**SPLAY TREES:**

1. #include<iostream>
2. using namespace std;
3. struct node
4. {
5. int key;
6. node \*l,\*r;
7. }\*root='\0',\*head;
8. node \*rightrotate(node \*x)
9. {
10. node \*y=x->l;
11. x->l=y->r;
12. y->r=x;
13. return y;
14. }
15. node \*leftrotate(node \*x)
16. {
17. node \*y=x->r;
18. x->r=y->l;
19. y->l=x;
20. return y;
21. }
22. node \*splay(node \*root,int key)
23. {
24. if(root=='\0'|| root->key==key)
25. return root;
26. if(root->key>key)
27. {
28. if(root->l=='\0')
29. return root;
30. if(root->l->key>key)
31. {
32. root->l->l=splay(root->l->l,key);
33. root=rightrotate(root);
34. }
35. else if(root->l->key<key)
36. {
37. root->l->r=splay(root->l->r,key);
38. if(root->l->r!='\0')
39. root->l=leftrotate(root->l);
40. }
41. if(root->l=='\0')
42. return root;
43. else
44. return rightrotate(root);
45. }
46. else
47. {
48. if(root->r=='\0')
49. return root;
50. if(root->r->key>key)
51. {
52. root->r->l=splay(root->r->l,key);
53. if(root->r->l!='\0')
54. root->r=rightrotate(root->r);
55. }
56. else if(root->r->key<key)
57. {
58. root->r->r=splay(root->r->r,key);
59. root=leftrotate(root);
60. }
61. if(root->r=='\0')
62. return root;
63. else
64. leftrotate(root);
65. }
66. }
67. void form(node \*n)
68. {
69. if(n->key > root->key)
70. {
71. if(root->r=='\0')
72. root->r=n;
73. else
74. {
75. root=root->r;
76. form(n);
77. }
78. }
79. else if(n->key <= root->key)
80. {
81. if(root->l=='\0')
82. root->l=n;
83. else
84. {
85. root=root->l;
86. form(n);
87. }
88. }
89. }
90. void preorder(node \*root)
91. {
92. cout<<root->key<<" ";
93. if(root->l!='\0')
94. preorder(root->l);
95. if(root->r!='\0')
96. preorder(root->r);
97. }
98. int main()
99. {
100. int key;
101. char ch;
102. node \*n=new node;
103. n->r='\0';
104. n->l='\0';
105. cout<<"\n enter root:";
106. cin>>n->key;
107. root=n;
108. head=root;
109. do
110. {
111. root=head;
112. cout<<"\n continue?";
113. cin>>ch;
114. if(ch=='y')
115. {
116. node \*n=new node;
117. n->r='\0';
118. n->l='\0';
119. cout<<"\n enter data:";
120. cin>>n->key;
121. form(n);
122. }
123. }while(ch!='n');
124. root=head;
125. cout<<"\n preorder is:";
126. preorder(root);
127. root=head;
128. cout<<"\n enter the key:";
129. cin>>key;
130. root=splay(root,key);
131. cout<<"\n after splaying the preorder is:";
132. preorder(root);
133. }