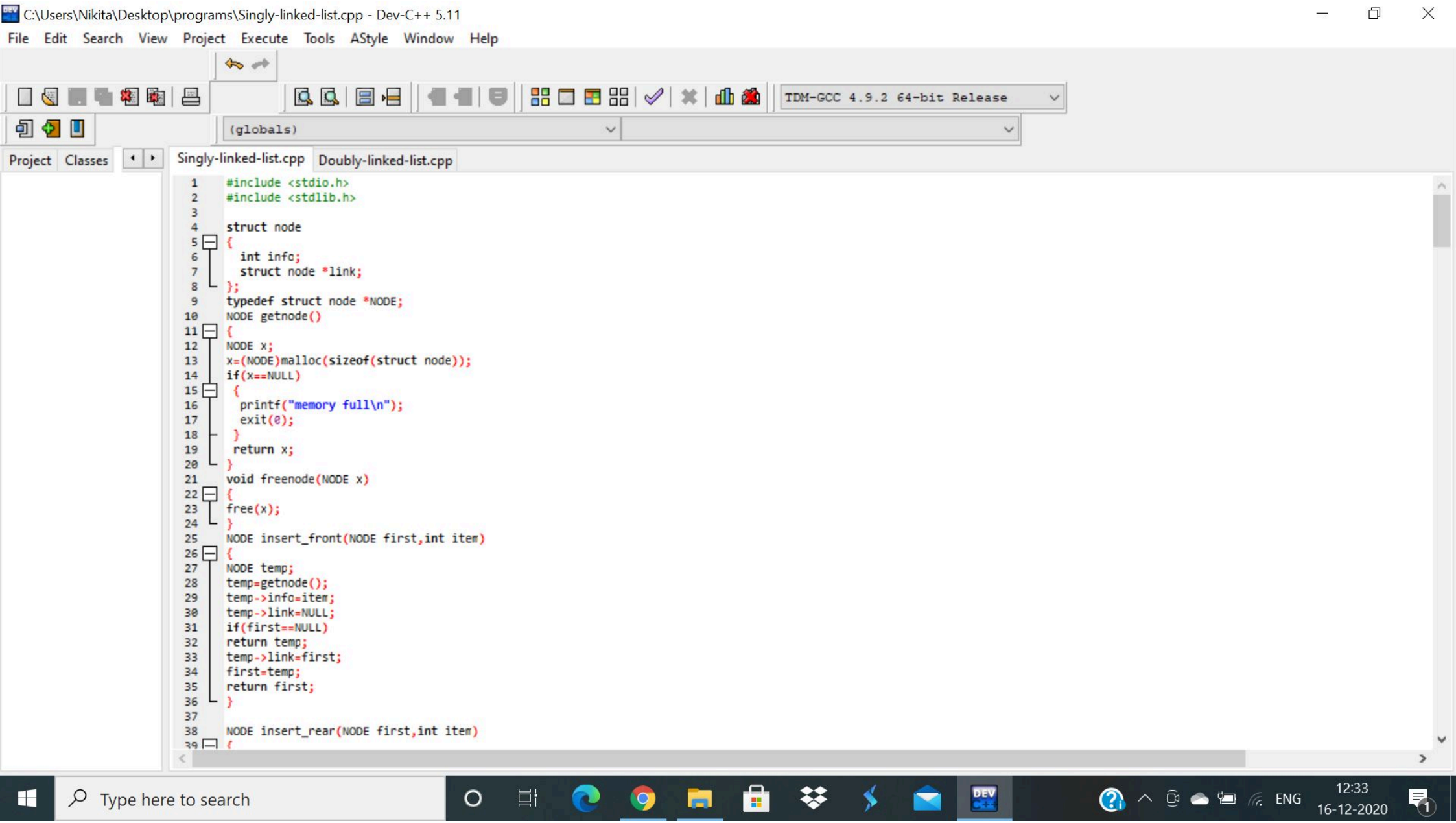
C:\Users\Nikita\Desktop\programs\Singly-linked-list.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes
Singly-linked-list.cpp Doubly-linked-list.cpp
#include <stdio.h>
#include <stdlib.h>
struct node
{
int info;
struct node *link;
};
typedef struct node *NODE;
NODE getnode()
{
NODE x;
x=(NODE)malloc(sizeof(struct node));
if(x==NULL)
{
printf("memory full\n");
exit(0);
}
return x;
}
void freenode(NODE x)
{
free(x);
}
NODE insert_front(NODE first,int item)
{
NODE temp;
temp=getnode();
temp->info=item;
temp->link=NULL;
if(first==NULL)
return temp;
temp->link=first;
first=temp;
return first;
}
NODE insert_rear(NODE first,int item)
Type here to search
ENG
12:33
16-12-2020

C:\Users\Nikita\Desktop\programs\Singly-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

```
37
38  NODE insert_rear(NODE first,int item)
39  {
40  NODE temp,cur;
41  temp=getnode();
42  temp->info=item;
43  temp->link=NULL;
44  if(first==NULL)
45   return temp;
46  cur=first;
47  while(cur->link!=NULL)
48   cur=cur->link;
49  cur->link=temp;
50  return first;
51  }
52
53  NODE insert_pos(int item,int pos,NODE first)
54  {
55      NODE temp,cur,prev;
56      int count;
57      temp=getnode();
58      temp->info=item;
59      temp->link=NULL;
60      if (first==NULL && pos==1)
61      {
62          return temp;
63      }
64      if (first==NULL)
65      {
66          printf("Invalid position\n");
67          return NULL;
68      }
69      if (pos==1)
70      {
71          temp->link=first;
72          return temp;
73      }
74      count=1;
75      prev=NULL;
```

Type here to search

ENG 12:33 16-12-2020

C:\Users\Nikita\Desktop\programs\Singly-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

```
72          return temp;
73      }
74      count=1;
75      prev=NULL;
76      cur=first;
77      while (cur!=NULL && count!=pos)
78      {
79          prev=cur;
80          cur=cur->link;
81          count++;
82      }
83      if (count==pos)
84      {
85          prev->link=temp;
86          temp->link=cur;
87          return first;
88      }
89      printf("Invalid position\n");
90      return first;
91  }
92
93
94  NODE delete_front(NODE first)
95  {
96  NODE temp;
97  if(first==NULL)
98  {
99  printf("list is empty cannot delete\n");
100 return first;
101 }
102 temp=first;
103 temp=temp->link;
104 printf("item deleted at front-end is=%d\n",first->info);
105 free(first);
106 return temp;
107 }
108
109 NODE delete_rear(NODE first)
110 {
```

Type here to search

ENG 12:34 16-12-2020

C:\Users\Nikita\Desktop\programs\Singly-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

```
107 }
108
109 NODE delete_rear(NODE first)
110 {
111 NODE cur,prev;
112 if(first==NULL)
113 {
114 printf("list is empty cannot delete\n");
115 return first;
116 }
117 if(first->link==NULL)
118 {
119 printf("item deleted is %d\n",first->info);
120 free(first);
121 return NULL;
122 }
123 prev=NULL;
124 cur=first;
125 while(cur->link!=NULL)
126 {
127 prev=cur;
128 cur=cur->link;
129 }
130 printf("item deleted at rear-end is %d",cur->info);
131 free(cur);
132 prev->link=NULL;
133 return first;
134 }
135 NODE delete_pos(int pos,NODE first)
136 {
137     NODE prev,cur;
138     int count;
139     if (first==NULL || pos<=0)
140     {
141         printf("Invalid position\n");
142         return NULL;
143     }
144     if (pos==1)
145     {
```

Type here to search

ENG 12:34 16-12-2020

C:\Users\Nikita\Desktop\programs\Singly-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

```
143     }
144     if (pos==1)
145     {
146         cur=first;
147         first=first->link;
148         freenode(cur);
149         return first;
150     }
151     prev=NULL;
152     cur=first;
153     count=1;
154     while (cur!=NULL)
155     {
156         if (count==pos)
157         {
158             break;
159         }
160         prev=cur;
161         cur=cur->link;count++;
162     }
163     if (count!=pos)
164     {
165         printf("Invalid position\n");
166         return first;
167     }
168     prev->link=cur->link;
169     freenode(cur);
170     return first;
171 }
172
173
174 NODE concat(NODE first,NODE second)
175 {
176 NODE cur;
177 if(first==NULL)
178 return second;
179 if(second==NULL)
180 return first;
181 cur=first;
```

Type here to search

ENG 12:34 16-12-2020

C:\Users\Nikita\Desktop\programs\Singly-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

```
174 NODE concat(NODE first,NODE second)
175 {
176 NODE cur;
177 if(first==NULL)
178 return second;
179 if(second==NULL)
180 return first;
181 cur=first;
182 while(cur->link!=NULL)
183 cur=cur->link;
184 cur->link=second;
185 return first;
186 }
187 NODE reverse(NODE first)
188 {
189 NODE cur,temp;
190 cur=NULL;
191 while(first!=NULL)
192 {
193 temp=first;
194 first=first->link;
195 temp->link=cur;
196 cur=temp;
197 }
198 return cur;
199 }
200 NODE sort_asc(NODE first)
201 {
202   int tmp;
203   NODE cur,next;
204   cur=first;
205   next=NULL;
206   if(first==NULL){
207   printf("List is empty\n");
208
209   }
210   while(cur!=NULL)
211   {
212    next=cur->link;
```

Type here to search

ENG 12:34 16-12-2020

C:\Users\Nikita\Desktop\programs\Singly-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

```
200 NODE sort_asc(NODE first)
201 {
202   int tmp;
203   NODE cur,next;
204   cur=first;
205   next=NULL;
206   if(first==NULL){
207   printf("List is empty\n");
208
209   }
210   while(cur!=NULL)
211   {
212    next=cur->link;
213     while(next!=NULL)
214    {
215    if(cur->info>next->info)
216    {
217     tmp=cur->info;
218     cur->info=next->info;
219     next->info=tmp;
220    }
221    next=next->link;
222   }
223   cur=cur->link;
224   }
225   return first;
226 }
227
228 void display(NODE first)
229 {
230  NODE temp;
231  if(first==NULL)
232  printf("list empty cannot display items\n");
233  for(temp=first;temp!=NULL;temp=temp->link)
234   {
235   printf("%d\n",temp->info);
236   }
237 }
238 NODE push(NODE first,int item)
```

Type here to search

ENG 12:34 16-12-2020

C:\Users\Nikita\Desktop\programs\Singly-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

```
228 void display(NODE first)
229 {
230  NODE temp;
231  if(first==NULL)
232  printf("list empty cannot display items\n");
233  for(temp=first;temp!=NULL;temp=temp->link)
234   {
235   printf("%d\n",temp->info);
236   }
237 }
238 NODE push(NODE first,int item)
239 {
240 NODE temp;
241 temp=getnode();
242 temp->info=item;
243 temp->link=NULL;
244 if(first==NULL)
245 return temp;
246 temp->link=first;
247 first=temp;
248 return first;
249 }
250 NODE pop(NODE first)
251 {
252 NODE temp;
253 if(first==NULL)
254 {
255 printf("list is empty cannot delete\n");
256 return first;
257 }
258 temp=first;
259 temp=temp->link;
260 printf("item deleted at front-end is=%d\n",first->info);
261 free(first);
262 return temp;
263 }
264 NODE enqueue(NODE first,int item)
265 {
266 NODE temp;
```

Type here to search

ENG 12:34 16-12-2020

C:\Users\Nikita\Desktop\programs\Singly-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

```
262     return temp;
263     }
264     NODE enqueue(NODE first,int item)
265     {
266     NODE temp;
267     temp=getnode();
268     temp->info=item;
269     temp->link=NULL;
270     if(first==NULL)
271     return temp;
272     temp->link=first;
273     first=temp;
274     return first;
275     }
276     NODE dequeue(NODE first)
277     {
278     NODE temp;
279     if(first==NULL)
280     {
281     printf("list is empty cannot delete\n");
282     return first;
283     }
284     temp=first;
285     temp=temp->link;
286     printf("item deleted at front-end is=%d\n",first->info);
287     free(first);
288     return temp;
289     }
290     int main()
291     {
292      int item,choice,pos,n1,i,n2,n,flag=0;
293      NODE first=NULL,a,b;
294     for(;;)
295     {
296         printf("\n 1:Insert_front\n 2:Delete_front\n 3:Insert_rear\n 4:Delete_rear\n5:Insert Position\n6:Delete Position\n7:Concat\n8:Reverse\n9:Sort\n10:Display\n11:Stack operation
297         printf("enter the choice\n");
298         scanf("%d",&choice);
299         switch(choice)
300         {
```

Type here to search

ENG 12:35 16-12-2020

C:\Users\Nikita\Desktop\programs\Singly-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

```
289 }
290 int main()
291 {
292  int item,choice,pos,n1,i,n2,n,flag=0;
293  NODE first=NULL,a,b;
294 for(;;)
295 {
296     printf("\n 1:Insert_front\n 2:Delete_front\n 3:Insert_rear\n 4:Delete_rear\n5:Insert Position\n6:Delete Position\n7:Concat\n8:Reverse\n9:Sort\n10:Display\n11:Stack operatior
297     printf("enter the choice\n");
298     scanf("%d",&choice);
299     switch(choice)
300     {
301         case 1:printf("enter the item at front-end\n");
302                scanf("%d",&item);
303                first=insert_front(first,item);
304                break;
305         case 2:first=delete_front(first);
306                break;
307         case 3:printf("enter the item at rear-end\n");
308                scanf("%d",&item);
309                first=insert_rear(first,item);
310                break;
311         case 4:first=delete_rear(first);
312                break;
313         case 5:printf("Enter the item and the position:\n");
314                 scanf("%d%d",&item,&pos);
315                 first=insert_pos(item,pos,first);
316                 break;
317         case 6:printf("Enter the position:\n");
318                 scanf("%d",&pos);
319                 first=delete_pos(pos,first);
320                 break;
321         case 7:printf("enter the no of nodes in 1\n");
322                scanf("%d",&n);
323                a=NULL;
324                 for( i=0;i<n;i++)
325                     {
326                         printf("enter the item\n");
327                         scanf("%d",&item);
```

Type here to search

ENG 12:35 16-12-2020

C:\Users\Nikita\Desktop\programs\Singly-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

```
321         case 7:printf("enter the no of nodes in 1\n");
322                scanf("%d",&n);
323                a=NULL;
324                for( i=0;i<n;i++)
325                    {
326                        printf("enter the item\n");
327                        scanf("%d",&item);
328                        a=insert_rear(a,item);
329                    }
330                printf("enter the no of nodes in 2\n");
331                scanf("%d",&n);
332                b=NULL;
333                    for(i=0;i<n;i++)
334                        {
335                           printf("enter the item\n");
336                            scanf("%d",&item);
337                            b=insert_rear(b,item);
338                        }
339                         a=concat(a,b);
340                         display(a);
341                         break;
342         case 8:first=reverse(first);
343                display(first);
344                break;
345         case 9:first=sort_asc(first);
346                break;
347         case 10:display(first);
348                break;
349         case 11:
350              do
351              {
352
353             printf("\n1:Push\n2:Pop\n3:Display\n");
354             printf("Enter your choice\n");
355             scanf("%d",&n);
356             switch(n)
357             {
358                 case 1: printf("enter the item to be inserted in the stack\n");
359                         scanf("%d",&item);
```

Type here to search

ENG 12:35 16-12-2020

C:\Users\Nikita\Desktop\programs\Singly-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

```
351             {
352
353             printf("\n1:Push\n2:Pop\n3:Display\n");
354             printf("Enter your choice\n");
355             scanf("%d",&n);
356             switch(n)
357             {
358                  case 1: printf("enter the item to be inserted in the stack\n");
359                          scanf("%d",&item);
360                          first=insert_front(first,item);
361                          break;
362                 case 2: first=delete_front(first);
363                          break;
364                 case 3: display(first);
365         }
366             }while(choice==11);
367             break;
368         case 12:
369             do
370             {
371             printf("\n1.Insert\n2:Delete\n3:Display\n");
372             printf("Enter your choice\n");
373             scanf("%d",&n1);
374             switch(n1)
375             {
376                 case 1: printf("enter the item to be inserted in the queue\n");
377                         scanf("%d",&item);
378                         first=insert_rear(first,item);
379                         break;
380                 case 2: first=delete_front(first);
381                        break;
382                 case 3: display(first);
383             }
384         }while(choice==12);
385         break;
386
387 }}
388 }
389
```

Type here to search

ENG 12:35 16-12-2020