1!=EOF)

{

if(ch1=='\n')

no\_inter++;

ch1=fgetc(fp);

}

rewind(fp);

printf("\nTotal interactions to be checked %d\n",no\_inter);

fp2=fopen("venues\_NW\_1.txt","r");

ch2=fgetc(fp2);

while(ch2!=EOF)

{

if(ch2=='@')

no\_session++;

ch2=fgetc(fp2);

}

rewind(fp2);

printf("Total authors to be checked %d\n",no\_session);

for(i=0;i<no\_inter;i++)

{

/\*fscanf(fp,"%ld",&auth[i][0]);

//printf("%ld",auth[i][0]);

fscanf(fp,"%ld",&auth[i][1]);

//printf(" %ld",auth[i][1]);

fscanf(fp,"%d",&year[i]);

//printf(" %d\n",year[i]);\*/

fscanf(fp,"%ld",&auth1\_id);

fscanf(fp,"%ld",&auth2\_id);

fscanf(fp,"%d",&year1);

ch2=fgetc(fp2);

while(ch2!=EOF)

{

if(ch2=='@')

{

fscanf(fp2,"%ld",&auth3\_id);

}

if(auth1\_id==auth3\_id)

{

fprintf(fp1,"@%ld",auth1\_id);

ch2=fgetc(fp2);

while(ch2!='@')

{

if(ch2==' ')

{

fprintf(fp1,"%c",ch2);

ch2=fgetc(fp2);

if(ch2=='1'||ch2=='2'||ch2=='3'||ch2=='4'||ch2=='5'||ch2=='6'||ch2=='7'||ch2=='8'||ch2=='9')

{

fprintf(fp1,"%c",ch2);

num1=ch2;

ch2=fgetc(fp2);

if(ch2==' ')

{

fprintf(fp1,"%c",ch2);

fseek(fp1,-3,SEEK\_CUR);

fprintf(fp1,"\*");

fprintf(fp1," !%c ",num1);

ch2=fgetc(fp2);

}

else if(ch2=='1'||ch2=='2'||ch2=='3'||ch2=='4'||ch2=='5'||ch2=='6'||ch2=='7'||ch2=='8'||ch2=='9')

{

fprintf(fp1,"%c",ch2);

num2=ch2;

ch2=fgetc(fp2);

if(ch2==' ')

{

fprintf(fp1,"%c",ch2);

fseek(fp1,-4,SEEK\_CUR);

fprintf(fp1,"\*");

fprintf(fp1," !%c%c ",num1,num2);

ch2=fgetc(fp2);

}}} }

fprintf(fp1,"%c",ch2);

ch2=fgetc(fp2);

}

fp3=fopen("venues\_NW\_1.txt","r");

ch3=fgetc(fp3);

while(ch3!=EOF)

{

if(ch3=='@')

{

fscanf(fp3,"%ld",&auth4\_id);

//check2=0;

}

if(auth2\_id==auth4\_id)

{

fprintf(fp1,"@%ld",auth2\_id);

//check2=1;

ch3=fgetc(fp3);

while(ch3!='@')

{

if(ch3==' ')

{

fprintf(fp1,"%c",ch3);

ch3=fgetc(fp3);

if(ch3=='1'||ch3=='2'||ch3=='3'||ch3=='4'||ch3=='5'||ch3=='6'||ch3=='7'||ch3=='8'||ch3=='9')

{

fprintf(fp1,"%c",ch3);

num1=ch3;

ch3=fgetc(fp3);

if(ch3==' ')

{

fprintf(fp1,"%c",ch3);

fseek(fp1,-3,SEEK\_CUR);

fprintf(fp1,"\*");

fprintf(fp1," !%c ",num1);

ch3=fgetc(fp3);

}

else if(ch3=='1'||ch3=='2'||ch3=='3'||ch3=='4'||ch3=='5'||ch3=='6'||ch3=='7'||ch3=='8'||ch3=='9')

{

fprintf(fp1,"%c",ch3);

num2=ch3;

ch3=fgetc(fp3);

if(ch3==' ')

{

fprintf(fp1,"%c",ch3);

fseek(fp1,-4,SEEK\_CUR);

fprintf(fp1,"\*");

fprintf(fp1," !%c%c ",num1,num2);

ch3=fgetc(fp3);

}}} }

fprintf(fp1,"%c",ch3);

ch3=fgetc(fp3);

}

break;

}

ch3=fgetc(fp3);

}

fclose(fp3);

break;

}

ch2=fgetc(fp2);

}

rewind(fp2);

printf("Interactions copied %d\n",i+1);

}}

**Flexi\_Session\_Layer.c**

#include <stdio.h>

#include <stdlib.h>

long int get\_first\_author(char \*paper) //search this paper in citation.txt and return index of its first author

{

FILE \*fp2;

char ch1,curr\_paper[2000],author[2000];

int index;

long int id;

fp2=fopen("citations.txt","r");

printf("called for %s\n",paper);

do

{

ch1=fgetc(fp2);

if(ch1=='#')

{

if((ch1=fgetc(fp2))=='\*')

{

index=0;

ch1=fgetc(fp2);

while(ch1!='\n'&&ch1!='.')

{

curr\_paper[index]=ch1;

index++;

ch1=fgetc(fp2);

}

curr\_paper[index]='\0';

if(strcmp(curr\_paper,paper)==0)

{

while(ch1!='@')

ch1=fgetc(fp2);

index=0;

ch1=fgetc(fp2);

while(ch1!=','&&ch1!='\n')

{

author[index]=ch1;

ch1=fgetc(fp2);

index++;

}

if(author[index-1]=='\r')

author[index-1]='\0';

author[index]='\0';

printf("first author %s found",author);

id=get\_author\_index(author);

fclose(fp2);

return id;

}

}

}

if(ch1!='\n' && ch1!=EOF)

{

ch1=fgetc(fp2);

while(ch1!='\n'&&ch1!=EOF)

ch1=fgetc(fp2);

}

}while(ch1!=EOF);

fclose(fp2);

printf("Paper not found...");

return -1;

}

int main()

{ FILE \*fp1,\*fp2,\*fp3,\*fp4;

char ch1,ch,conf\_list[40][400],names[800][8000],first[800][800],last[800][800],filename1[80],filename2[80],filename3[80],paper[2000];

int conf\_year,loopctr1,no\_conf=0,loopctr2,loopctr3,field\_author,count=0, already\_read\_paper\_name, index,l\_index,last\_comma\_position,start\_year,end\_year,last\_author\_flag[800],session\_presence,representative[800];

long int first\_id[800],last\_id[800];

float rand\_no,p1,p2;

count\_authors();

printf("Total no of authors %ld\n",no\_authors);

fp1=fopen("year\_conferences.txt","r");

if(fp1==NULL){

printf("conferene list not found\n");

return 0;

}

ch1=fgetc(fp1);

while(ch1!=EOF)

{

if(ch1=='\n')

no\_conf++;

ch1=fgetc(fp1);

}

rewind(fp1);

fscanf(fp1,"%d",&start\_year);

fscanf(fp1,"%d",&end\_year);

printf("Selected Start Year %d End Year %d\nSelected Conferences\n",start\_year,end\_year);

no\_conf--;

no\_conf--;

for(loopctr1=0;loopctr1<no\_conf;loopctr1++)

{

fscanf(fp1,"%s",conf\_list[loopctr1]);

printf("%s\n",conf\_list[loopctr1]);

}

fclose(fp1);

printf("Enter 0 if only FIRST authors come to present the papers\nEnter 1 if only LAST authors come to present the papers\nEnter 2 if anyone or both can come\nEnter your choice:");

scanf("%d",&session\_presence);

if(session\_presence == 2)

{

printf("Enter the probabilities with which either first author or second author will be chosen:");

scanf("%f%f",&p1,&p2);

printf("So with %f probability first author will be chosen, with %f probability last author will be chosen and with %f probability both will be chosen",p1,p2,1-p1-p2);

}

fp4=fopen("All\_interactions\_\_artint\_upto2007\_mixed\_0.45\_0.45\_0.1\_without\_workshops.txt","w");

for(conf\_year=start\_year;conf\_year<=end\_year;conf\_year++)

{

if(session\_presence==2)

{

sprintf(filename1,"./flexi\_artint\_session\_files/flexi\_session\_mixed\_0.45\_0.45\_0.1/session\_layer\_names\_%d.txt",conf\_year);

fp2=fopen(filename1,"w");

sprintf(filename2,"./flexi\_artint\_session\_files/flexi\_session\_mixed\_0.45\_0.45\_0.1/session\_layer\_id\_%d.txt",conf\_year);

fp3=fopen(filename2,"w");

}

else if(session\_presence==0)

{

sprintf(filename1,"./flexi\_artint\_session\_files/flexi\_session\_first\_author/session\_layer\_names\_%d.txt",conf\_year);

fp2=fopen(filename1,"w");

sprintf(filename2,"./flexi\_artint\_session\_files/flexi\_session\_first\_author/session\_layer\_id\_%d.txt",conf\_year);

fp3=fopen(filename2,"w");

}

else

{

sprintf(filename1,"./flexi\_artint\_session\_files/flexi\_session\_last\_author/session\_layer\_names\_%d.txt",conf\_year);

fp2=fopen(filename1,"w");

sprintf(filename2,"./flexi\_artint\_session\_files/flexi\_session\_last\_author/session\_layer\_id\_%d.txt",conf\_year);

fp3=fopen(filename2,"w");

}

for(loopctr1=0;loopctr1<no\_conf;loopctr1++)

{

sprintf(filename3,"./art\_int\_conference\_sessions/%s\_%d.txt",conf\_list[loopctr1],conf\_year);

fp1=fopen(filename3,"r");

if(fp1==NULL)

printf("\nData not found for: %s Conference of year %d\n",conf\_list[loopctr1],conf\_year);

else

{

ch=fgetc(fp1);

while(ch=='\n')

ch=fgetc(fp1);

while(ch!=EOF)

{

if(ch==' ')

{

field\_author=1;

for(loopctr2=0;loopctr2<4;loopctr2++)

ch=fgetc(fp1);

index=0;

while(ch!=':')

{

names[count][index]=ch;

ch=fgetc(fp1);

index++;

if(ch=='\n') //for session entries like "The 17th Heterogeneity in Computing Workshop (HCW 2008). 1-6"

{

count--;

break;

}

}

last\_author\_flag[count]=0;

names[count][index]='\0';

//printf("name %s\n",names[count]);

if(ch==':') //reading first and last authors from names

{

for(loopctr2=0;loopctr2<index;loopctr2++)

{

if(names[count][loopctr2]==',')//||names[count][loopctr2]==':')

break;

first[count][loopctr2]=names[count][loopctr2];

last[count][loopctr2]=names[count][loopctr2];

}

first[count][loopctr2]='\0';

last[count][loopctr2]='\0';

l\_index=0;

if(names[count][loopctr2]==',')

{

last\_author\_flag[count]=1;

last\_comma\_position=0;

for(loopctr3=0;loopctr3<index;loopctr3++)

{

if(names[count][loopctr3]==',')

last\_comma\_position=loopctr3;

}

for(loopctr3=last\_comma\_position+2;loopctr3<index;loopctr3++)

{

last[count][l\_index]=names[count][loopctr3];

l\_index++;

}

last[count][l\_index]='\0';

}

}

if(ch!='\n')

{

already\_read\_paper\_name=0;

first\_id[count]=get\_author\_index(first[count]);

if(first\_id[count]==-1 {

index=0;

ch=fgetc(fp1);

ch=fgetc(fp1);

while(ch!='\n'&&ch!='.')

{

paper[index]=ch;

index++;

ch=fgetc(fp1);

}

paper[index]='\0';

already\_read\_paper\_name=1;

printf("Required first author of Paper %s\n",paper);

first\_id[count]=get\_first\_author(paper);

}if(last\_author\_flag[count]==1)

{

last\_id[count]=get\_author\_index(last[count]);

if(last\_id[count]==-1) // author not found.... may be abbreviation used in citation data

{

if(already\_read\_paper\_name==0)

{

index=0;

ch=fgetc(fp1);

ch=fgetc(fp1);

while(ch!='\n'&&ch!='.')

{

paper[index]=ch;

index++;

ch=fgetc(fp1);

}

paper[index]='\0';

}

printf("Required last author of Paper %s\n",paper);

last\_id[count]=get\_last\_author(paper);

}

}

}

count++;

}

if(ch!='\n')

{ ch=fgetc(fp1);

while(ch!='\n' && ch!=EOF)

ch=fgetc(fp1);

}

ch=fgetc(fp1);

if(field\_author==1 && (ch!=' '||ch==EOF) )

{

field\_author=0;

printf("\n%s session with %d authors year %d",conf\_list[loopctr1],count,conf\_year);

if(session\_presence==0) //means only first author has come to present for each paper

{

for(loopctr2=0;loopctr2<count;loopctr2++)

{

for(loopctr3=loopctr2+1;loopctr3<count;loopctr3++)

{

if(first\_id[loopctr2]!=-1 && first\_id[loopctr3]!=-1 && first\_id[loopctr2]!=first\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",first[loopctr2],first[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",first\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",first\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

}

}

count=0;

}

else if(session\_presence==1) //means only last author has come to present for each paper

{

for(loopctr2=0;loopctr2<count;loopctr2++)

{

for(loopctr3=loopctr2+1;loopctr3<count;loopctr3++)

{

if(last\_id[loopctr2]!=-1 && last\_id[loopctr3]!=-1&& last\_id[loopctr2]!=last\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",last[loopctr2],last[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",last\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",last\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

}

}

count=0;

}

else //for each paper with probability p1 first author comes, with probability p2 last author comes and with probability 1-p1-p2 both comes

{

for(loopctr2=0;loopctr2<count;loopctr2++)

{

if(last\_author\_flag[loopctr2]==0)

representative[loopctr2]=0;

else

{

rand\_no=rand\_num();

//printf("rand %f\n\n",rand\_no);

if(rand\_no<p1)

representative[loopctr2]=0;

else if(rand\_no<(p1+p2))

representative[loopctr2]=1;

else

representative[loopctr2]=2;

}

//printf("chosen %d\n\n",representative[loopctr2]);

}

for(loopctr2=0;loopctr2<count;loopctr2++)

{

for(loopctr3=loopctr2+1;loopctr3<count;loopctr3++)

{

if(representative[loopctr2]==0 && representative[loopctr3]==0)

{

if(first\_id[loopctr2]!=-1 && first\_id[loopctr3]!=-1&& first\_id[loopctr2]!=first\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",first[loopctr2],first[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",first\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",first\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

}

if(representative[loopctr2]==0 && representative[loopctr3]==1)

{

if(first\_id[loopctr2]!=-1 && last\_id[loopctr3]!=-1&& first\_id[loopctr2]!=last\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",first[loopctr2],last[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",first\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",first\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

}

if(representative[loopctr2]==0 && representative[loopctr3]==2)

{

if(first\_id[loopctr2]!=-1 && first\_id[loopctr3]!=-1&& first\_id[loopctr2]!=first\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",first[loopctr2],first[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",first\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",first\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

if(first\_id[loopctr2]!=-1 && last\_id[loopctr3]!=-1&& first\_id[loopctr2]!=last\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",first[loopctr2],last[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",first\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",first\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

}

if(representative[loopctr2]==1 && representative[loopctr3]==0)

{

if(last\_id[loopctr2]!=-1 && first\_id[loopctr3]!=-1&& last\_id[loopctr2]!=first\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",last[loopctr2],first[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",last\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",last\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

}

if(representative[loopctr2]==1 && representative[loopctr3]==1)

{

if(last\_id[loopctr2]!=-1 && last\_id[loopctr3]!=-1&& last\_id[loopctr2]!=last\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",last[loopctr2],last[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",last\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",last\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

}

if(representative[loopctr2]==1 && representative[loopctr3]==2)

{

if(last\_id[loopctr2]!=-1 && first\_id[loopctr3]!=-1&& last\_id[loopctr2]!=first\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",last[loopctr2],first[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",last\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",last\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

if(last\_id[loopctr2]!=-1 && last\_id[loopctr3]!=-1&& last\_id[loopctr2]!=last\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",last[loopctr2],last[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",last\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",last\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

}

if(representative[loopctr2]==2 && representative[loopctr3]==0)

{

if(first\_id[loopctr2]!=-1 && first\_id[loopctr3]!=-1&& first\_id[loopctr2]!=first\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",first[loopctr2],first[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",first\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",first\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

if(last\_id[loopctr2]!=-1 && first\_id[loopctr3]!=-1&& last\_id[loopctr2]!=first\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",last[loopctr2],first[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",last\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",last\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

}

if(representative[loopctr2]==2 && representative[loopctr3]==1)

{

if(first\_id[loopctr2]!=-1 && last\_id[loopctr3]!=-1&& first\_id[loopctr2]!=last\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",first[loopctr2],last[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",first\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",first\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

if(last\_id[loopctr2]!=-1 && last\_id[loopctr3]!=-1&& last\_id[loopctr2]!=last\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",last[loopctr2],last[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",last\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",last\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

}

if(representative[loopctr2]==2 && representative[loopctr3]==2)

{

if(first\_id[loopctr2]!=-1 && first\_id[loopctr3]!=-1&& first\_id[loopctr2]!=first\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",first[loopctr2],first[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",first\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",first\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

if(first\_id[loopctr2]!=-1 && last\_id[loopctr3]!=-1&& first\_id[loopctr2]!=last\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",first[loopctr2],last[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",first\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",first\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

if(last\_id[loopctr2]!=-1 && first\_id[loopctr3]!=-1&& last\_id[loopctr2]!=first\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",last[loopctr2],first[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",last\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",last\_id[loopctr2],first\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}

if(last\_id[loopctr2]!=-1 && last\_id[loopctr3]!=-1&& last\_id[loopctr2]!=last\_id[loopctr3])

{

fprintf(fp2,"%s$ %s$ %s$ %d\n",last[loopctr2],last[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp3,"%ld %ld %s %d\n",last\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

fprintf(fp4,"%ld %ld %s %d\n",last\_id[loopctr2],last\_id[loopctr3],conf\_list[loopctr1],conf\_year);

}}}}count=0;}}}fclose(fp1); }}fclose(fp2);fclose(fp3);} fclose(fp4);}

**Same\_venue\_1.c**

#include<stdio.h>

#include<stdlib.h>

int main()

{

char conf1[130][200],conf2[130][200],ch;

int year1[130][130],year2[130][130],noy1[130],noy2[130],i,cnf1,cnf2,j,y1,y2,p1,p2,k,l,s,m,n,t,check1,check2,year3[30],q;

int check3,check4;

long int auth1\_id,auth2\_id;

FILE \*fp,\*fp1,\*fp2;

fp=fopen("venue\_noc\_new.txt","r");

fp1=fopen("venue\_noc\_new\_finite.txt","w");

fp2=fopen("noc\_publish.txt","w");

ch=fgetc(fp);

for(i=0;i<11734;i++)

{

/\*for(m=0;m<130;m++)

{

noy1[m]=0;noy2[m]=0;

}\*/

p1=0;p2=0;

fscanf(fp,"%ld",&auth1\_id);

printf("%ld",auth1\_id);

fscanf(fp,"%d",&cnf1);

for(j=0;j<cnf1;j++)

{

k=0;

ch=fgetc(fp);

ch=fgetc(fp);

while(ch!='\*')

{

conf1[j][k]=ch;

ch=fgetc(fp);

k++;

}

conf1[j][k]='\0';

//printf("%s",conf1[j]);

while(ch!='!')

{ch=fgetc(fp);}

fscanf(fp,"%d",&y1);

p1+=y1;

noy1[j]=y1;

for(l=0;l<y1;l++)

{

fscanf(fp,"%d",&year1[j][l]);

}

}

ch=fgetc(fp);

while(ch!='@')

{ch=fgetc(fp);}

fscanf(fp,"%ld",&auth2\_id);

printf(" %ld",auth2\_id);

fscanf(fp,"%d",&cnf2);

for(j=0;j<cnf2;j++)

{

k=0;

ch=fgetc(fp);

ch=fgetc(fp);

while(ch!='\*')

{

conf2[j][k]=ch;

ch=fgetc(fp);

k++;

}

conf2[j][k]='\0';

//printf("%s",conf1[j]);

while(ch!='!')

{ch=fgetc(fp);}

fscanf(fp,"%d",&y2);

p2+=y2;

noy2[j]=y2;

for(l=0;l<y2;l++)

{

fscanf(fp,"%d",&year2[j][l]);

}

}

ch=fgetc(fp);

while(ch!='@')

{ch=fgetc(fp);}

fprintf(fp2,"@%ld pub #%d @%ld pub #%d\n",auth1\_id,p1,auth2\_id,p2);

check1=0;check4=0;

for(m=0;m<cnf1;m++)

{

for(n=0;n<cnf2;n++)

{

if(strcmp(conf1[m],conf2[n])==0)

{

check2=0;

for(q=0;q<30;q++)

{year3[q]=0;}

for(s=0;s<noy1[m];s++)

{

for(t=0;t<noy2[n];t++)

{

if(year1[m][s]==year2[n][t])

{

check3=0;

for(q=0;year3[q]!=0;q++)

{

if(year3[q]==year1[m][s])

{

check3=1;

break;

}

}

if(check3==0)

year3[q]=year1[m][s];

if(check1==0&&check2==0&&check3==0)

{

fprintf(fp1,"\n\n@%ld @%ld\n\*%s !%d",auth1\_id,auth2\_id,conf1[m],year1[m][s]);

check2++;

check4=1;

}

else if(check1>=0&&check3==0&&check2>0)

{

fprintf(fp1," !%d",year1[m][s]);

check2++;

}

else if(check1>0&&check2==0&&check3==0)

{

fprintf(fp1,"\n\*%s !%d",conf1[m],year1[m][s]);

check2++;

}}}}}

}

if(check4==1)

check1++;

}

printf("\nRecords checked %d\n",i+1);}}

**Unique.py**

import collections

lines\_seen = set() # holds lines already seen

outfile = open("All\_interactions\_\_artint\_upto2007\_mixed\_0.45\_0.45\_0.1\_without\_workshops\_unique.txt", "w")

for line in open("All\_interactions\_\_artint\_upto2007\_mixed\_0.45\_0.45\_0.1\_without\_workshops.txt", "r"):

if line not in lines\_seen: # not a duplicate

outfile.write(line)

lines\_seen.add(line)

outfile.close()

y=collections.Counter(lines\_seen)

y