C:\Users\Nikita\Desktop\programs\Doubly-linked-list.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Release

(globals)

Project Classes

Singly-linked-list.cpp Doubly-linked-list.cpp

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  struct node
4  {
5      int info;
6      struct node *rlink;
7      struct node *llink;
8  };
9  typedef struct node *NODE;
10 NODE getnode()
11 {
12     NODE x;
13     x=(NODE)malloc(sizeof(struct node));
14     if (x==NULL)
15     {
16         printf("Memory full\n");
17         exit(0);
18     }
19     return x;
20 }
21 NODE dinsert_front(int item,NODE head)
22 {
23     NODE temp,cur;
24     temp=getnode();
25     temp->info=item;
26     temp->llink=NULL;
27     temp->rlink=NULL;
28     cur=head->rlink;
29     head->rlink=temp;
30     temp->llink=head;
31     temp->rlink=cur;
32     cur->llink=temp;
33     return head;
34 }
35 NODE dinsert_rear(int item,NODE head)
36 {
37     NODE temp,cur;
38     temp=getnode();
39     temp->info=item;
40     temp->llink=NULL;
```

Type here to search

ENG 12:27 16-12-2020

C:\Users\Nikita\Desktop\programs\Doubly-linked-list.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Release

(globals)

Project Classes

Singly-linked-list.cpp Doubly-linked-list.cpp

```
35  NODE dinsert_rear(int item,NODE head)
36  {
37      NODE temp,cur;
38      temp=getnode();
39      temp->info=item;
40      temp->llink=NULL;
41      temp->rlink=NULL;
42      cur=head->llink;
43      head->llink=temp;
44      temp->rlink=head;
45      cur->rlink=temp;
46      temp->llink=cur;
47      return head;
48  }
49  NODE ddelete_front(NODE head)
50  {
51      NODE cur,next;
52      if (head->rlink==head)
53      {
54          printf("List is empty\n");
55          return head;
56      }
57      cur=head->rlink;
58      next=cur->rlink;
59      head->rlink=next;
60      next->llink=head;
61      printf("Item deleted at the front end is:%d\n",cur->info);
62      free(cur);
63      return head;
64  }
65  NODE ddelete_rear(NODE head)
66  {
67      NODE cur,prev;
68      if (head->rlink==head)
69      {
70          printf("List is empty\n");
71          return head;
72      }
73      cur=head->llink;
74      prev=cur->llink;
```

Type here to search

ENG 12:27 16-12-2020

C:\Users\Nikita\Desktop\programs\Doubly-linked-list.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Release

(globals)

Project Classes

Singly-linked-list.cpp Doubly-linked-list.cpp

```
64  }
65  NODE ddelete_rear(NODE head)
66  {
67      NODE cur,prev;
68      if (head->rlink==head)
69      {
70          printf("List is empty\n");
71          return head;
72      }
73      cur=head->llink;
74      prev=cur->llink;
75      prev->rlink=head;
76      head->llink=prev;
77      printf("Item deleted at the rear end is:%d\n",cur->info);
78      free(cur);
79      return head;
80  }
81  void ddisplay(NODE head)
82  {
83      NODE temp;
84      if (head->rlink==head)
85      {
86          printf("List is empty\n");
87      }
88      printf("The contents of the list are:\n");
89      temp=head->rlink;
90      while (temp!=head)
91      {
92          printf("%d\n",temp->info);
93          temp=temp->rlink;
94      }
95  }
96  void dsearch(int key,NODE head)
97  {
98      NODE cur;
99      int count;
100     if (head->rlink==head)
101     {
102         printf("List is empty\n");
103     }
```

Type here to search

ENG 12:27 16-12-2020

C:\Users\Nikita\Desktop\programs\Doubly-linked-list.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Release

(globals)

Project Classes

Singly-linked-list.cpp Doubly-linked-list.cpp

```
96  void dsearch(int key,NODE head)
97  {
98      NODE cur;
99      int count;
100     if (head->rlink==head)
101     {
102         printf("List is empty\n");
103     }
104     cur=head->rlink;
105     count=1;
106     while (cur!=head && cur->info!=key)
107     {
108         cur=cur->rlink;
109         count++;
110     }
111     if (cur==head)
112     {
113         printf("Search unsuccessfull\n");
114     }
115     else
116     {
117         printf("Key element found at the position %d\n",count);
118     }
119 }
120 NODE dinsert_leftpos(int item,NODE head)
121 {
122     NODE cur,prev,temp;
123     if (head->rlink==head)
124     {
125         printf("List is empty\n");
126         return head;
127     }
128     cur=head->rlink;
129     while (cur!=head)
130     {
131         if (cur->info==item)
132         {
133             break;
134         }
135         cur=cur->rlink;
```

Type here to search

ENG 12:27 16-12-2020

C:\Users\Nikita\Desktop\programs\Doubly-linked-list.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Release

(globals)

Project Classes

Singly-linked-list.cpp Doubly-linked-list.cpp

```
120 NODE dinsert_leftpos(int item,NODE head)
121 {
122     NODE cur,prev,temp;
123     if (head->rlink==head)
124     {
125         printf("List is empty\n");
126         return head;
127     }
128     cur=head->rlink;
129     while (cur!=head)
130     {
131         if (cur->info==item)
132         {
133             break;
134         }
135         cur=cur->rlink;
136     }
137     if (cur==head)
138     {
139         printf("No such item found in the list\n");
140         return head;
141     }
142     prev=cur->llink;
143     temp=getnode();
144     temp->llink=NULL;
145     temp->rlink=NULL;
146     printf("Enter the item to be inserted at the left of the given item:\n");
147     scanf("%d",&temp->info);
148     prev->rlink=temp;
149     temp->llink=prev;
150     temp->rlink=cur;
151     cur->llink=temp;
152     return head;
153 }
154 NODE dinsert_rightpos(int item,NODE head)
155 {
156     NODE temp,cur,next;
157     if (head->rlink==head)
158     {
159         printf("List is empty\n");
```

Type here to search

ENG 12:27 16-12-2020

C:\Users\Nikita\Desktop\programs\Doubly-linked-list.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Release

(globals)

Project Classes

Singly-linked-list.cpp Doubly-linked-list.cpp

```
154 NODE dinsert_rightpos(int item,NODE head)
155 {
156     NODE temp,cur,next;
157     if (head->rlink==head)
158     {
159         printf("List is empty\n");
160         return head;
161     }
162     cur=head->rlink;
163     while (cur!=head)
164     {
165         if (cur->info==item)
166         {
167             break;
168         }
169         cur=cur->rlink;
170     }
171     if (cur==head)
172     {
173         printf("No such item found in the list\n");
174         return head;
175     }
176     next=cur->rlink;
177     temp=getnode();
178     temp->llink=NULL;
179     temp->rlink=NULL;
180     printf("Enter the item to be inserted at the right of the given item:\n");
181     scanf("%d",&temp->info);
182     cur->rlink=temp;
183     temp->llink=cur;
184     next->llink=temp;
185     temp->rlink=next;
186     return head;
187 }
188 NODE ddelete_duplicates(int item,NODE head)
189 {
190     NODE prev,cur,next;
191     int count=0;
192     if (head->rlink==head)
193     {
```

Type here to search

ENG 12:27 16-12-2020

C:\Users\Nikita\Desktop\programs\Doubly-linked-list.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Release

(globals)

Project Classes

Singly-linked-list.cpp Doubly-linked-list.cpp

```
188 NODE ddelete_duplicates(int item,NODE head)
189 {
190     NODE prev,cur,next;
191     int count=0;
192     if (head->rlink==head)
193     {
194         printf("List is empty\n");
195         return head;
196     }
197     cur=head->rlink;
198     while (cur!=head)
199     {
200         if (cur->info!=item)
201         {
202             cur=cur->rlink;
203         }
204         else
205         {
206             count++;
207             if (count==1)
208             {
209                 cur=cur->rlink;
210                 continue;
211             }
212             else
213             {
214                 prev=cur->llink;
215                 next=cur->rlink;
216                 prev->rlink=next;
217                 next->llink=prev;
218                 free(cur);
219                 cur=next;
220             }
221         }
222     }
223     if (count==0)
224     {
225         printf("No such item found in the list\n");
226     }
227     else
```

Type here to search

ENG 12:27 16-12-2020

C:\Users\Nikita\Desktop\programs\Doubly-linked-list.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Release

(globals)

Project Classes

Singly-linked-list.cpp Doubly-linked-list.cpp

```
221         }
222     }
223     if (count==0)
224     {
225         printf("No such item found in the list\n");
226     }
227     else
228     {
229         printf("Removed all the duplicate elements of the given item successfully\n");
230     }
231     return head;
232 }
233 int main()
234 {
235 NODE head;
236 int item, choice,key;
237 head=getnode();
238 head->llink=head;
239 head->rlink=head;
240 for(;;)
241 {
242     printf("\n1:dinsert front\n2:dinsert rear\n3:ddelete front\n4:ddelete rear\n5:ddisplay\n6:dsearch\n7:dinsert lestpos\n8:dinsert rightpos\n9:ddelete duplicates\n10:exit\n");
243     printf("enter the choice\n");
244     scanf("%d",&choice);
245     switch(choice)
246     {
247         case 1: printf("Enter the item at front end:\n");
248                 scanf("%d",&item);
249                 head=dinsert_front(item,head);
250                 break;
251         case 2: printf("Enter the item at rear end:\n");
252                 scanf("%d",&item);
253                 head=dinsert_rear(item,head);
254                 break;
255         case 3:head=ddelete_front(head);
256                break;
257         case 4:head=ddelete_rear(head);
258                break;
259         case 5:ddisplay(head);
260                break;
```

Type here to search

ENG 12:28 16-12-2020

C:\Users\Nikita\Desktop\programs\Doubly-linked-list.cpp - Dev-C++ 5.11

File Edit Search View Project Execute Tools AStyle Window Help

TDM-GCC 4.9.2 64-bit Release

(globals)

Project Classes

Singly-linked-list.cpp Doubly-linked-list.cpp

```
242     printf("\n1:dinsert front\n2:dinsert rear\n3:ddelete front\n4:ddelete rear\n5:ddisplay\n6:dsearch\n7:dinsert lestpos\n8:dinsert rightpos\n9:ddelete duplicates\n10:exit\n");
243     printf("enter the choice\n");
244     scanf("%d",&choice);
245     switch(choice)
246     {
247         case 1: printf("Enter the item at front end:\n");
248                 scanf("%d",&item);
249                 head=dinsert_front(item,head);
250                 break;
251         case 2: printf("Enter the item at rear end:\n");
252                 scanf("%d",&item);
253                 head=dinsert_rear(item,head);
254                 break;
255         case 3:head=ddelete_front(head);
256                break;
257         case 4:head=ddelete_rear(head);
258                break;
259         case 5:ddisplay(head);
260                break;
261         case 6:printf("Enter the key element to be searched:\n");
262                scanf("%d",&key);
263                dsearch(key,head);
264                break;
265         case 7:printf("Enter the key element:\n");
266                scanf("%d",&key);
267                head=dinsert_leftpos(key,head);
268                break;
269         case 8:printf("Enter the key element:\n");
270                scanf("%d",&key);
271                head=dinsert_rightpos(key,head);
272                break;
273         case 9:printf("Enter the key element whose duplicates should be removed:\n");
274                scanf("%d",&key);
275                head=ddelete_duplicates(key,head);
276                break;
277         default:exit(0);
278         }
279     }
280     return 0;
281 }
```

Type here to search

ENG 12:28 16-12-2020