Open | test19.cpp ~/Documents/FDS | Save

```cpp
# include<stdio.h>
# include <iostream>
# include<string>
using namespace std;
class list;
class node
{
        int prn;
        string name;
        node *next;
        public:
                node(int x, string nm)
                {
                        prn = x;
                        next = NULL;
                        name = nm;
                }
        friend class list;
};
class list
{
        node * start;
        public:
        list()
        {
                start = NULL;
        }
        void create();
        void display();
        void insertAtBeginning();
        void insertAtEnd();
        void insertAfter();
        void deleteAtFirst();
        void deleteByValue();
        void deleteAtEnd();
        int computeTotal();
        void sortList();
        void concatList(list & q1);
        void displayRev(node * t);
```

C++ | Tab Width: 8 | Ln 375, Col 2 | INS

Open | test19.cpp ~/Documents/FDS | Save

```cpp
        void displayRev(node * t);
        bool reverseDisplay() // function is only for passing start as argument to recursive function
        {
                if (start == NULL)
                        return false;
                        node * temp = start;
                        displayRev(temp);
                        return true;
        }
};
void list::displayRev(node * t)
{
        if (t == NULL)
                return;
        else
        {
                displayRev(t->next);
                cout << "\nPRN NO:" << t->prn << " Name: " << t->name;
        }
}
void list::create()
{
        int no;
        string nam;
        if (start == NULL)
        {
                cout << "Enter PRN number: ";
                cin >> no;
                cout << "Enter name: ";
                cin >> nam;
                cout << nam;
                start = new
                node(no, nam);
                cout << "\n================ List Created ================";
        }
        else
        {
        cout << "\nList is already created.";
        }
```

C++ | Tab Width: 8 | Ln 77, Col 1 | INS

Open | test19.cpp ~/Documents/FDS | Save

```
}
void list::display()
{
        node * t;
        t = start;
        if (start == NULL)
                cout << "\nList is Empty";
        else
                {cout << "\n=============== List: ===============\n";
                while (t != NULL){
                        cout << t->prn << " " << t->name << " \n";
                        t=t->next;
                }
}
}
void list::insertAtBeginning()
{
        int no;
        string nam;
        node * temp;
        if (start == NULL)
        {
                create();
        }
        else
        {
                cout << "\nEnter PRN Number : ";
                cin >> no;
                cout << "Enter Name : ";
                cin >> nam;
                temp = new
                node(no, nam);
                temp->next = start;
                start = temp;
                cout << "Inserted " << temp->name << " at the beginning.";
        }
}
void list::insertAtEnd()
{
```

C++ | Tab Width: 8 | Ln 77, Col 1 | INS

Open | test19.cpp ~/Documents/FDS | Save

```
void list::insertAtEnd()
{
        int no;
        string nam;
        node * t;
        if (start == NULL)
                create();
        else
        {
                cout << "\nEnter PRN Number : ";
                cin >> no;
                cout << "Enter Name : ";
                cin >> nam;
                t = start;
                while (t->next != NULL)
                        t = t->next;
                        node * p = new
                        node(no, nam);
                        t->next = p;
        }
}

void list::insertAfter()
{
        int prev_no;
        cout << "\nEnter PRN No. after do you want insert : ";
        cin >> prev_no;
        node * t;
        t = start;
        string nam;
        int flag = 0, no;
        while (t != NULL)
        {
                if (t->prn == prev_no)
                {
                        flag = 1;
                        break;
                }
                t = t->next;
```

C++ | Tab Width: 8 | Ln 153, Col 1 | INS

Open | test19.cpp ~/Documents/FDS | Save

```
153                 t = t->next;
154         }
155         if (flag == 1)
156         {
157                 node * p;
158                 cout << "\nEnter PRN Number : ";
159                 cin >> no;
160                 cout << "Enter Name : ";
161                 cin >> nam;
162                 p=new node(no, nam);
163                 p->next=t->next;
164                 t->next=p;
165         }
166         else
167         {
168                 cout << "\n" << prev_no << " is not in list.";
169         }
170 }
171 void list::deleteAtFirst()
172 {
173         node * t;
174         if (start == NULL)
175                 cout << "\nClub is Empty..";
176         else
177         {
178                 t=start;
179                 start=start->next;
180                 t->next=NULL; // Not necessary
181                 delete t;
182                 cout << "\nPresident deleted..";
183         }
184 }
185 void list::deleteByValue()
186 {
187         int no, flag = 0;
188         node * t, *prev;
189         if (start == NULL)
190                 cout << "\nList/Club is empty";
191         else
```

C++ | Tab Width: 8 | Ln 191, Col 1 | INS

Open | test19.cpp ~/Documents/FDS | Save

```
191         else
192         {
193                 cout << "\nEnter PRN No. of member to be deleted : ";
194                 cin >> no;
195                 t=start->next;
196                 while (t->next != NULL)
197                 {
198                         if (t->prn == no)
199                         {
200                                 flag = 1;
201                                 break;
202                         }
203                         prev = t;
204                         t = t->next;
205                 }
206                 if (flag == 1)
207                 {
208                         prev->next=t->next;
209                         t->next=NULL;
210                         delete t;
211                         cout << "\nMember with PRN No: " << no << " is deleted.";
212                 }
213                 else
214                         cout << "\nMember not found in List./President or Secretary cannot be deleted.";
215         }
216 }
217 void list::deleteAtEnd()
218 {
219         node * t, *prev;
220         t = start;
221         if (start == NULL)
222                 cout << "\nClub is Empty..";
223         else
224         {
225                 while (t->next != NULL)
226                 {
227                         prev = t;
228                         t = t->next;
229                 }
```

C++ | Tab Width: 8 | Ln 229, Col 1 | INS

Open | test19.cpp ~/Documents/FDS | Save

```
229                 }
230                 prev->next = NULL;
231                 delete t;
232                 cout << "\nSecretary Deleted.";
233         }
234 }
235 int list::computeTotal()
236 {
237         node * t;
238         int count = 0;
239         t = start;
240         if (start == NULL)
241         {
242                 cout << "\nList is empty.";
243                 return 0;
244         }
245         while (t != NULL)
246         {
247                 count ++;
248                 t = t->next;
249         }
250         return count;
251 }
252 void list::sortList()
253 {
254         node * i, *j, *last = NULL;
255         int tprn;
256         string tname;
257         if (start == NULL)
258         {
259                 cout << "\nList is empty.";
260                 return;
261         }
262         for (i=start;i->next != NULL;i=i->next)
263         {
264                 for (j=start;j->next != last;j=j->next)
265                 {
266                         if ((j->prn) > (j->next->prn))
267                         {
```

C++ | Tab Width: 8 | Ln 267, Col 1 | INS

Open | test19.cpp ~/Documents/FDS | Save

```
266                         if ((j->prn) > (j->next->prn))
267                         {
268                                 tprn = j->prn;
269                                 tname = j->name;
270                                 j->prn = j->next->prn;
271                                 j->name = j->next->name;
272                                 j->next->prn = tprn;
273                                 j->next->name = tname;
274                         }
275                 }
276         }
277         cout << "\n List is sorted.";
278         display();
279 }
280 void list::concatList(list & q1)
281 {
282         node * t, *p;
283         t = q1.start;
284         if (t == NULL)
285         {
286                 cout << "\nList 2 is empty";
287                 return;
288         }
289         p = start; // first list
290         while (p->next != NULL)
291         {
292                 p = p->next;
293         }
294         p->next = t;
295         q1.start = NULL; // second list is set to null
296         cout << "\nAfter concatenation list : \n";
297         display();
298 }
299 int main()
300 {
301         list * l;
302         int choice, selectList;
303         list l1, l2;
304         l = & l1;
```

C++ | Tab Width: 8 | Ln 304, Col 1 | INS

Open test19.cpp ~/Documents/FDS Save

```
305         X: cout << "\nSelect List\n1.List 1\n2.List 2\nEnter choice : ";
306         cin >> selectList;
307         if (selectList == 1)
308         {
309                 l = & l1;
310         }
311         else if (selectList == 2)
312         {
313                 l = & l2;
314         }
315         else
316         {
317                 cout << "\nWrong list Number.";
318                 goto X;
319         }
320         do
321         {
322                 cout << "\n1. Create\n2. Insert President\n3. Insert secretary\n4. Insert after position(member)\n";
323                 cout<<"5. Display list\n6. Delete President\n7.Delete Secretary\n8. Delete Member";
324                 cout<<"\n9. Find total No. of   members\n10. Sort list\n11. Reselect List";
325                 cout<< "\n12. Combine lists\n13.Reverse Display\n0. Exit\nEnter your choice :\t";
326                 cin >> choice;
327                 switch(choice)
328                 {
329                         case 1:
330                                 l->create();
331                                 break;
332                         case 2:
333                                 l->insertAtBeginning();
334                                 break;
335                         case 3:
336                                 l->insertAtEnd();
337                                 break;
338                         case 4:
339                                 l->insertAfter();
340                                 break;
341                         case 5:
342                                 l->display();
343                                 break;
```

C++ Tab Width: 8 Ln 343, Col 1 INS

Open test19.cpp ~/Documents/FDS Save

```
337                                 break;
338                         case 4:
339                                 l->insertAfter();
340                                 break;
341                         case 5:
342                                 l->display();
343                                 break;
344                         case 6:
345                                 l->deleteAtFirst();
346                                 break;
347                         case 7:
348                                 l->deleteAtEnd();
349                                 break;
350                         case 8:
351                                 l->deleteByValue();
352                                 break;
353                         case 9:
354                                 cout << "\nTotal members(including President & Secretary) : " << l->computeTotal();
355                                 break;
356                         case 10:
357                                 l->sortList();
358                                 break;
359                         case 11:
360                                 goto X;
361                                 break;
362                         case 12:
363                                 l1.concatList(l2);
364                                 break;
365                         case 13:
366                                 l->reverseDisplay();
367                                 break;
368                         deafult:
369                                 cout << "Wrong choice";
370                 }
371         }
372         while (choice != 0);
373                 cout << "\n=============== GOOD BYE ===============\n";
374         return 0;
375 }
```

C++ Tab Width: 8 Ln 343, Col 1 INS

```
onkar@ubuntu: ~/Documents/FDS
onkar@ubuntu:~$ cd Documents/FDS
onkar@ubuntu:~/Documents/FDS$ g++ test19.cpp
onkar@ubuntu:~/Documents/FDS$ ./a.out

Select List
1.List 1
2.List 2
Enter choice : 1

1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
Enter your choice :     1
Enter PRN number: 123
Enter name: kunal
kunal
============== List Created ==============
1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
```

```
onkar@ubuntu: ~/Documents/FDS
13.Reverse Display
0. Exit
Enter your choice :     2

Enter PRN Number : 70
Enter Name : raj
Inserted raj at the beginning.
1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
Enter your choice :     3

Enter PRN Number : 150
Enter Name : tejas

1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
```

```
onkar@ubuntu: ~/Documents/FDS
Enter your choice :     4

Enter PRN No. after do you want insert : 123

Enter PRN Number : 124
Enter Name : shekhar

1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
Enter your choice :     11

Select List
1.List 1
2.List 2
Enter choice : 2

1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
```

```
onkar@ubuntu: ~/Documents/FDS
13.Reverse Display
0. Exit
Enter your choice :     1
Enter PRN number: 121
Enter name: laxmi
laxmi
=============== List Created ===============
1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
Enter your choice :     2

Enter PRN Number : 56
Enter Name : sarika
Inserted sarika at the beginning.
1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
```

```
onkar@ubuntu: ~/Documents/FDS
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
Enter your choice :     3

Enter PRN Number : 149
Enter Name : sanika

1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
Enter your choice :     4

Enter PRN No. after do you want insert : 122

122 is not in list.
1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
```

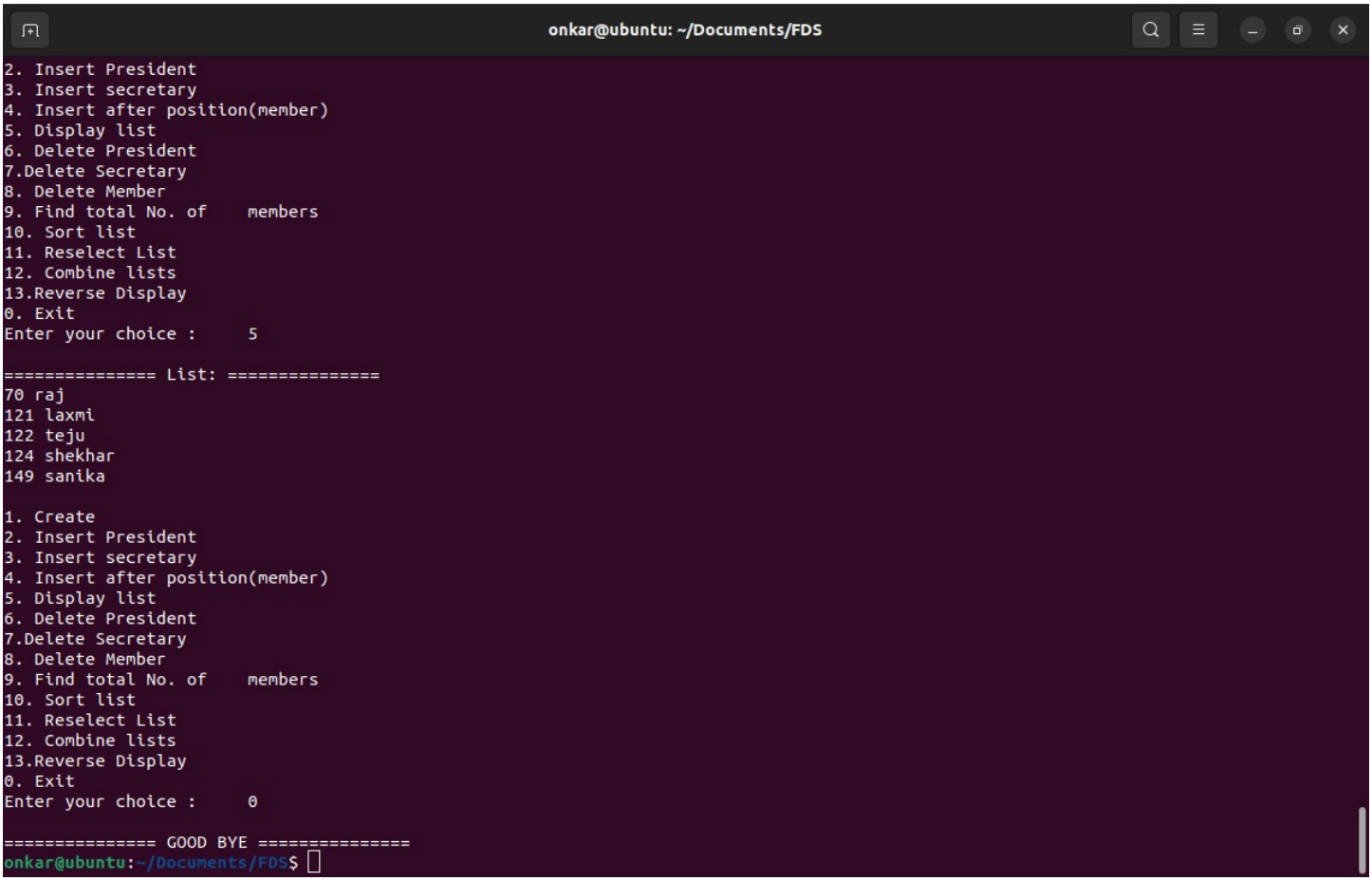
```
onkar@ubuntu: ~/Documents/FDS
13.Reverse Display
0. Exit
Enter your choice :     4

Enter PRN No. after do you want insert : 121

Enter PRN Number : 122
Enter Name : teju

1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
Enter your choice :     11

Select List
1.List 1
2.List 2
Enter choice : 1

1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
```

```
onkar@ubuntu: ~/Documents/FDS
Enter your choice :     12

After concatenation list :

=============== List: ===============
70 raj
123 kunal
124 shekhar
150 tejas
56 sarika
121 laxmi
122 teju
149 sanika

1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
Enter your choice :     5

=============== List: ===============
70 raj
123 kunal
124 shekhar
150 tejas
56 sarika
121 laxmi
122 teju
149 sanika
```

```
onkar@ubuntu: ~/Documents/FDS
13.Reverse Display
0. Exit
Enter your choice :     10

 List is sorted.
=============== List: ===============
56 sarika
70 raj
121 laxmi
122 teju
123 kunal
124 shekhar
149 sanika
150 tejas

1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
Enter your choice :     13

PRN NO:150 Name: tejas
PRN NO:149 Name: sanika
PRN NO:124 Name: shekhar
PRN NO:123 Name: kunal
PRN NO:122 Name: teju
PRN NO:121 Name: laxmi
PRN NO:70 Name: raj
PRN NO:56 Name: sarika
1. Create
```

```
onkar@ubuntu: ~/Documents/FDS
0. Exit
Enter your choice :     6

President deleted..
1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
Enter your choice :     7

Secretary Deleted.
1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
Enter your choice :     8

Enter PRN No. of member to be deleted : 123

Member with PRN No: 123 is deleted.
```

```
onkar@ubuntu: ~/Documents/FDS
13.Reverse Display
0. Exit
Enter your choice :     9

Total members(including President & Secretary) : 5
1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
Enter your choice :     5

=============== List: ===============
70 raj
121 laxmi
122 teju
124 shekhar
149 sanika

1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of    members
10. Sort list
11. Reselect List
12. Combine lists
```


onkar@ubuntu: ~/Documents/FDS
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of members
10. Sort list
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
Enter your choice : 5
========== List: ==========
70 raj
121 laxmi
122 teju
124 shekhar
149 sanika
1. Create
2. Insert President
3. Insert secretary
4. Insert after position(member)
5. Display list
6. Delete President
7.Delete Secretary
8. Delete Member
9. Find total No. of members
10. Sort list
11. Reselect List
12. Combine lists
13.Reverse Display
0. Exit
Enter your choice : 0
========== GOOD BYE ==========
onkar@ubuntu:~/Documents/FDS$