**SEPARATE CHAINING:**

1. #include<iostream>
2. #include<cstdlib>
3. using namespace std;
4. struct student
5. {
6. int rno;
7. char name[20];
8. student \*link;
9. }\*head,\*temp,\*temp1,\*p;
10. struct node
11. {
12. int key;
13. student \*slink;
14. node \*next;
15. }\*t,\*prev;
16. node \*hash[20];
17. int n=7;
18. void del1(student \*n)
19. {
20. temp=head;
21. while(temp->rno!=n->rno)
22. {
23. p=temp;
24. temp=temp->link;
25. }
26. if(temp==head)
27. head=head->link;
28. else if(temp->link=='\0')
29. p->link='\0';
30. else
31. {
32. p->link=temp->link;
33. temp->link='\0';
34. }
35. }
36. void del()
37. {
38. int no;
39. cout<<"\n enter roll no to be deleted:";
40. cin>>no;
41. int k=no%7;
42. t=hash[k];
43. student \*n=new student;
44. while(t!='\0')
45. {
46. if(t->key==no)
47. {
48. if(t==hash[k] && t->next=='\0')
    1. hash[k]='\0';
49. else if(t==hash[k] && t->next!='\0')
    1. hash[k]=t->next;
50. else if(t->next=='\0')
51. prev->next='\0';
52. else
53. {
54. prev->next=t->next;
55. t->next='\0';
56. }
57. }
58. if(t->key==no)
59. break;
60. prev=t;
61. t=t->next;
62. }
63. n=t->slink;
64. //t->next='\0';
65. //delete t;
66. del1(n);
67. }
68. void insert()
69. {
70. cout<<"\n enter student details:";
71. student \*s=new student;
72. s->link='\0';
73. cout<<"\n enter rno:";
74. cin>>s->rno;
75. cout<<"\n enter name:";
76. cin>>s->name;
77. if(head=='\0')
78. {
79. head=s;
80. temp1=s;
81. }
82. else
83. {
84. temp1->link=s;
85. temp1=s;
86. }
87. cout<<"\n inserted into the list:";
88. int k;
89. k=(s->rno)%7;
90. node \*n=new node;
91. n->key=s->rno;
92. n->slink=s;
93. n->next='\0';
94. if(hash[k]=='\0')
95. hash[k]=n;
96. else
97. {
98. t=hash[k];
99. while(t->next!='\0')
100. t=t->next;
101. t->next=n;
102. }
103. cout<<"\n inserted into hash table:";
104. }
105. void displayhash()
106. {
107. for(int i=0;i<7;++i)
108. {
109. t=hash[i];
110. cout<<i<<"->";
111. while(t!='\0')
112. {
113. cout<<t->key<<" ";
114. t=t->next;
115. }
116. cout<<"\n";
117. }
118. }
119. void displaylist()
120. {
121. temp=head;
122. while(temp!='\0')
123. {
124. cout<<temp->rno<<" "<<temp->name;
125. temp=temp->link;
126. cout<<"\n";
127. }
128. }
129. void search()
130. {
131. int key,x,flag=0;
132. cout<<"\n enter the rno to be searched:";
133. cin>>key;
134. x=key%7;
135. t=hash[x];
136. while(t!='\0')
137. {
138. if(t->key==key)
139. {
140. flag=1;
141. break;
142. }
143. t=t->next;
144. }
145. if(flag==0)
146. cout<<"\n student not found:";
147. else
148. {
149. cout<<"\n Roll no:"<<t->slink->rno;
150. cout<<"\n Name:"<<t->slink->name;
151. }
152. }
153. int main()
154. {
155. int ch;
156. do
157. {
158. cout<<"\n1.insert \n2.delete \n3.display list \n4.display hash table \n5.search \n6.exit:";
159. cin>>ch;
160. switch(ch)
161. {
162. case 1:insert();
163. break;
164. case 2:del();
165. break;
166. case 3:displaylist();
167. break;
168. case 4:displayhash();
169. break;
170. case 5:search();
171. break;
172. case 6:exit(0);
173. }
174. }while(ch!=6);
175. return 0;
176. }