```cpp
 1 /*****************************************************************************
 2  * PROGRAMMER : Ali Eshghi
 3  * STUDENT ID : 1112261
 4  * CLASS      : CS1B
 5  * SECTION    : MW 7:30pm
 6  * Assign #3  : Searching linked list
 7  * DUE DATE   : 6 November 2019
 8  *****************************************************************************/
 9
10 #include "Header.h"
11
12 /*****************************************************************************
13  * Function - CreatList
14  * -----------------------------------------------------------------------
15  * This function will get the data from the input file and puts the data into
16  * the contents of the nodes, then add the nodes to the empty lists passed by
17  * Reference to the function
18  *
19  * return type - nothing
20  *                 the function is void type
21  *****************************************************************************/
22
23 void CreatList(MovieNode *&head)
24 {
25     /*************
26      * VARIABLES *
27      *************/
28
29     fstream     inFile;
30
31     //OPEN FILE
32
33     inFile.open("inFile.txt");
34
35     MovieNode *moviePtr;     //In & Calc  - node to input data
36     MovieNode *searchPtr;      //Calc      - node to go through the loop
37     MovieNode  node;         //In        - node to set the input into main
38                              //              main linked list
39
40     bool found;                  //Proc - lcv  varibale
41
42     /****************
43      * INITIALIZING *
44      ****************/
45
46     moviePtr = head;
47
48     /*****************************************************************************
49      * INPUT - Gets the data from the input file and puts them into the content
50      *         of the nodes in the linked list
51      *****************************************************************************/
52
53     while (inFile)
54     {
55         //FIRST INPUT
```

```
56
57          getline(inFile,node.title);
58          getline(inFile,node.leadActor);
59          getline(inFile,node.supportActor);
60          getline(inFile,node.genre);
61          getline(inFile,node.altGenre);
62          inFile >> node.year;
63          inFile.ignore(10000,'\n');
64          inFile >> node.rate;
65          getline(inFile,node.synopsis);
66          inFile.ignore(10000,'\n');
67          inFile.ignore(10000,'\n');
68
69
70          //CREATE A NEW NODE
71
72          moviePtr    = new MovieNode;
73          *moviePtr   = node;
74          searchPtr   = head;
75
76          found       = false;
77
78          //IF EMPTY
79          if (head == NULL)
80          {
81
82              moviePtr -> next = head;
83              moviePtr -> prev = NULL;
84
85              head            = moviePtr;
86
87              moviePtr        = NULL;
88
89          }
90
91          //ADDING IN TO THE FRONT
92
93          else if(head -> title > moviePtr -> title)
94          {
95
96              moviePtr -> prev  = NULL;
97              moviePtr -> next  = head;
98
99              head    -> prev   = moviePtr;
100             head              = moviePtr;
101
102             moviePtr          = NULL;
103
104         }
105
106         else
107         {
108             //COMPARING NODES TO SEE WHICH SHOULD GO WHERE
109
110             while ((searchPtr -> next != NULL) && !found)
```

```
111             {
112
113                 if (searchPtr -> next -> title > moviePtr -> title)
114                 {
115
116                     found = true;
117
118                 }
119
120                 else
121                 {
122
123                     searchPtr = searchPtr -> next;
124
125                 }
126             }
127
128             //TAIL
129
130             if(searchPtr -> next == NULL)
131             {
132
133                 moviePtr  -> next = NULL;
134                 moviePtr  -> prev = searchPtr;
135
136                 searchPtr -> next = moviePtr;
137
138                 moviePtr          = NULL;
139             }
140
141             //MIDDLE
142
143             else
144             {
145
146                 moviePtr -> next = searchPtr -> next;
147                 moviePtr -> prev = searchPtr;
148
149                 searchPtr -> next -> prev = moviePtr;
150                 searchPtr -> next = moviePtr;
151
152                 searchPtr = moviePtr = NULL;
153
154             }
155
156         } // END - ELSE
157
158     } // END - WHILE
159
160     //CLOSE FILE
161
162     inFile.close();
163
164 }
165
```