

SearchRate.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 - SearchRate
14  * -----
15  * This function will get a rating from the user and search the genre content of
16  * each node to see if there is a matching rating in the nodes with the rating
17  * searched by the user. if found, the function prints out every content of
18  * every node for the user, if not found, the function outputs that the rating
19  * searched by the user was not found.
20  *
21  * return type - nothing
22  *               the function is void type
23  *****/
24
25 void SearchRate(MovieNode *head, ofstream &outFile)
26 {
27     /*****
28      * VARIABLES *
29      *****/
30
31     MovieNode *movPtr; //Proc - stores the target name
32     MovieNode node;     //Proc & In - passes the info into persPtr
33
34
35     int i;                //In - counts the founded data
36
37     int checkInt;         //In - stores input value
38
39     bool checkInp;        //Proc - LCV value
40
41     /*****
42      * INITIALIZING *
43      *****/
44     checkInp = false;
45
46     i = 1;
47
48     /*****
49      * PROCESSING - searching for the rating in the list one by one and output
50      *               each movie with the matching rating to the output file
51      *****/
52
53
54     if (head == NULL)
55     {
```

SearchRate.cpp

```

56         cout << endl;
57
58         cout << "Can not search for a rating in an empty list.";
59
60         cout << endl;
61     }
62
63     else
64     {
65         //creating new node
66
67         movPtr = new MovieNode;
68
69         //INPUT
70
71         cout << endl;
72
73         do
74         {
75             //INPUT
76
77             cout << "\nWhich rating are you looking for? ";
78
79             //CHECKS FOR THE CHAR INPUT
80
81             if (!(cin >> checkInt))
82             {
83
84                 cin.clear();
85                 cin.ignore(numeric_limits<streamsize>::max(), '\n');
86
87                 cout << endl;
88                 cout << "**** Please input a NUMBER between 0 and 9 ****";
89                 cout << endl;
90
91                 checkInp = false;
92
93             }
94
95             //CHECKS FOR THE RANGE ERROR
96
97             else if (checkInt >= 10 || checkInt <= -1 )
98             {
99
100
101                 cout << endl;
102                 cout << "**** The number " << node.rate
103                     << " is an invalid entry ****" << endl;
104                 cout << "**** Please input a number between 0 and 9 ****";
105                 cout << endl;
106
107                 checkInp = false;
108
109             }
110

```

SearchRate.cpp

```

111         //PASS
112
113         else
114         {
115
116             cin.ignore(numeric_limits<streamsize>::max(), '\n');
117             checkInp = true;
118
119         }
120
121     }while(!checkInp);
122
123     node.rate = checkInt;
124
125     *movPtr = node;
126
127     cout << endl;
128
129     cout << "Searching for the rating";
130
131     cout << endl;
132
133     outFile << "\nSearch by rating for " << movPtr -> rate << " found:";
134
135     outFile << endl << endl;
136
137     outFile << left;
138
139     outFile << setw(9) << "MOVIE #" << " ";
140     outFile << setw(48) << "TITLE" << " ";
141     outFile << setw(5) << "YEAR" << " ";
142     outFile << setw(8) << "RATING" << " ";
143     outFile << setw(18) << "GENRE" << " ";
144     outFile << setw(18) << "ALT GENRE" << " ";
145     outFile << setw(20) << "LEAD ACTOR" << " ";
146     outFile << setw(20) << "SUPPORTING ACTOR" << " ";
147
148     outFile << left;
149
150     outFile << endl;
151
152     outFile << "----- ";
153     outFile << "----- ";
154     outFile << "---- ";
155     outFile << "----- ";
156     outFile << "----- ";
157     outFile << "----- ";
158     outFile << "----- ";
159     outFile << "----- ";
160     outFile << endl;
161
162     //GOES THROUGH THE LIST
163
164     while(head -> next != NULL)
165     {

```

SearchRate.cpp

```

166     if (head -> rate == movPtr -> rate)
167     {
168
169         outFile << right;
170         outFile << setw(4) << i << "      "      ;
171         outFile << left;
172         outFile << setw(47) << head -> title      ;
173         outFile << setw(8) << head -> year      ;
174         outFile << setw(5) << head -> rate      ;
175         outFile << setw(18) << head -> genre      ;
176         outFile << setw(18) << head -> altGenre      ;
177         outFile << setw(20) << head -> leadActor      ;
178         outFile << setw(20) << head -> supportActor;
179         outFile << endl;
180         outFile << left;
181
182         i++;
183         head = head -> next;
184
185     }
186
187     else
188     {
189         head = head -> next;
190     }
191 } // END - WHILE
192
193
194
195 if (head -> next == NULL && (head -> rate == movPtr -> rate))
196 {
197
198     outFile << right;
199     outFile << setw(4) << i << "      "      ;
200     outFile << left;
201     outFile << setw(47) << head -> title      ;
202     outFile << setw(8) << head -> year      ;
203     outFile << setw(5) << head -> rate      ;
204     outFile << setw(18) << head -> genre      ;
205     outFile << setw(18) << head -> altGenre      ;
206     outFile << setw(20) << head -> leadActor      ;
207     outFile << setw(20) << head -> supportActor;
208     outFile << endl;
209     outFile << left;
210     i++;
211     head = head -> next;
212
213 }
214
215 //IF NOT FOUND IN THE LIST
216
217 if (i == 1)
218 {
219     cout << "Sorry, no movies for the rating, " << movPtr -> rate
220         << " were found.";

```

SearchRate.cpp

```
221
222     cout << endl;
223 }
224
225 //OUTPUT RESULT INTO THE CONSOLE
226
227 if(i > 1)
228 {
229     cout << "Found " << i-1 << " movies for the rating "
230         << movPtr -> rate << '!' << endl;
231 }
232
233 movPtr = NULL;
234 delete movPtr;
235
236 }
237
238 }
239
```