



BLM3021 Algoritma Analizi

Graf Üzerinde Arama İşlemi-Dönem Projesi

Doç. Dr. Elif KARSLIGİL

Halil İbrahim ULUOĞLU

16011093

1)KEVIN BACON ÖRNEKLERİ

```
Devam için 1 cikis için 0 giriniz
1
[MENU]Enter 2 if you want to choose Kevin Bacon comparision.
[MENU]Enter 3 if you want to choose different actors comparison.
2
Enter actor name,please. Cheong, June

The index of actor is: 206
Cheong, June 's Kevin Bacon number is 4
Cheong, June-Nielsen, Leslie : ...And Millions Die! (1973)
Nielsen, Leslie-Hong, James : Airplane! (1980)
Hong, James-Sutherland, Donald : Art of War, The (2000)
Sutherland, Donald-Bacon, Kevin : Animal House (1978)
Press any key to continue . . .
```

```
Devam için 1 cikis için 0 giriniz
1
[MENU]Enter 2 if you want to choose Kevin Bacon comparision.
[MENU]Enter 3 if you want to choose different actors comparison.
2
Enter actor name,please. Burmester, Leo

The index of actor is: 233
Burmester, Leo 's Kevin Bacon number is 2
Burmester, Leo-Little, Michele : Article 99 (1992)
Little, Michele-Bacon, Kevin : Apollo 13 (1995)
Press any key to continue . . .
```

```
C:\Users\h_ulu\OneDrive\Belgeler\3-1\Algoritma Analizi\Project\16011093.exe
Enter 1 for start.Enter 0 for stop,please.
1
[MENU]Enter 2 if you want to choose Kevin Bacon comparision.
[MENU]Enter 3 if you want to choose different actors comparison.
2
Enter actor name,please. Calfa, Don

The index of actor is: 309
Calfa, Don 's Kevin Bacon number is 2
Calfa, Don-Matheson, Tim : 1941 (1979)
Matheson, Tim-Bacon, Kevin : Animal House (1978)
Press any key to continue . . .
```

```
Devam icin 1 cikis icin 0 giriniz
1
[MENU]Enter 2 if you want to choose Kevin Bacon comparision.
[MENU]Enter 3 if you want to choose different actors comparison.
2
Enter actor name,please. Adkins, Seth

The index of actor is: 229
Adkins, Seth 's Kevin Bacon number is 3
Adkins, Seth-Janney, Allison : ...First Do No Harm (1997)
Janney, Allison-Short, John : Associate, The (1996)
Short, John-Bacon, Kevin : Apollo 13 (1995)
Press any key to continue . . . █
```

```
Devam icin 1 cikis icin 0 giriniz
1
[MENU]Enter 2 if you want to choose Kevin Bacon comparision.
[MENU]Enter 3 if you want to choose different actors comparison.
2
Enter actor name,please. Capron, Brian

The index of actor is: 616
Capron, Brian 's Kevin Bacon number is 3
Capron, Brian-Daniels, Jeff : 101 Dalmatians (1996)
Daniels, Jeff-Sinise, Gary : All the Rage (1999)
Sinise, Gary-Bacon, Kevin : Apollo 13 (1995)
Press any key to continue . . . █
```

```
Devam icin 1 cikis icin 0 giriniz
1
[MENU]Enter 2 if you want to choose Kevin Bacon comparision.
[MENU]Enter 3 if you want to choose different actors comparison.
2
Enter actor name,please. Blackett, Tony

The index of actor is: 748
Blackett, Tony 's Kevin Bacon number is 4
Blackett, Tony-Caton, Michael : 13th Floor, The (1988)
Caton, Michael-Truman, Jack : Animal, The (2001)
Truman, Jack-McGill, Bruce : Ali (2001)
McGill, Bruce-Bacon, Kevin : Animal House (1978)
Press any key to continue . . . █
```

```
Devam icin 1 cikis icin 0 giriniz
1
[MENU]Enter 2 if you want to choose Kevin Bacon comparision.
[MENU]Enter 3 if you want to choose different actors comparison.
2
Enter actor name,please. Danza, Tony

The index of actor is: 692
Danza, Tony 's Kevin Bacon number is 3
Danza, Tony-McConaughey, Matthew : Angels in the Outfield (1994)
McConaughey, Matthew-Berkeley, Xander : Amistad (1997)
Berkeley, Xander-Bacon, Kevin : Apollo 13 (1995)
Press any key to continue . . . █
```

```
Devam icin 1 cikis icin 0 giriniz
1
[MENU]Enter 2 if you want to choose Kevin Bacon comparision.
[MENU]Enter 3 if you want to choose different actors comparison.
2
Enter actor name,please. Cassidy, Sheila

The index of actor is: 310
Cassidy, Sheila 's Kevin Bacon number is 3
Cassidy, Sheila-Noble, James : 10 (1979)
Noble, James-Vernon, John : Airplane II: The Sequel (1982)
Vernon, John-Bacon, Kevin : Animal House (1978)
Press any key to continue . . .
```

```
Devam icin 1 cikis icin 0 giriniz
1
[MENU]Enter 2 if you want to choose Kevin Bacon comparision.
[MENU]Enter 3 if you want to choose different actors comparison.
2
Enter actor name,please. Little, Charles

The index of actor is: 138
Little, Charles 's Kevin Bacon number is 3
Little, Charles-Arquette, Rosanna : ...Almost (1990)
Arquette, Rosanna-Bloom, Verna : After Hours (1985)
Bloom, Verna-Bacon, Kevin : Animal House (1978)
Press any key to continue . . .
```

```
Devam icin 1 cikis icin 0 giriniz
1
[MENU]Enter 2 if you want to choose Kevin Bacon comparision.
[MENU]Enter 3 if you want to choose different actors comparison.
2
Enter actor name,please. Costa, Cosie

The index of actor is: 471
Costa, Cosie 's Kevin Bacon number is 3
Costa, Cosie-Billingslea, Beau : 10 to Midnight (1983)
Billingslea, Beau-Barry, Thom : American President, The (1995)
Barry, Thom-Bacon, Kevin : Apollo 13 (1995)
Press any key to continue . . .
```

2)İKİ FARKLI OYUNCU ÖRNEKLERİ

Devam için 1 cikis için 0 giriniz

1

[MENU]Enter 2 if you want to choose Kevin Bacon comparision.

[MENU]Enter 3 if you want to choose different actors comparision.

3

Enter first actor name,please. Harewood, Dorian

Enter second actor name,please. Angelini, Josephine

The index of first actor is: 695

The index of second actor is: 503

Harewood, Dorian 's Kevin Bacon number is 4

Harewood, Dorian-Rubinek, Saul : Against All Odds (1984)

Rubinek, Saul-Kusatsu, Clyde : And the Band Played On (1993)

Kusatsu, Clyde-DeBello, James : American Pie (1999)

DeBello, James-Angelini, Josephine : 100 Girls (2000)

Press any key to continue . . .

Devam için 1 cikis için 0 giriniz

1

[MENU]Enter 2 if you want to choose Kevin Bacon comparision.

[MENU]Enter 3 if you want to choose different actors comparision.

3

Enter first actor name,please. Council, Richard

Enter second actor name,please. Billman, Ange

The index of first actor is: 734

The index of second actor is: 504

Council, Richard 's Kevin Bacon number is 4

Council, Richard-McConaughey, Matthew : 13 Conversations About One Thing (2001)

McConaughey, Matthew-Levitt, Joseph : Angels in the Outfield (1994)

Levitt, Joseph-Oleynik, Larisa : 10 Things I Hate About You (1999)

Oleynik, Larisa-Billman, Ange : 100 Girls (2000)

Press any key to continue . . .

Devam için 1 cikis için 0 giriniz

1

[MENU]Enter 2 if you want to choose Kevin Bacon comparision.

[MENU]Enter 3 if you want to choose different actors comparision.

3

Enter first actor name,please. Balke, Turid

Enter second actor name,please. Bravo, Charly

The index of first actor is: 765

The index of second actor is: 571

Balke, Turid 's Kevin Bacon number is 4

Balke, Turid-Acheson, Mark : 13th Warrior, The (1999)

Acheson, Mark-Russo, Gianni : 3000 Miles to Graceland (2001)

Russo, Gianni-Brown, Jim : Any Given Sunday (1999)

Brown, Jim-Bravo, Charly : 100 Rifles (1969)

Press any key to continue . . .

Devam için 1 cikis için 0 giriniz

1

[MENU]Enter 2 if you want to choose Kevin Bacon comparision.

[MENU]Enter 3 if you want to choose different actors comparision.

3

Enter first actor name,please. Edwards, Glynn

Enter second actor name,please. Brown, Jim

The index of first actor is: 668

The index of second actor is: 278

Edwards, Glynn 's Kevin Bacon number is 3

Edwards, Glynn-Hawkins, Michael : 11 Harrowhouse (1974)

Hawkins, Michael-Stroud, Don : Amityville Horror, The (1979)

Stroud, Don-Brown, Jim : ...tick...tick...tick... (1970)

Press any key to continue . . .

```
Devam için 1 cikis için 0 giriniz
1
[MENU]Enter 2 if you want to choose Kevin Bacon comparision.
[MENU]Enter 3 if you want to choose different actors comparison.
3
Enter first actor name,please. Harewood, Dorian
Enter second actor name,please. Aron, Adrian

The index of first actor is: 695
The index of second actor is: 307
Harewood, Dorian 's Kevin Bacon number is 3
Harewood, Dorian-Valentine, Paul : Against All Odds (1984)
Valentine, Paul-Kiser, Virginia : All Night Long (1981)
Kiser, Virginia-Aron, Adrian : 10 (1979)
Press any key to continue . . .
```

```
Devam için 1 cikis için 0 giriniz
1
[MENU]Enter 2 if you want to choose Kevin Bacon comparision.
[MENU]Enter 3 if you want to choose different actors comparison.
3
Enter first actor name,please. Beck, Vincent
Enter second actor name,please. Close, Glenn

The index of first actor is: 156
The index of second actor is: 617
Beck, Vincent 's Kevin Bacon number is 3
Beck, Vincent-Nelson, Craig T. : ...And Justice for All (1979)
Nelson, Craig T.-Libby, Brian : Action Jackson (1988)
Libby, Brian-Close, Glenn : Air Force One (1997)
Press any key to continue . . .
```

```
Devam için 1 cikis için 0 giriniz
2
[MENU]Enter 2 if you want to choose Kevin Bacon comparision.
[MENU]Enter 3 if you want to choose different actors comparison.
3
Enter first actor name,please. Avari, Erick
Enter second actor name,please. Backlinie, Susan

The index of first actor is: 764
The index of second actor is: 1066
Avari, Erick 's Kevin Bacon number is 4
Avari, Erick-Jensen, Brian : 13th Warrior, The (1999)
Jensen, Brian-Carson, Paul : 6th Day, The (2000)
Carson, Paul-Cameron, Dave : Air Bud: World Pup (2000)
Cameron, Dave-Backlinie, Susan : 1941 (1979)
Press any key to continue . . .
```

```
C:\Users\h_u\OneDrive\Belgeler\3-1\Algoritma Analizi\Project\16011093.exe
Enter 1 for start.Enter 0 for stop,please.
1
[MENU]Enter 2 if you want to choose Kevin Bacon comparision.
[MENU]Enter 3 if you want to choose different actors comparison.
3
Enter first actor name,please. Attenborough, Richard
Enter second actor name,please. Hay, Alexandra

The index of first actor is: 338
The index of second actor is: 588
Attenborough, Richard 's Kevin Bacon number is 4
Attenborough, Richard-Burnham, Edward : 10 Rillington Place (1971)
Burnham, Edward-Franklyn, John : Abominable Dr. Phibes, The (1971)
Franklyn, John-ElDs, Sandor : And Soon the Darkness (1970)
ElDs, Sandor-Hay, Alexandra : 1000 Convicts and a Woman (1971)
Press any key to continue . . .
```

Devam için 1 cikis için 0 giriniz

1

[MENU]Enter 2 if you want to choose Kevin Bacon comparision.

[MENU]Enter 3 if you want to choose different actors comparison.

3

Enter first actor name,please. Edwards, Glynn

Enter second actor name,please. Arkin, Alan

The index of first actor is: 668

The index of second actor is: 731

Edwards, Glynn 's Kevin Bacon number is 4

Edwards, Glynn-Howard, Trevor : 11 Harrowhouse (1974)

Howard, Trevor-Firth, Peter : Aces High (1976)

Firth, Peter-McConaughey, Matthew : Amistad (1997)

McConaughey, Matthew-Arkin, Alan : 13 Conversations About One Thing (2001)

Press any key to continue . . .

Devam için 1 cikis için 0 giriniz

1

[MENU]Enter 2 if you want to choose Kevin Bacon comparision.

[MENU]Enter 3 if you want to choose different actors comparison.

3

Enter first actor name,please. Hunt, Paul

Enter second actor name,please. Fraser, Hugh

The index of first actor is: 753

The index of second actor is: 620

Hunt, Paul 's Kevin Bacon number is 5

Hunt, Paul-Caton, Michael : 13th Floor, The (1988)

Caton, Michael-Maples, Holly : Animal, The (2001)

Maples, Holly-Paquin, Anna : Almost Famous (2000)

Paquin, Anna-Daniels, Jeff : All the Rage (1999)

Daniels, Jeff-Fraser, Hugh : 101 Dalmatians (1996)

Press any key to continue . . .

3)KOD

```
16011093.c
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4
5  #define NUMBEROFLINES 750//14129
6  #define MOVIEBUFFER 500
7  #define NAMEBUFFER 150
8  #define LINEBUFFER 3800
9
10
11 char allMovies[NUMBEROFLINES][MOVIEBUFFER];
12 char **allActors;
13 int noneRepeatActor=0;
14
15 int createData();
16 int createFiles();
17 int readFiles();
18 struct graph *initilizationOfGraph(int vertex);
19 void addEdge(struct graph* aGraph, int vertex1,int edge1,int vertex2, int edge2);
20 struct node* creatingNode(int actorIDP,int movieIDP);
21 int detectActor(char *actorName);
22 void BFSALGORITHM(struct graph *aGraph, int actorIndex, int searchIndex);
23 struct queueEl *newQueue(struct node *aNode);
24 struct queue *initializationOfQueue();
25 int isEmpty(struct queue *aQueue);
26 void enqueue(struct queue *q, struct node *value);
27 struct queueEl *dequeue(struct queue *q);
28 int detectBacon();
29
30 //It keeps actorID as a node and movieID as a vertex.
31 struct node
32 {
33     int actorID;
34     int movieID;
35     int baconNumber;
36     struct node *next;
```

```
36     struct node *next;
37 };
38 //It keeps which actors cast which movie.
39 typedef struct
40 {
41     int *movies;
42     int movieNumber;
43     int actorID;
44 }ACTORStoMOVIE;
45 //It keeps which movie has role which actor.
46 typedef struct
47 {
48     int *actors;
49     int actorNumber;
50     int movieID;
51 }MOVIEtoACTORS;
52
53 ACTORStoMOVIE *actors_to_movie;
54 MOVIEtoACTORS movie_to_actors[NUMBEROFLINES];
55 //It keeps adjacency list,vertex number and visited nodes.
56 struct graph
57 {
58     struct node **adjacencyList;
59     int numberOfVertex;
60     int *visited;
61 };
62 //Elements of queue
63 typedef struct queueEl
64 {
65     struct node value;
66     struct queueEl *next;
67 }queueEl;
68 //Basic queue
69 struct queue
70 {
71     struct queueEl *front;
72     struct queueEl *rear;
```



```

74 //Option process, reading, creating, files and making bfs.
75 int main()
76 {
77     int i,j,k;
78     FILE *fp1;
79     int choice;
80     struct graph *aGraph;
81     int actorIndex,movieIndex;
82     struct node *aNode;
83     int control;
84     char *inputName=malloc(NAMEBUFFER*sizeof(char));
85     char *inputName2=malloc(NAMEBUFFER*sizeof(char));
86     int nameIndex,nameIndex2;
87     int deepChoise;
88
89
90     createData();
91     createFiles();
92     readFiles();
93
94     printf("Enter 1 for start.Enter 0 for stop,please.\n");
95     scanf("%d",&choice);
96
97     while(choice)
98     {
99         aGraph=initilizationOfGraph(noneRepeatActor);
100
101         for(i=0; i<noneRepeatActor; i++)
102         {
103             for(j=0; j<actors_to_movie[i].movieNumber; j++)
104             {
105                 movieIndex=actors_to_movie[i].movies[j];
106
107                 for(k=0; k<movie_to_actors[movieIndex].actorNumber; k++)
108                 {
109                     actorIndex=movie_to_actors[movieIndex].actors[k];
110                     if(actorIndex != i)

```

```

110                     if(actorIndex != i)
111                     {
112                         aNode = aGraph->adjacencyList[i];
113                         control=0;
114
115                         while((aNode != NULL) && (!control))
116                         {
117                             if(aNode->actorID == actorIndex)
118                             {
119                                 control=1;
120                             }
121                             aNode=aNode->next;
122                         }
123                         if(!control)
124                         {
125                             addEdge(aGraph, i, movieIndex, actorIndex, movieIndex);
126                         }
127                     }
128                 }
129             }
130         }
131
132         printf("[MENU]Enter 2 if you want to choose Kevin Bacon comparision.\n");
133         printf("[MENU]Enter 3 if you want to choose different actors comparison.\n");
134         scanf("%d", &deepChoise);
135
136         if(deepChoise==2)
137         {
138             fgets(inputName,NAMEBUFFER,stdin);
139             printf("Enter actor name,please. ");
140             scanf ("%[^\\n]*c", inputName);
141             nameIndex=detectActor(inputName);
142             printf("\nThe index of actor is: %d",nameIndex);
143             BFSALGORITHM(aGraph,detectBacon("Bacon, Kevin"),nameIndex);
144         }
145         else if(deepChoise==3)

```

```

145     else if(deepChoise==3)
146     {
147         fgets(inputName,NAMEBUFFER,stdin);
148         printf("Enter first actor name,please. ");
149         scanf ("%[^\\n]%c", inputName);
150         printf("Enter second actor name,please. ");
151         scanf ("%[^\\n]%c", inputName2);
152         nameIndex=detectActor(inputName);
153         nameIndex2=detectActor(inputName2);
154         printf("\\n\\nThe index of first actor is: %d",nameIndex);
155         printf("\\n\\nThe index of second actor is: %d",nameIndex2);
156         BFSALGORITHM(aGraph,detectBacon(inputName2),nameIndex);
157     }
158
159     printf("\\nDevam icin 1 cikis icin 0 giriniz \\n");
160     scanf("%d",&choise);
161 }
162
163 printf("\\n\\nKBacon's index is %d\\n",detectBacon());
164 for (i = 0; i < noneRepeatActor; i++)
165 {
166     free(allActors[i]);
167 }
168 free(allActors);
169
170 return 0;
171 }
172 //Taking movies, actors from main data input.
173 int createData()
174 {
175     FILE *graphFile,*inputFile;
176     int lineCounter;
177     char *line;
178     char *tempMovies;
179     char *tempActors;
180     int i=0,k=0;

```

```

181     int sizeofMovies=0;
182
183     allActors=(char **)malloc(sizeof(char *));
184     allActors[0]=(char *)malloc(NAMEBUFFER*sizeof(char));
185
186     graphFile=fopen("graph.txt","w");
187     if(graphFile==NULL)
188     {
189         printf("[ERROR-1]File can not opened.\\n");
190         exit(-1);
191     }
192     inputFile=fopen("input-mpaa.txt","rb");
193     if(inputFile==NULL)
194     {
195         printf("[ERROR-2]File can not opened.\\n");
196         exit(-2);
197     }
198
199     line=(char *)calloc(LINEBUFFER,sizeof(char));
200     for(lineCounter=0; lineCounter<NUMBEROFLINES; lineCounter++)
201     {
202         fgets(line,sizeof(char)*LINEBUFFER,inputFile);
203         line[strlen(line)-1]='\\0';
204         tempMovies=strtok(line,"/");
205
206         fprintf(graphFile, "%d", k++);
207         strcpy(allMovies[sizeofMovies++],tempMovies);
208         tempActors=strtok(NULL,"/");
209
210         while(tempActors != NULL)
211         {
212             i=0;
213             while((i<noneRepeatActor) && (strcmp(allActors[i],tempActors,NAMEBUFFER)) != 0)
214             {
215                 i++;

```

```

217     }
218     if(noneRepeatActor==i)
219     {
220         noneRepeatActor++;
221         allActors=realloc(allActors,sizeof(char *)*noneRepeatActor);
222         allActors[noneRepeatActor-1]=(char *)malloc(NAMEBUFFER*sizeof(char));
223         strcpy(allActors[noneRepeatActor-1],tempActors);
224         fprintf(graphFile, " %d", i);
225     }
226     else
227     {
228         fprintf(graphFile, " %d", i);
229     }
230     tempActors=strtok(NULL,"/");
231 }
232 fputc('\n',graphFile);
233 }
234 fclose(inputFile);
235 fclose(graphFile);
236 }
237
238 //Seperating actors and movies from main data file.
239 int createFiles()
240 {
241     FILE *movieFile,*actorFile;
242     int i;
243
244     movieFile=fopen("movies.txt","w");
245     if(movieFile==NULL)
246     {
247         printf("[ERROR-3!]File can not opened.\n");
248         exit(-3);
249     }
250
251     for(i=0; i<NUMBEROFLINES; i++)
252     {

```

```

253         fprintf(movieFile,"%d-%s\n",i,allMovies[i]);
254     }
255     fclose(movieFile);
256
257     actorFile=fopen("actors.txt","w");
258     if(actorFile==NULL)
259     {
260         printf("[ERROR-4!]File can not opened.\n");
261         exit(-4);
262     }
263
264     fprintf(actorFile,"%d\n",noneRepeatActor);
265     for(i=0; i<noneRepeatActor; i++)
266     {
267         fprintf(actorFile,"%d-%s\n",i,allActors[i]);
268     }
269     fclose(actorFile);
270
271     return 0;
272 }
273
274 //Reading movie and actors from my own create files.
275 int readFiles()
276 {
277     FILE *actorFile,*movieFile,*graphFile;
278     char *line,*temp;
279     int i,j=0,k;
280
281     actorFile=fopen("actors.txt", "r");
282     if(actorFile==NULL)
283     {
284         printf("[ERROR-5!]File can not opened.\n");
285         exit(-5);
286     }
287     line=calloc(LINEBUFFER,sizeof(char));
288     fscanf(actorFile,"%d",&noneRepeatActor);

```

```

288 fscanf(actorFile,"%d",&noneRepeatActor);
289 fgets(line,LINEBUFFER*sizeof(char),actorFile);
290 allActors=(char**)(char**)malloc(sizeof(char*)*noneRepeatActor);
291
292 for(i=0; i<noneRepeatActor; i++)
293 {
294     fgets(line,LINEBUFFER*sizeof(char),actorFile);
295     line[strlen(line)-1]='\0';
296
297     temp=strtok(line,"-");
298     temp=strtok(NULL,"-");
299
300     while(temp != NULL)
301     {
302         allActors[i] = (char*)malloc(NAMEBUFFER*sizeof(char));
303         strcpy(allActors[i],temp);
304         temp=strtok(NULL,"-");
305     }
306 }
307 fclose(actorFile);
308
309 movieFile=fopen("movies.txt","r");
310 if(movieFile==NULL)
311 {
312     printf("[ERROR-6]File can not opened.\n");
313     exit(-6);
314 }
315
316 for(i=0; i<NUMBEROFLINES; i++)
317 {
318     fgets(line,LINEBUFFER*sizeof(char),movieFile);
319     line[strlen(line)-1]='\0';
320
321     temp=strtok(line,"-");
322     temp=strtok(NULL,"-");
323
324     while(temp != NULL)

```

```

325     {
326         strcpy(allMovies[i],temp);
327         temp=strtok(NULL,"-");
328     }
329 }
330 fclose(movieFile);
331
332 graphFile=fopen("graph.txt","r");
333 if(graphFile==NULL)
334 {
335     printf("[ERROR-7]File can not opened.\n");
336     exit(-7);
337 }
338
339 for(i=0; i<NUMBEROFLINES; i++)
340 {
341     fgets(line,LINEBUFFER*sizeof(char),graphFile);
342     line[strlen(line)-1]='\0';
343     temp=strtok(line," ");
344
345     movie_to_actors[i].movieID=atoi(temp);
346     temp=strtok(NULL," ");
347     movie_to_actors[i].actors=(int *)calloc(1,sizeof(int));
348     j=0;
349     while(temp != NULL)
350     {
351         movie_to_actors[i].actors[j]=atoi(temp);
352         j++;
353         movie_to_actors[i].actors=(int *)realloc(movie_to_actors[i].actors,sizeof(int)*(j+1));
354         temp=strtok(NULL," ");
355     }
356
357     movie_to_actors[i].actors=realloc(movie_to_actors[i].actors,sizeof(int)*j);
358     movie_to_actors[i].actorNumber=j;
359 }

```

```

359     }
360     fclose(graphFile);
361
362     actors_to_movie=(ACTORStoMOVIE *)malloc(sizeof(ACTORStoMOVIE)*noneRepeatActor);
363     for(i=0; i<noneRepeatActor; i++)
364     {
365         actors_to_movie[i].movieNumber=0;
366         actors_to_movie[i].actorID=i;
367         actors_to_movie[i].movies=(int *)malloc(sizeof(int));
368     }
369
370     for(i=0; i<noneRepeatActor; i++)
371     {
372         for(j=0; j<NUMBEROFLINES; j++)
373         {
374             for(k=0; k<movie_to_actors[j].actorNumber; k++)
375             {
376                 if(movie_to_actors[j].actors[k] == i)
377                 {
378                     actors_to_movie[i].movies=realloc(actors_to_movie[i].movies,sizeof(int)*(actors_to_movie[i].movieNumber + 1));
379                     actors_to_movie[i].movies[actors_to_movie[i].movieNumber]=j;
380                     actors_to_movie[i].movieNumber++;
381                 }
382             }
383         }
384     }
385
386     return 0;
387 }
388 //Initializing graph for the first time.
389 struct graph *initilizationOfGraph(int vertex)
390 {
391     int i;
392     struct graph *aGraph;
393
394     aGraph=malloc(sizeof(struct graph));
395     aGraph->numberOfVertex=vertex;

```

```

395     aGraph->numberOfVertex=vertex;
396     aGraph->adjacencyList=malloc(sizeof(struct node *)*vertex);
397     aGraph->visited=malloc(vertex*sizeof(int));
398
399     for(i=0; i<vertex; i++)
400     {
401         aGraph->adjacencyList[i]=NULL;
402         aGraph->visited[i]=0;
403     }
404
405     return aGraph;
406 }
407 //Adding an edge to nodes.
408 void addEdge(struct graph* aGraph, int vertex1,int edge1,int vertex2, int edge2)
409 {
410     struct node *aNode = creatingNode(vertex2,edge2);
411     aNode->next=aGraph->adjacencyList[vertex1];
412     aGraph->adjacencyList[vertex1]=aNode;
413
414     aNode = creatingNode(vertex1,edge1);
415     aNode->next=aGraph->adjacencyList[vertex2];
416     aGraph->adjacencyList[vertex2]=aNode;
417 }
418 //Creating a new node first time.
419 struct node* creatingNode(int actorIDP,int movieIDP)
420 {
421     struct node *aNode = malloc(sizeof(struct node));
422     aNode->actorID=actorIDP;
423     aNode->movieID=movieIDP;
424     aNode->next=NULL;
425     aNode->baconNumber=0;
426
427     return aNode;
428 }
429 //Detecting actor.
430 int detectActor(char *actorName)
431 {

```

```

431 {
432     int i=0;
433
434     while(i<noneRepeatActor && strcmp(actorName,allActors[i]) != 0)
435     {
436         i++;
437     }
438     if(i==noneRepeatActor)
439     {
440         printf("[ERROR-8!]The actor could not find.\n");
441         exit(-8);
442     }
443     else
444     {
445         return i;
446     }
447 }
448 //Bfs algoirithm, keeping track of path.
449 void BFSALGORITHM(struct graph *aGraph, int actorIndex, int searchIndex)
450 {
451     int i,j,count=0,found=1;
452     int *track;
453
454     track=malloc(sizeof(int));
455     struct queue *aQueue=initializationOfQueue();
456
457     aGraph->visited[actorIndex]=-1;
458     struct node first;
459     first.actorID=actorIndex;
460     first.baconNumber=0;
461     first.next=NULL;
462
463     enqueue(aQueue,&first);
464
465     while(!isEmpty(aQueue))
466     {

```

```

467         struct queueEl *temp1 = dequeue(aQueue);
468         track=realloc(track,(count+1)*sizeof(int));
469         track[count++]=temp1->value.actorID;
470
471         struct node *aNode = aGraph->adjacencyList[temp1->value.actorID];
472
473         while(aNode)
474         {
475
476             if(aGraph->visited[aNode->actorID]==0)
477             {
478                 aNode->baconNumber = temp1->value.baconNumber+1;
479                 aGraph->visited[aNode->actorID]=aNode->baconNumber;
480                 enqueue(aQueue,aNode);
481             }
482
483             if(searchIndex==aNode->actorID && aGraph->visited[aNode->actorID] == aNode->baconNumber)
484             {
485                 found=0;
486                 printf("\n%s 's Kevin Bacon number is %d\n",allActors[aNode->actorID],aNode->baconNumber);
487                 int actorID1 = aNode->actorID;
488                 int actorBN1 =aNode->baconNumber;
489                 int actorID2;
490                 int actorBN2;
491
492                 struct node *temp;
493
494                 for(i=0; i<count; i++)
495                 {
496                     int found=0;
497                     int tmpID;
498                     actorID2=track[count-i-1];
499                     actorBN2=aGraph->visited[actorID2];
500                     struct node *temp = aGraph->adjacencyList[actorID2];
501                     if(actorBN2==(actorBN1-1) || (actorBN2==1))
502
503

```

```

503         if(actorBN2==(actorBN1-1) || (actorBN2==-1))
504         {
505             while(temp && !found)
506             {
507                 tmpID = temp->actorID;
508                 if(tmpID == actorID1)
509                 {
510                     printf("%s-%s : %s \n",allActors[actorID1],allActors[actorID2],allMovies[temp->movieID]);
511                     found=1;
512                     actorID1=actorID2;
513                     actorBN1=actorBN2;
514                 }
515                 temp=temp->next;
516             }
517         }
518         system("PAUSE");
519     }
520     aNode=aNode->next;
521 }
522 }
523 }
524 if(found)
525 {
526     printf("\nInfinite\n");
527     system("PAUSE");
528 }
529 }
530 //Creating queue first time.
531 struct queueEl *newQueue(struct node *aNode)
532 {
533     struct queueEl *aQueue;
534     aQueue=(struct queueEl *)malloc(sizeof(struct queueEl));
535     aQueue->value=*aNode;
536     aQueue->next=NULL;
537     return aQueue;
538 }
539 }
540 //Initializaing queue.
541 struct queue* initializationOfQueue()
542 {
543     struct queue *q=(struct queue *)malloc(sizeof(struct queue));
544     q->front=NULL;
545     q->rear=NULL;
546     return q;
547 }
548 }
549 //Check if queue is empty.
550 int isEmpty(struct queue *aQueue)
551 {
552     if(aQueue->front==NULL)
553     {
554         return 1;
555     }
556     else
557     {
558         return 0;
559     }
560 }
561 //Enqueue process
562 void enqueue(struct queue *q, struct node *value)
563 {
564     struct queueEl *aElement=newQueue(value);
565     aElement->value=*value;
566     aElement->next=NULL;
567     if (q->rear==NULL)
568     {
569         q->front=aElement;
570         q->rear=aElement;
571     }
572     else
573     {
574         q->rear->next=aElement;
575         q->rear=aElement;

```

```

576     }
577 }
578 //Dequeue process
579 struct queueEl *dequeue(struct queue *q)
580 {
581     if(q->front==NULL)
582     {
583         return NULL;
584     }
585
586     struct queueEl *temp = q->front;
587     q->front=q->front->next;
588     if(q->front== NULL)
589     {
590         q->rear=NULL;
591     }
592
593     return temp;
594 }
595 //Detecting Kevin Bacon function
596 int detectBacon(char *name)
597 {
598     int i=0;
599
600     while(i<noneRepeatActor && strcmp(name, allActors[i]) != 0)
601     {
602         i++;
603     }
604     if(i==noneRepeatActor)
605     {
606         return -1;
607     }
608     else
609     {
610         return i;
611     }
612 }

```