Phat Panhareach

Professor Name: Buo Channa

Subject Name: Advance Algorithm

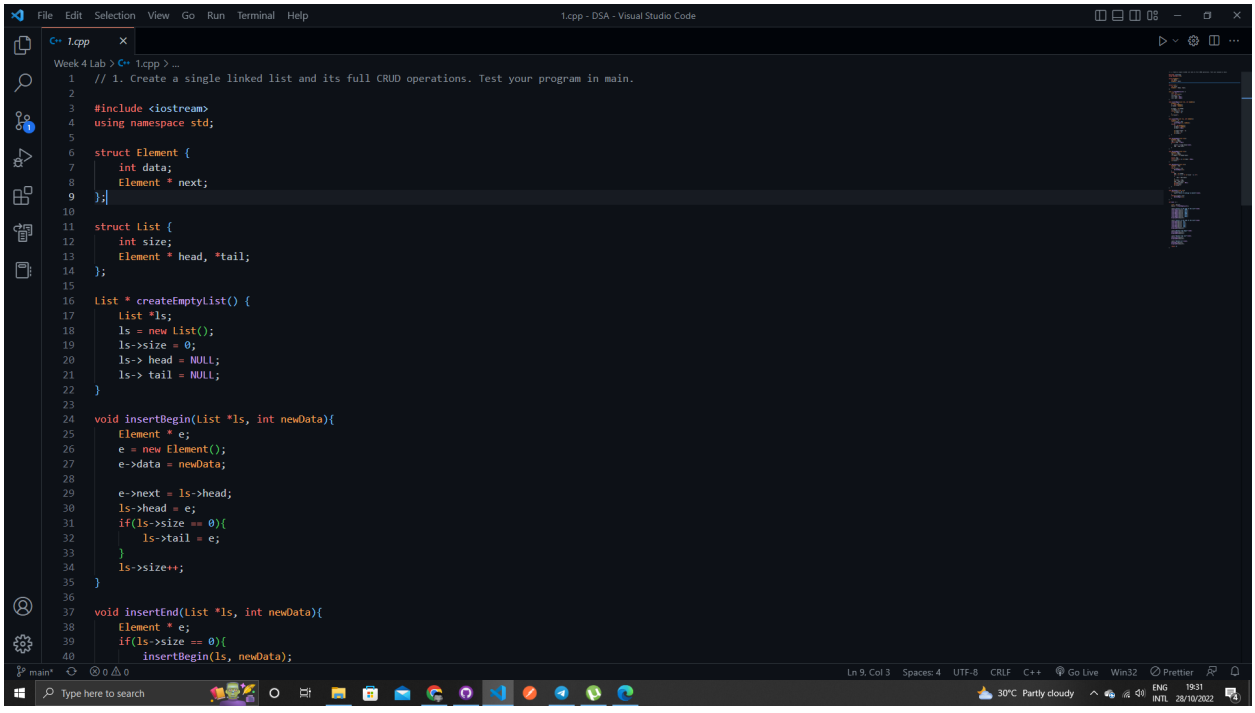
Major: Computer Science (Gen 8)

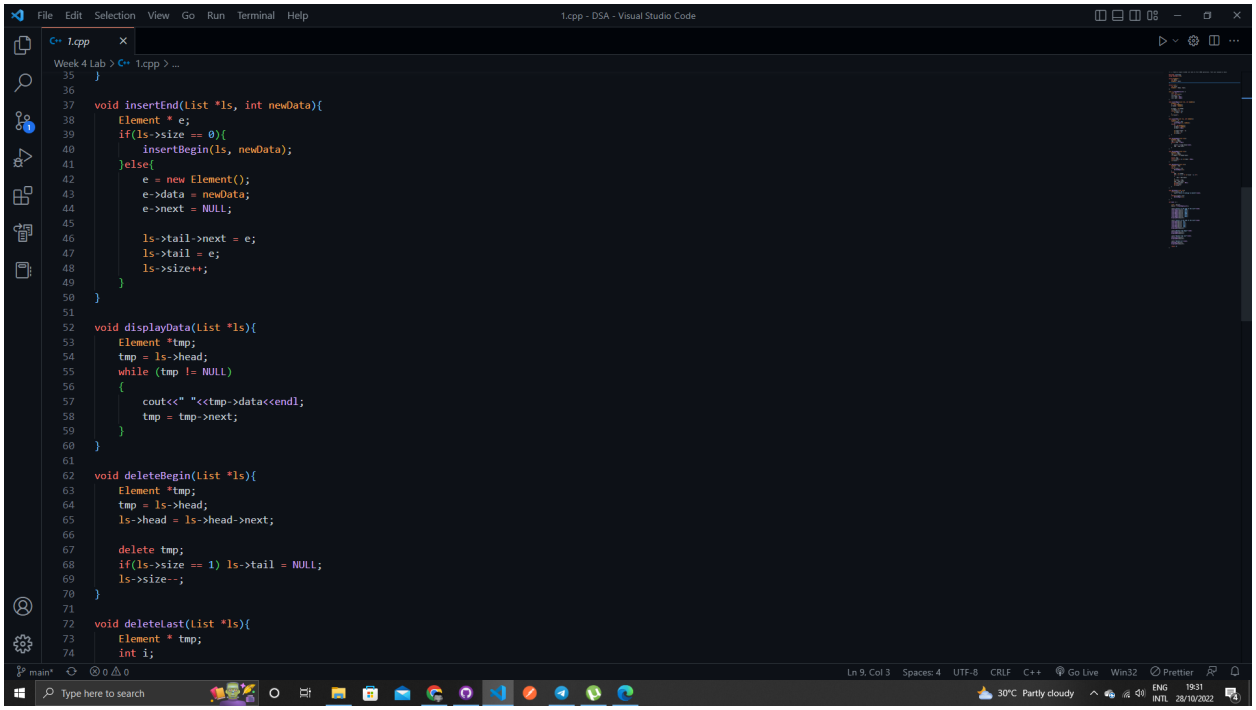
Group: B

Report Advanced Algorithm Lab 4

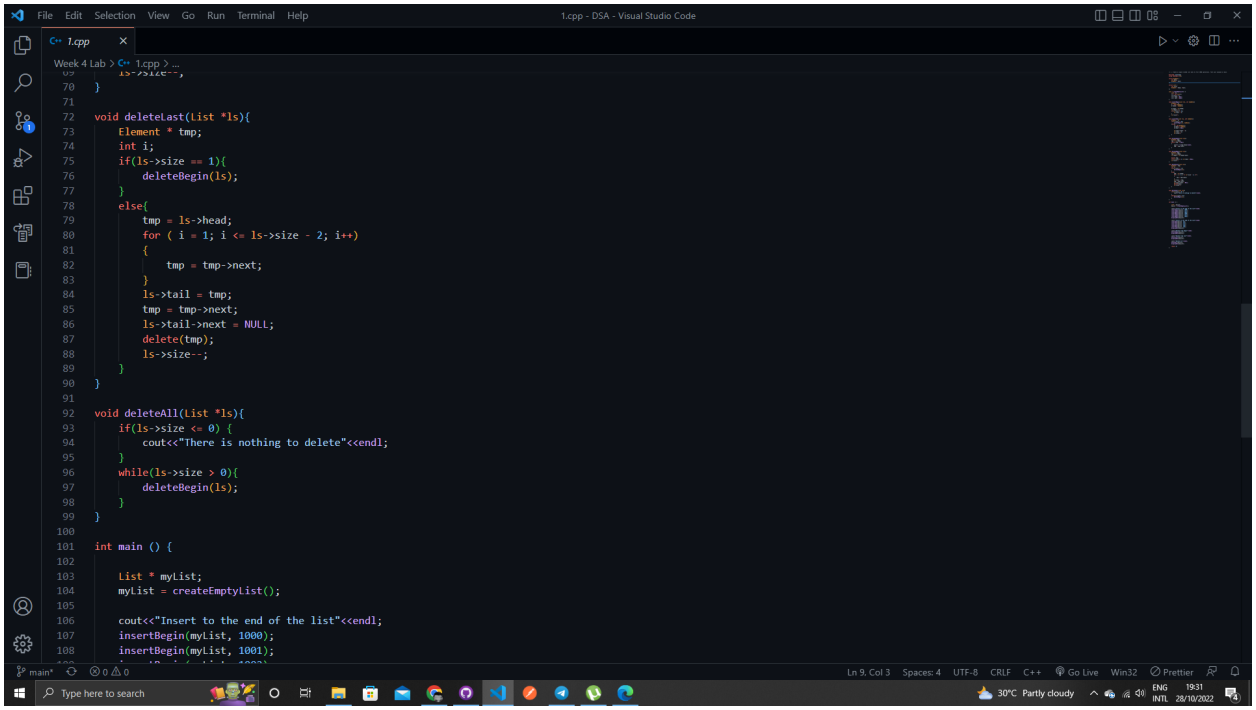
Exercise 1:

1. Create a single linked list and its full CRUD operations. Test your program in main.

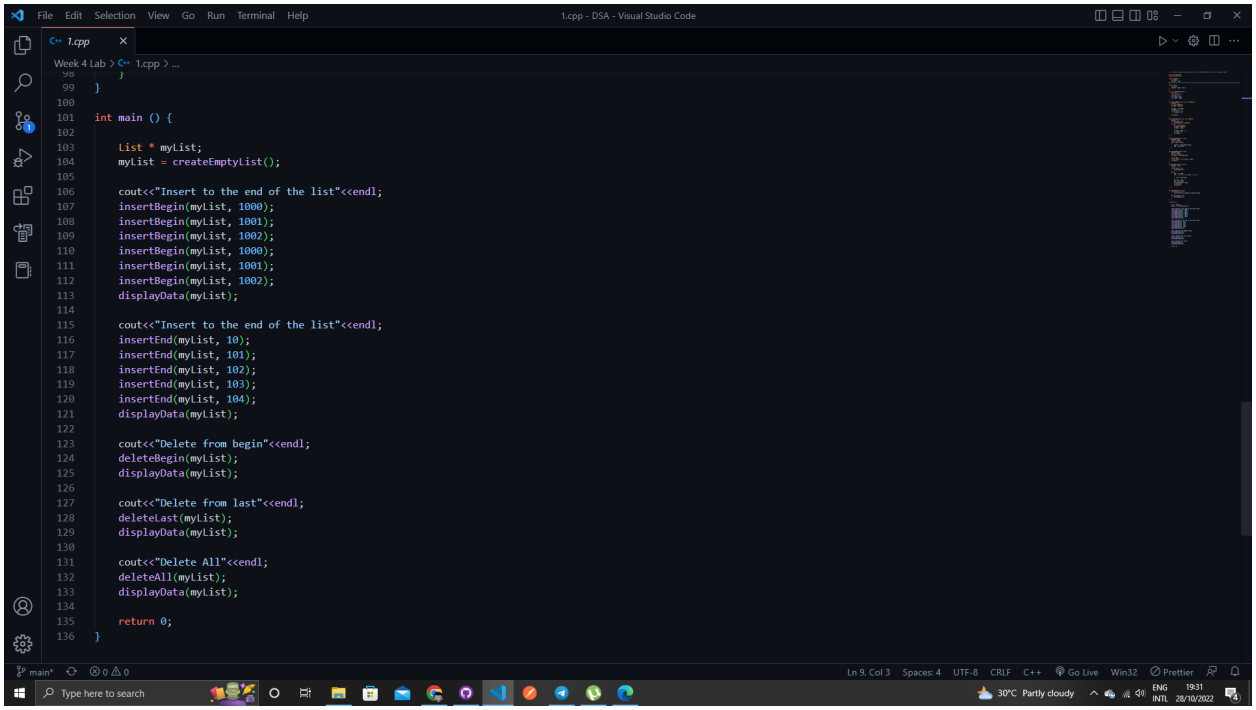




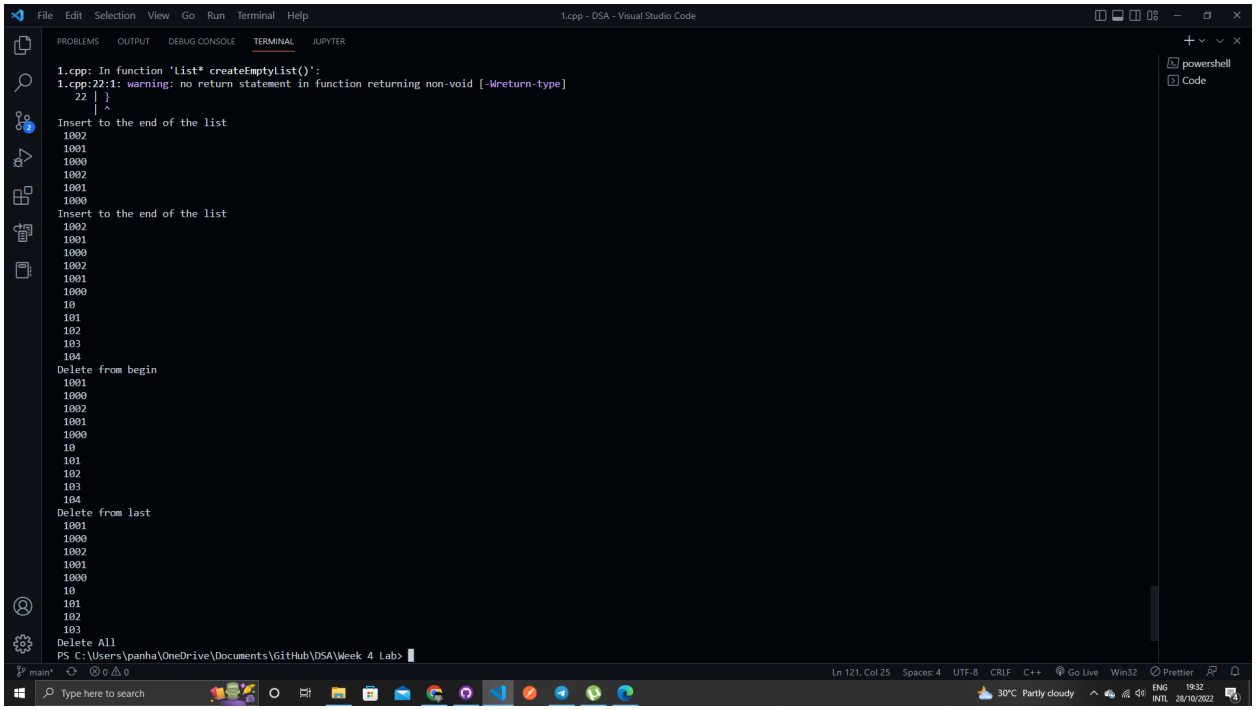
```cpp
}

void insertEnd(List *ls, int newData){
    Element * e;
    if(ls->size == 0){
        insertBegin(ls, newData);
    }else{
        e = new Element();
        e->data = newData;
        e->next = NULL;

        ls->tail->next = e;
        ls->tail = e;
        ls->size++;
    }
}

void displayData(List *ls){
    Element *tmp;
    tmp = ls->head;
    while (tmp != NULL)
    {
        cout<<" "<<tmp->data<<endl;
        tmp = tmp->next;
    }
}

void deleteBegin(List *ls){
    Element *tmp;
    tmp = ls->head;
    ls->head = ls->head->next;

    delete tmp;
    if(ls->size == 1) ls->tail = NULL;
    ls->size--;
}

void deleteLast(List *ls){
    Element * tmp;
    int i;
```



```cpp
}

void deleteLast(List *ls){
    Element * tmp;
    int i;
    if(ls->size == 1){
        deleteBegin(ls);
    }
    else{
        tmp = ls->head;
        for ( i = 1; i <= ls->size - 2; i++)
        {
            tmp = tmp->next;
        }
        ls->tail = tmp;
        tmp = tmp->next;
        ls->tail->next = NULL;
        delete(tmp);
        ls->size--;
    }
}

void deleteAll(List *ls){
    if(ls->size <= 0) {
        cout<<"There is nothing to delete"<<endl;
    }
    while(ls->size > 0){
        deleteBegin(ls);
    }
}

int main () {

    List * myList;
    myList = createEmptyList();

    cout<<"Insert to the end of the list"<<endl;
    insertBegin(myList, 1000);
    insertBegin(myList, 1001);
```



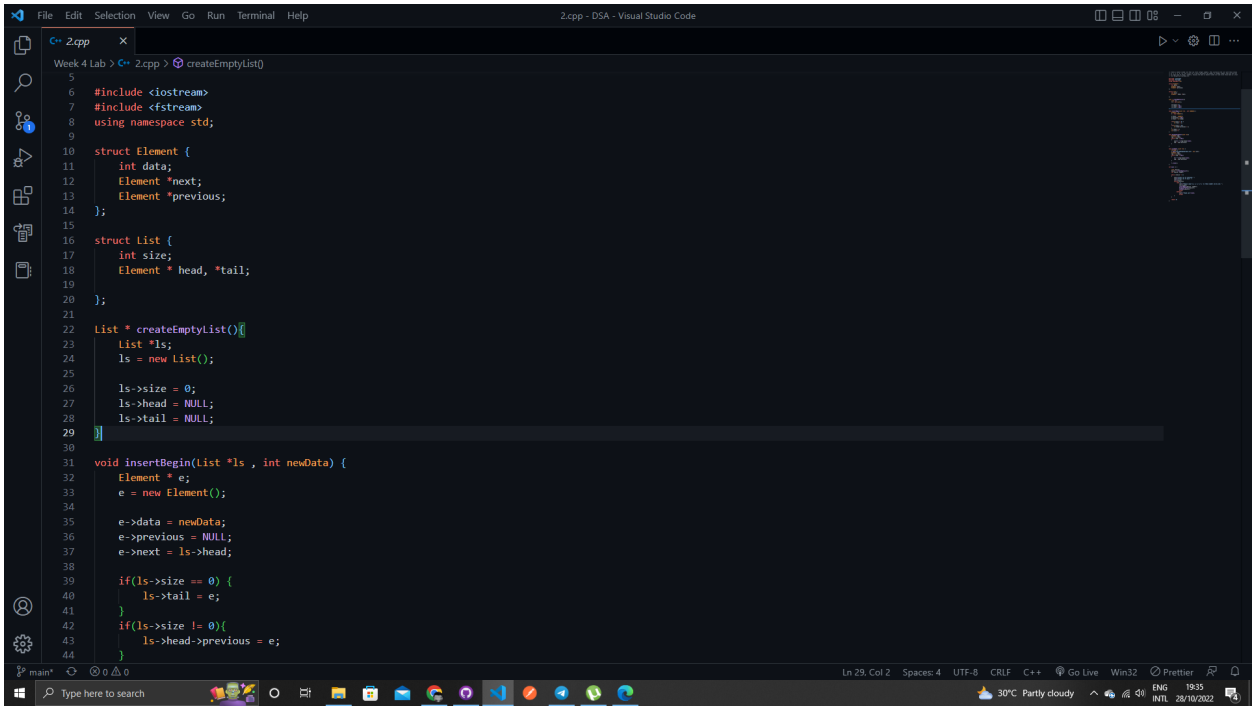
```
}
}

int main () {

    List * myList;
    myList = createEmptyList();

    cout<<"Insert to the end of the list"<<endl;
    insertBegin(myList, 1000);
    insertBegin(myList, 1001);
    insertBegin(myList, 1002);
    insertBegin(myList, 1000);
    insertBegin(myList, 1001);
    insertBegin(myList, 1002);
    displayData(myList);

    cout<<"Insert to the end of the list"<<endl;
    insertEnd(myList, 10);
    insertEnd(myList, 101);
    insertEnd(myList, 102);
    insertEnd(myList, 103);
    insertEnd(myList, 104);
    displayData(myList);

    cout<<"Delete from begin"<<endl;
    deleteBegin(myList);
    displayData(myList);

    cout<<"Delete from last"<<endl;
    deleteLast(myList);
    displayData(myList);

    cout<<"Delete All"<<endl;
    deleteAll(myList);
    displayData(myList);

    return 0;
}
```

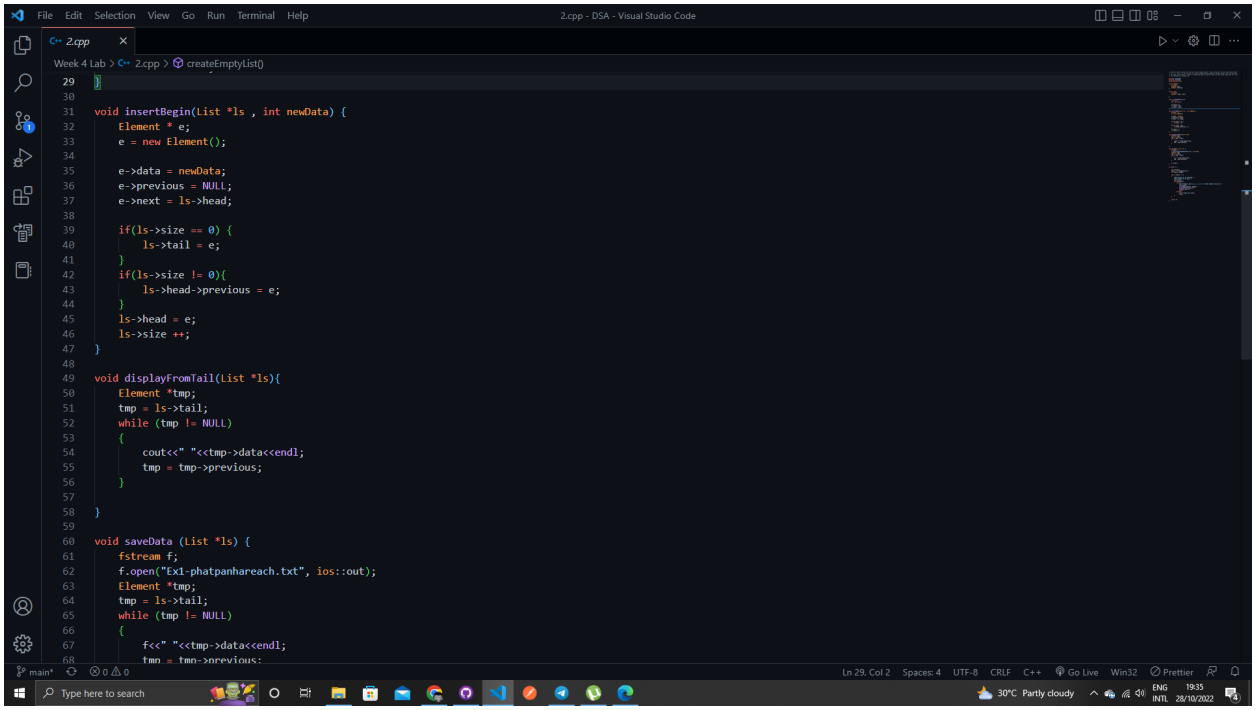


```
1.cpp: In function 'List* createEmptyList()':
1.cpp:22:1: warning: no return statement in function returning non-void [-Wreturn-type]
   22 | }
      | ^
Insert to the end of the list
 1002
 1001
 1000
 1002
 1001
 1000
Insert to the end of the list
 1002
 1001
 1000
 1002
 1001
 1000
 10
 101
 102
 103
 104
Delete from begin
 1001
 1000
 1002
 1001
 1000
 10
 101
 102
 103
 104
Delete from last
 1001
 1000
 1002
 1001
 1000
 10
 101
 102
 103
Delete All
PS C:\Users\panha\OneDrive\Documents\GitHub\DSA\Week 4 Lab>
```

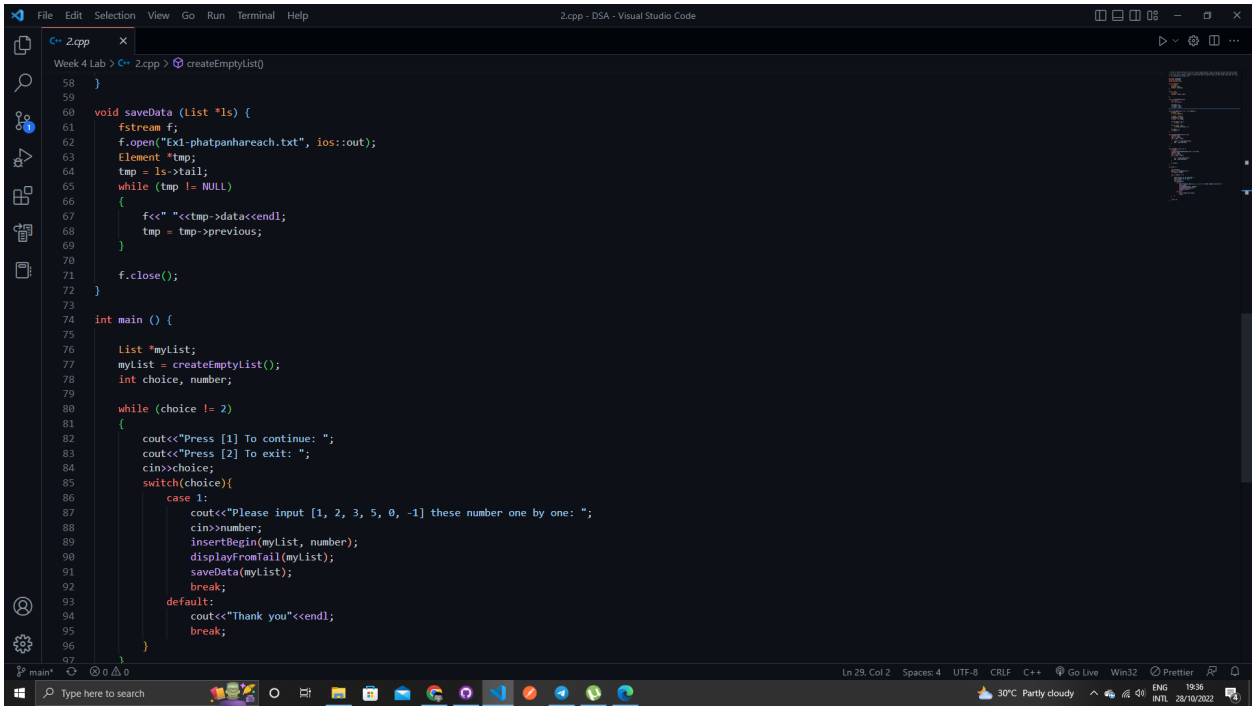
Exercise 2:

Create a double linked list that can store integer numbers. Then initialize the list with these values [1, 2, 3, 5, 0, -1]. Ask a user for n which represents the number of integer numbers then ask the user to input each of those n numbers. Display the data in double linked list and write those data in a text file Output-Ex1-YourName.txt.

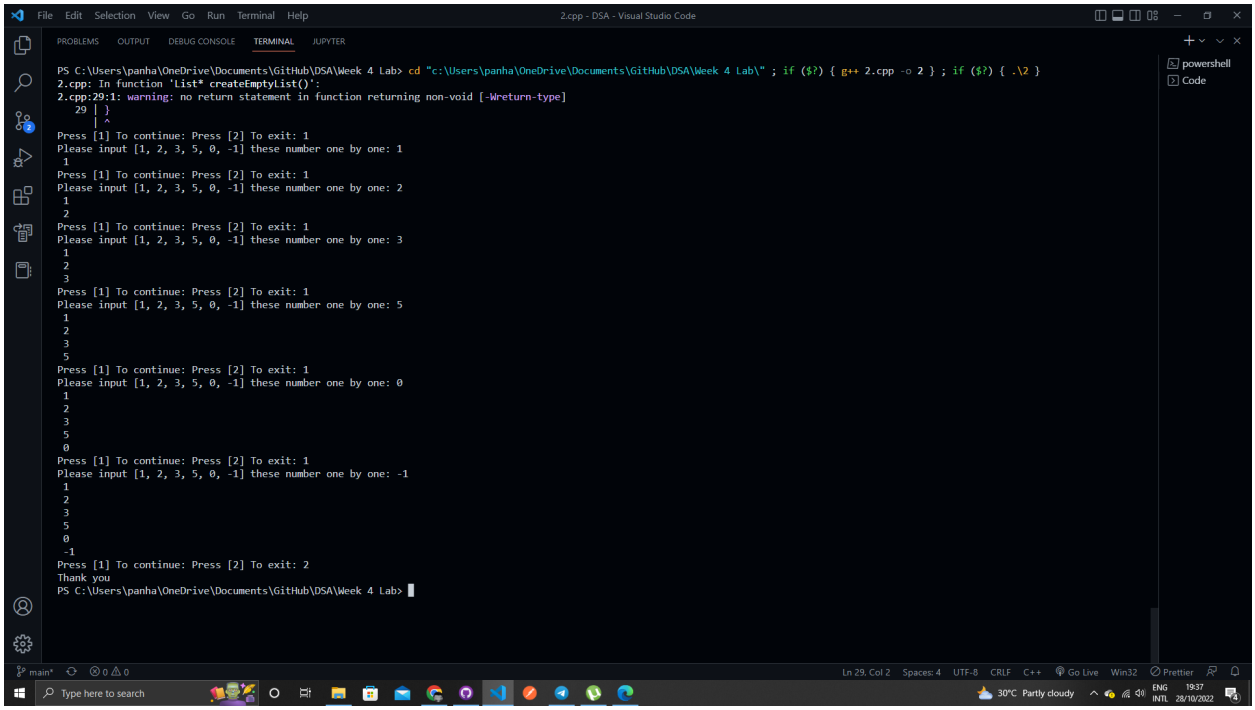


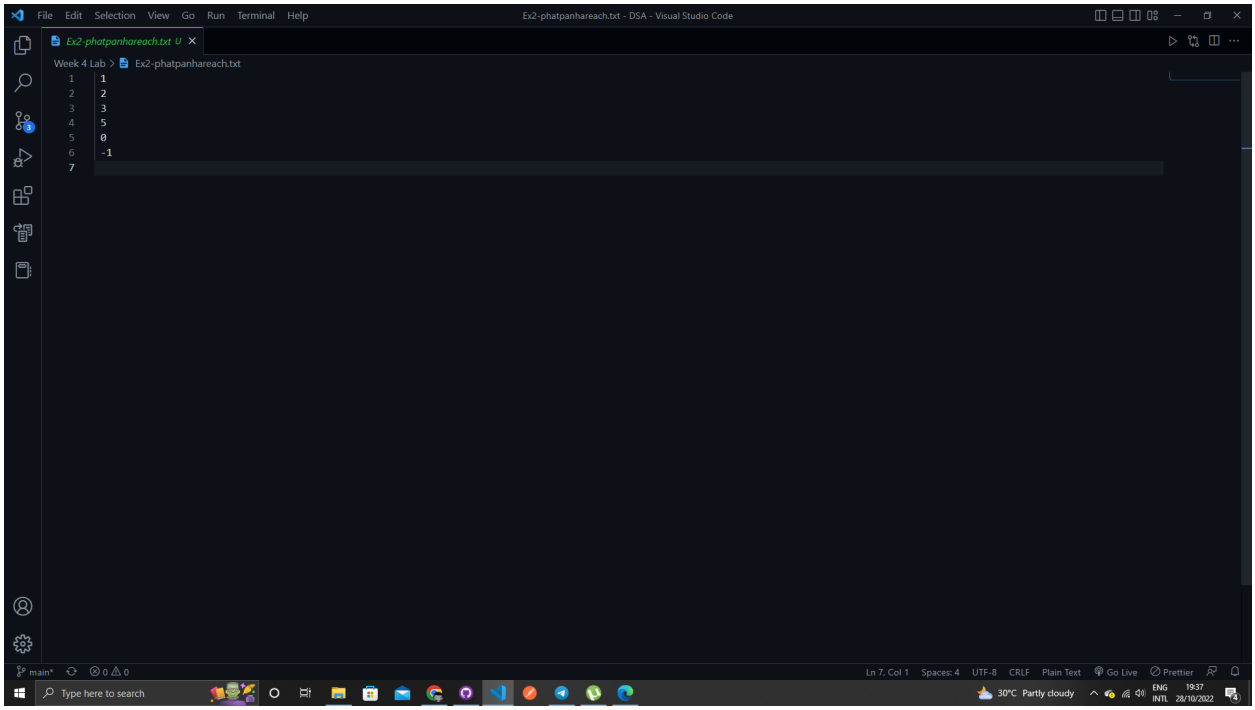


```cpp
}

void insertBegin(List *ls , int newData) {
    Element * e;
    e = new Element();

    e->data = newData;
    e->previous = NULL;
    e->next = ls->head;

    if(ls->size == 0) {
        ls->tail = e;
    }
    if(ls->size != 0){
        ls->head->previous = e;
    }
    ls->head = e;
    ls->size ++;
}

void displayFromTail(List *ls){
    Element *tmp;
    tmp = ls->tail;
    while (tmp != NULL)
    {
        cout<<" "<<tmp->data<<endl;
        tmp = tmp->previous;
    }

}

void saveData (List *ls) {
    fstream f;
    f.open("Ex1-phatpanhareach.txt", ios::out);
    Element *tmp;
    tmp = ls->tail;
    while (tmp != NULL)
    {
        f<<" "<<tmp->data<<endl;
        tmp = tmp->previous;
```



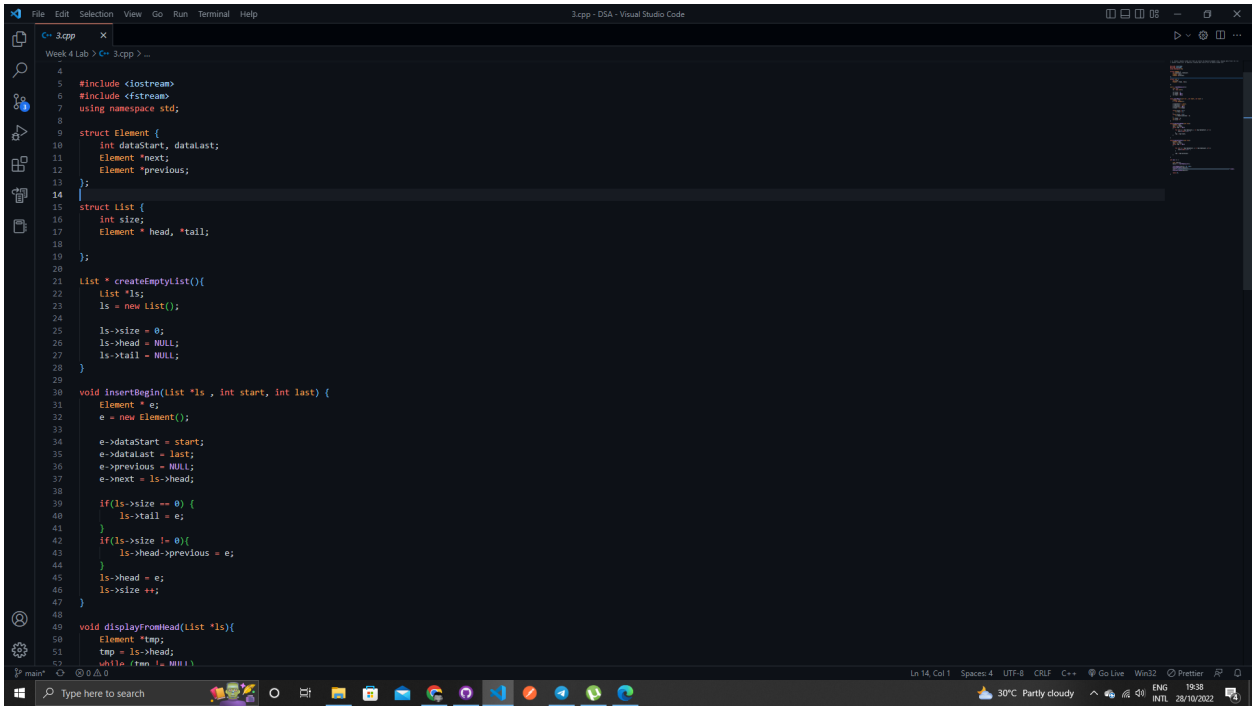
```cpp
}

void saveData (List *ls) {
    fstream f;
    f.open("Ex1-phatpanhareach.txt", ios::out);
    Element *tmp;
    tmp = ls->tail;
    while (tmp != NULL)
    {
        f<<" "<<tmp->data<<endl;
        tmp = tmp->previous;
    }

    f.close();
}

int main () {

    List *myList;
    myList = createEmptyList();
    int choice, number;

    while (choice != 2)
    {
        cout<<"Press [1] To continue: ";
        cout<<"Press [2] To exit: ";
        cin>>choice;
        switch(choice){
            case 1:
                cout<<"Please input [1, 2, 3, 5, 0, -1] these number one by one: ";
                cin>>number;
                insertBegin(myList, number);
                displayFromTail(myList);
                saveData(myList);
                break;
            default:
                cout<<"Thank you"<<endl;
                break;
        }
```


File Edit Selection View Go Run Terminal Help
2.cpp - DSA - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
PS C:\Users\panha\OneDrive\Documents\GitHub\DSA\Week 4 Lab> cd "c:\Users\panha\OneDrive\Documents\GitHub\DSA\Week 4 Lab\" ; if ($?) { g++ 2.cpp -o 2 } ; if ($?) { .\2 }
2.cpp: In function 'List* createEmptyList()':
2.cpp:29:1: warning: no return statement in function returning non-void [-Wreturn-type]
29 | }
| ^
Press [1] To continue: Press [2] To exit: 1
Please input [1, 2, 3, 5, 0, -1] these number one by one: 1
1
Press [1] To continue: Press [2] To exit: 1
Please input [1, 2, 3, 5, 0, -1] these number one by one: 2
1
2
Press [1] To continue: Press [2] To exit: 1
Please input [1, 2, 3, 5, 0, -1] these number one by one: 3
1
2
3
Press [1] To continue: Press [2] To exit: 1
Please input [1, 2, 3, 5, 0, -1] these number one by one: 5
1
2
3
5
Press [1] To continue: Press [2] To exit: 1
Please input [1, 2, 3, 5, 0, -1] these number one by one: 0
1
2
3
5
0
Press [1] To continue: Press [2] To exit: 1
Please input [1, 2, 3, 5, 0, -1] these number one by one: -1
1
2
3
5
0
-1
Press [1] To continue: Press [2] To exit: 2
Thank you
PS C:\Users\panha\OneDrive\Documents\GitHub\DSA\Week 4 Lab>
powershell
Code
Ln 29, Col 2 Spaces: 4 UTF-8 CRLF C++ Go Live Win32 Prettier
Type here to search
30°C Partly cloudy
ENG INTL 19:37 28/10/2022

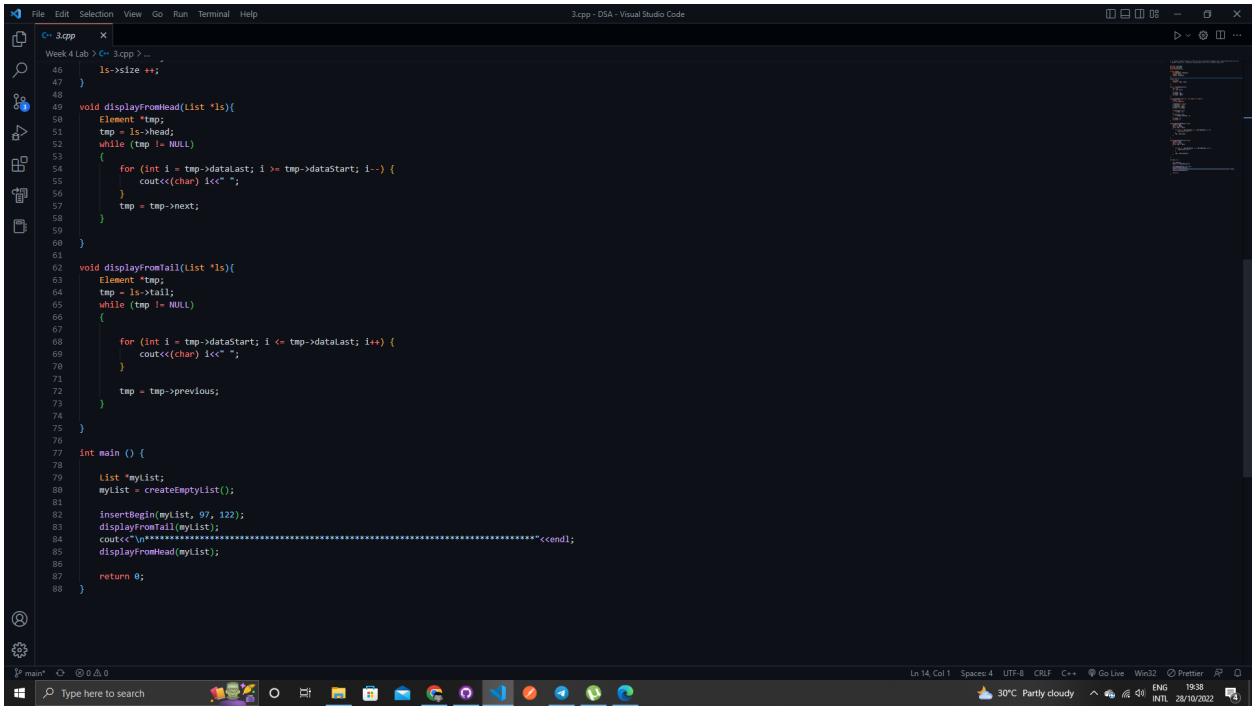

File Edit Selection View Go Run Terminal Help
Ex2-phatpanhareach.txt - DSA - Visual Studio Code
Ex2-phatpanhareach.txt U
Week 4 Lab > Ex2-phatpanhareach.txt
1
2
3
5
0
-1
Ln 7, Col 1 Spaces: 4 UTF-8 CRLF Plain Text Go Live Prettier
Type here to search
30°C Partly cloudy
ENG INTL 19:37 28/10/2022

Exercise 3:

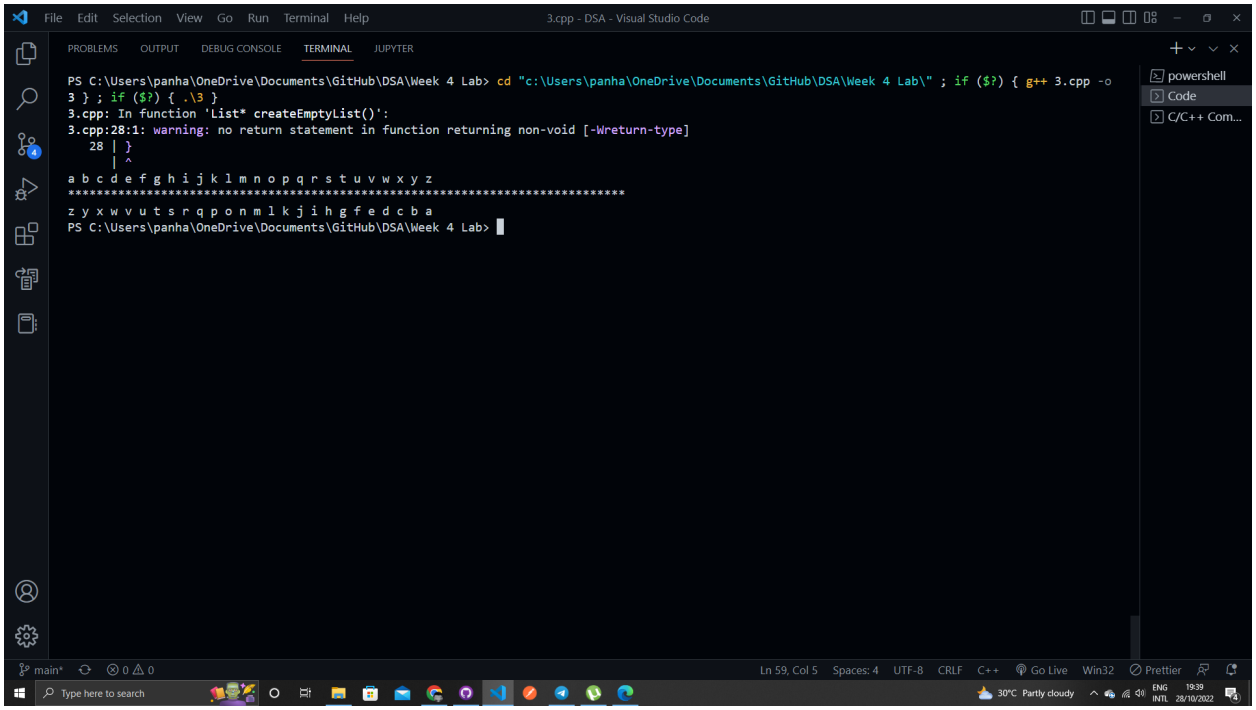
3. Create a Double linked list that can stores the English alphabet (A-Z). Display data from A to Z in double linked list. In addition, display data from Z to A in double linked list.



```cpp
#include <iostream>
#include <fstream>
using namespace std;

struct Element {
    int dataStart, dataLast;
    Element *next;
    Element *previous;
};
struct List {
    int size;
    Element * head, *tail;

};

List * createEmptyList(){
    List *ls;
    ls = new List();

    ls->size = 0;
    ls->head = NULL;
    ls->tail = NULL;
}

void insertBegin(List *ls , int start, int last) {
    Element * e;
    e = new Element();

    e->dataStart = start;
    e->dataLast = last;
    e->previous = NULL;
    e->next = ls->head;

    if(ls->size == 0) {
        ls->tail = e;
    }
    if(ls->size != 0){
        ls->head->previous = e;
    }
    ls->head = e;
    ls->size ++;
}

void displayFromHead(List *ls){
    Element *tmp;
    tmp = ls->head;
```



```
    ls->size ++;
}

void displayFromHead(List *ls){
    Element *tmp;
    tmp = ls->head;
    while (tmp != NULL)
    {
        for (int i = tmp->dataLast; i >= tmp->dataStart; i--) {
            cout<<(char) i<<" ";
        }
        tmp = tmp->next;
    }

}

void displayFromTail(List *ls){
    Element *tmp;
    tmp = ls->tail;
    while (tmp != NULL)
    {

        for (int i = tmp->dataStart; i <= tmp->dataLast; i++) {
            cout<<(char) i<<" ";
        }

        tmp = tmp->previous;
    }

}

int main () {

    List *myList;
    myList = createEmptyList();

    insertBegin(myList, 97, 122);
    displayFromTail(myList);
    cout<<"\n**************************************************************************"<<endl;
    displayFromHead(myList);

    return 0;
}
```



```
PS C:\Users\panha\OneDrive\Documents\GitHub\DSA\Week 4 Lab> cd "c:\Users\panha\OneDrive\Documents\GitHub\DSA\Week 4 Lab\" ; if ($?) { g++ 3.cpp -o
3 } ; if ($?) { .\3 }
3.cpp: In function 'List* createEmptyList()':
3.cpp:28:1: warning: no return statement in function returning non-void [-Wreturn-type]
   28 | }
      | ^
a b c d e f g h i j k l m n o p q r s t u v w x y z
**************************************************************************
z y x w v u t s r q p o n m l k j i h g f e d c b a
PS C:\Users\panha\OneDrive\Documents\GitHub\DSA\Week 4 Lab>
```

Exercise 4:

4. Create a Double linked list that can store integer numbers. Write functions below and test in your main program to implement double link list

a. Create a function to add a number entered by a user to the end of the list

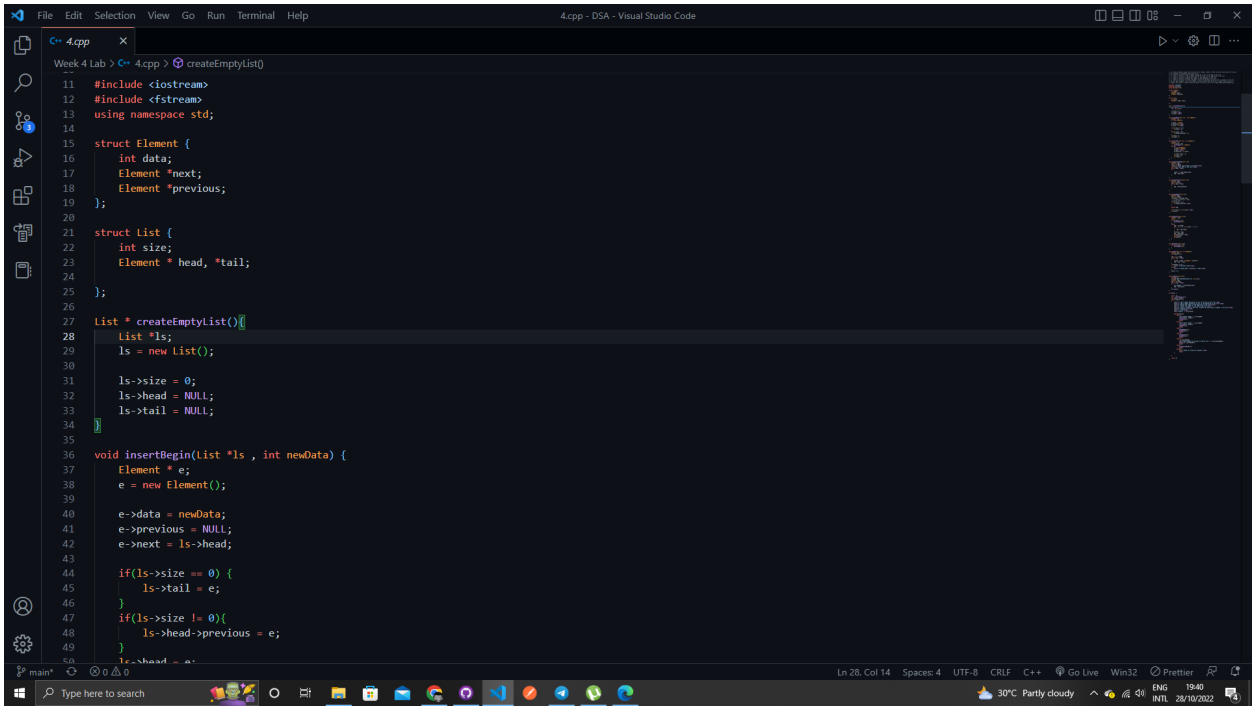
b. Create a function to add a number entered by a user to the beginning of the list

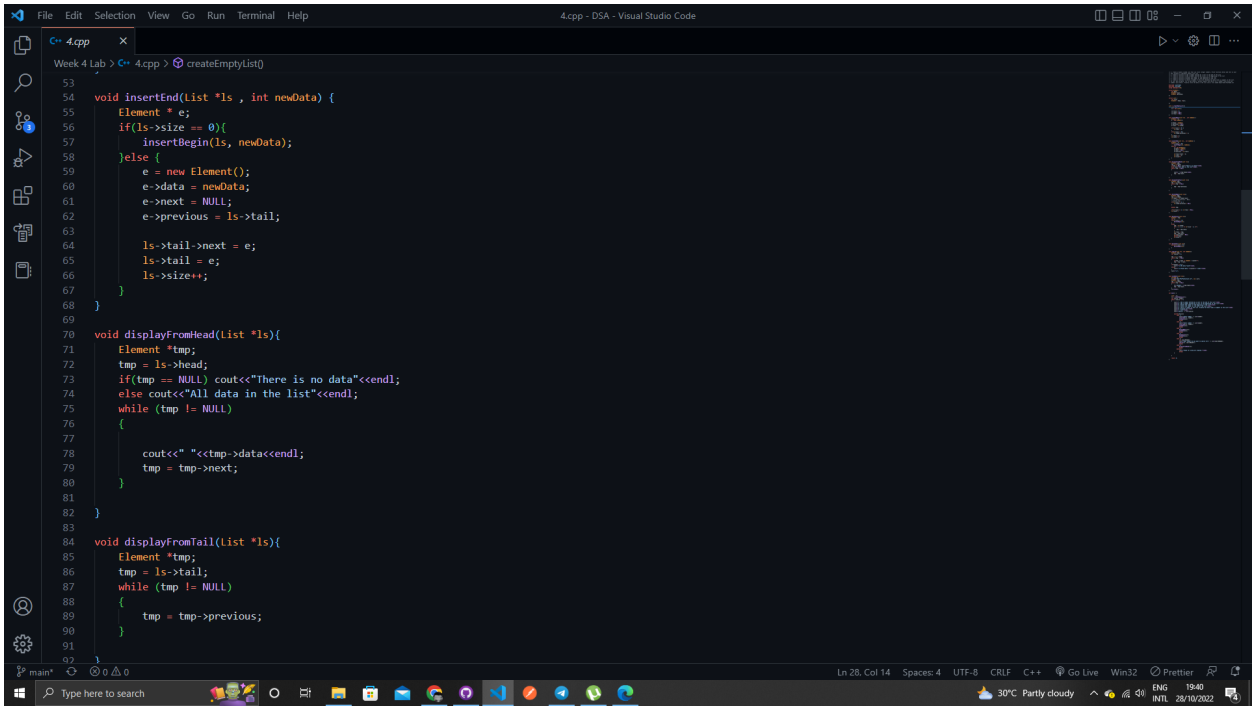
c. Create a function to delete the number at the beginning of the list

d. Create a function to delete the number at the beginning of the list

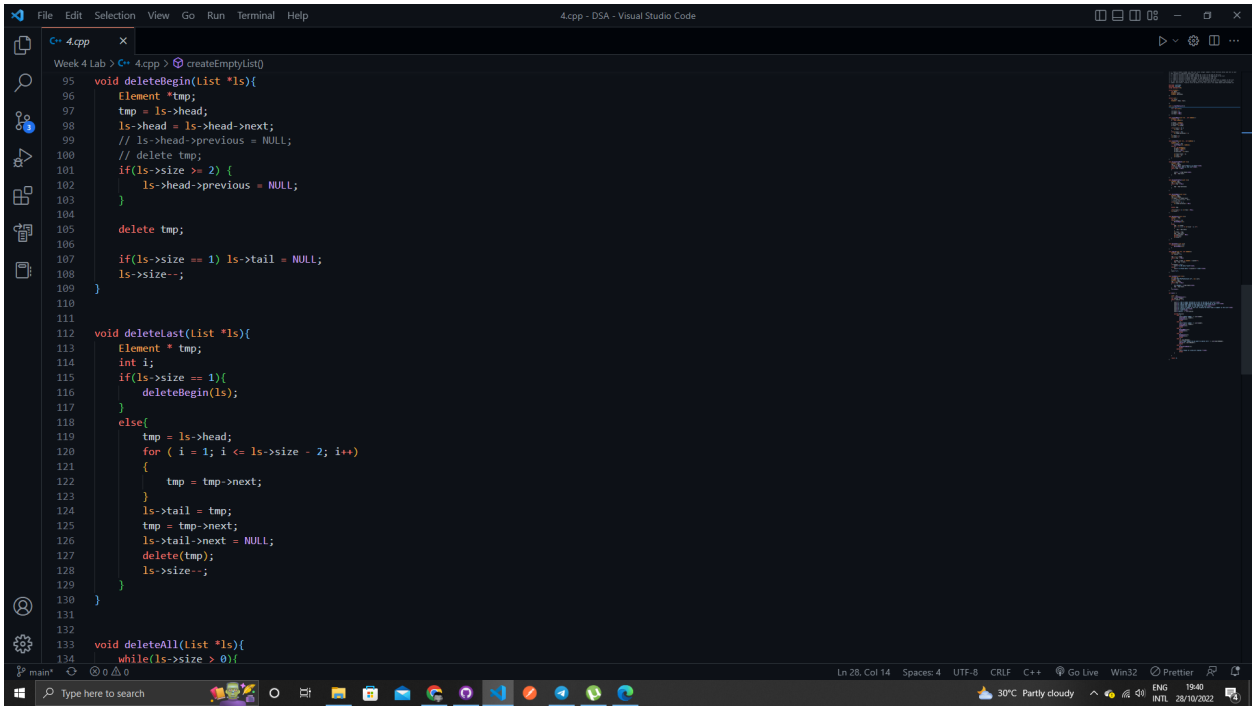
e. Create a function to search a number in the list. Display how many times in appear in the list.

In your main program, try to add numbers, delete numbers and search for a number (test with any numbers you prefer). Finally store the data in the list into a file named Ouput-Ex3e-YourName.txt

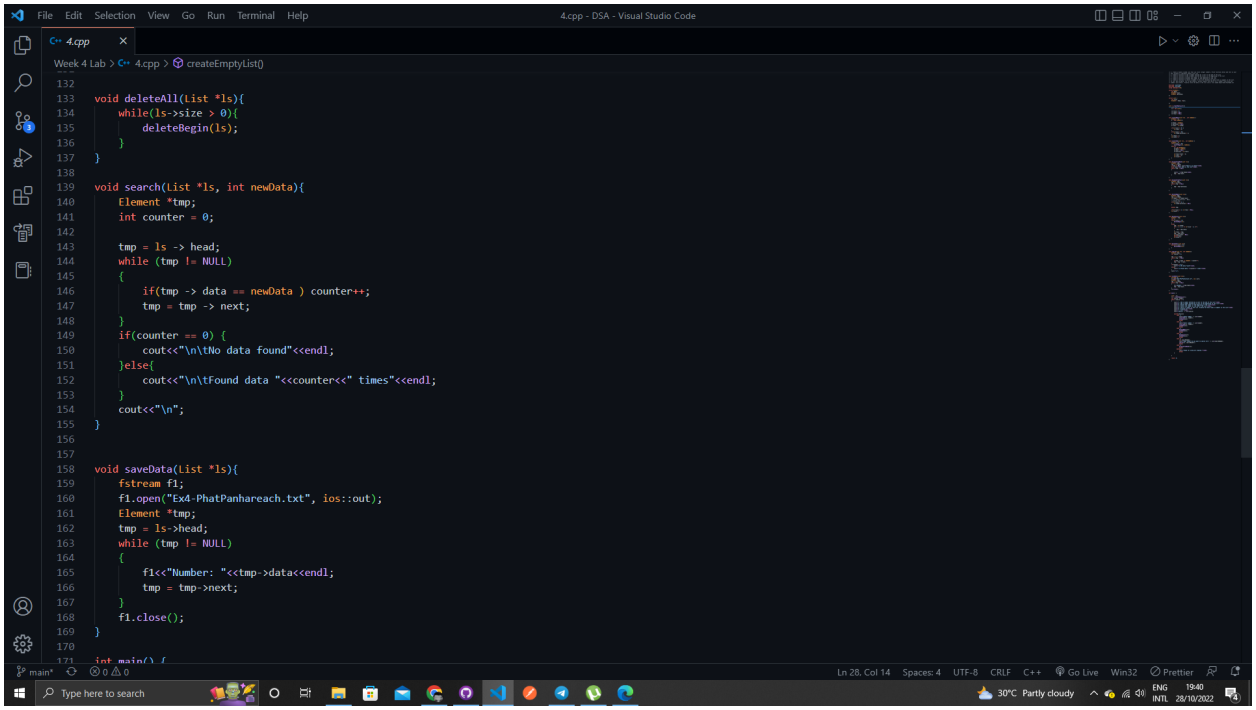




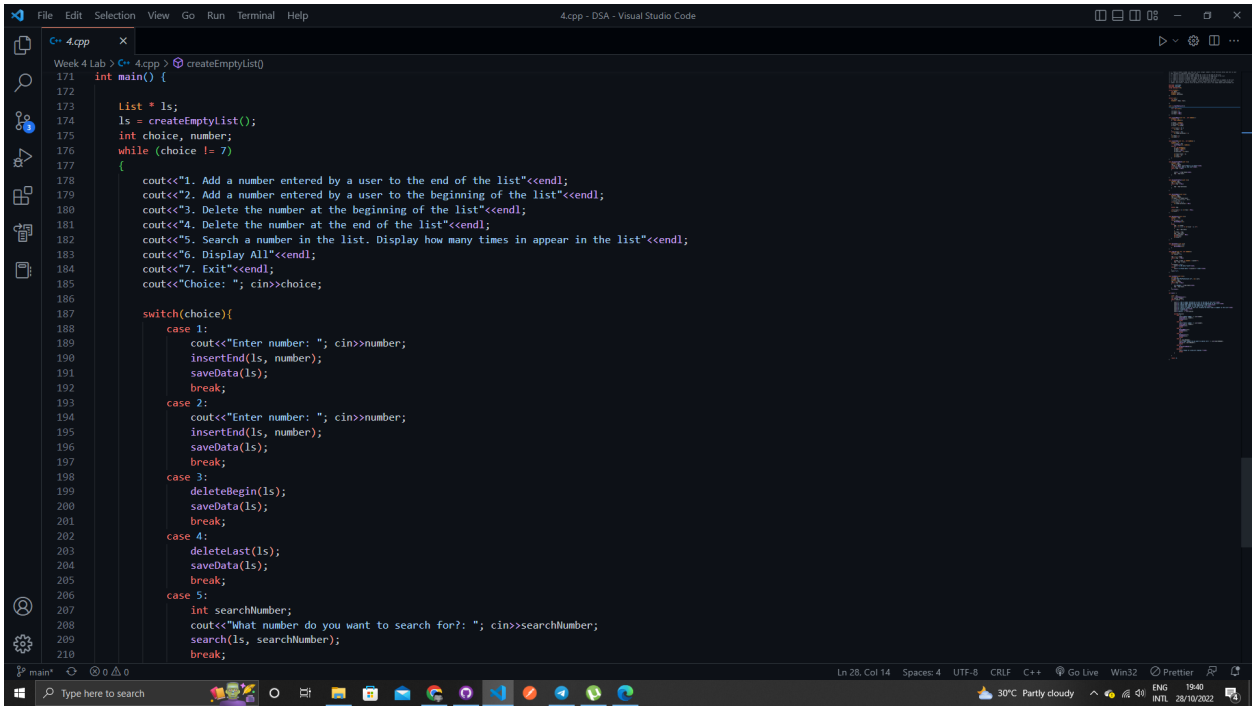
```
53
54  void insertEnd(List *ls , int newData) {
55      Element * e;
56      if(ls->size == 0){
57          insertBegin(ls, newData);
58      }else {
59          e = new Element();
60          e->data = newData;
61          e->next = NULL;
62          e->previous = ls->tail;
63
64          ls->tail->next = e;
65          ls->tail = e;
66          ls->size++;
67      }
68  }
69
70  void displayFromHead(List *ls){
71      Element *tmp;
72      tmp = ls->head;
73      if(tmp == NULL) cout<<"There is no data"<<endl;
74      else cout<<"All data in the list"<<endl;
75      while (tmp != NULL)
76      {
77
78          cout<<" "<<tmp->data<<endl;
79          tmp = tmp->next;
80      }
81
82  }
83
84  void displayFromTail(List *ls){
85      Element *tmp;
86      tmp = ls->tail;
87      while (tmp != NULL)
88      {
89          tmp = tmp->previous;
90      }
91
92  }
```



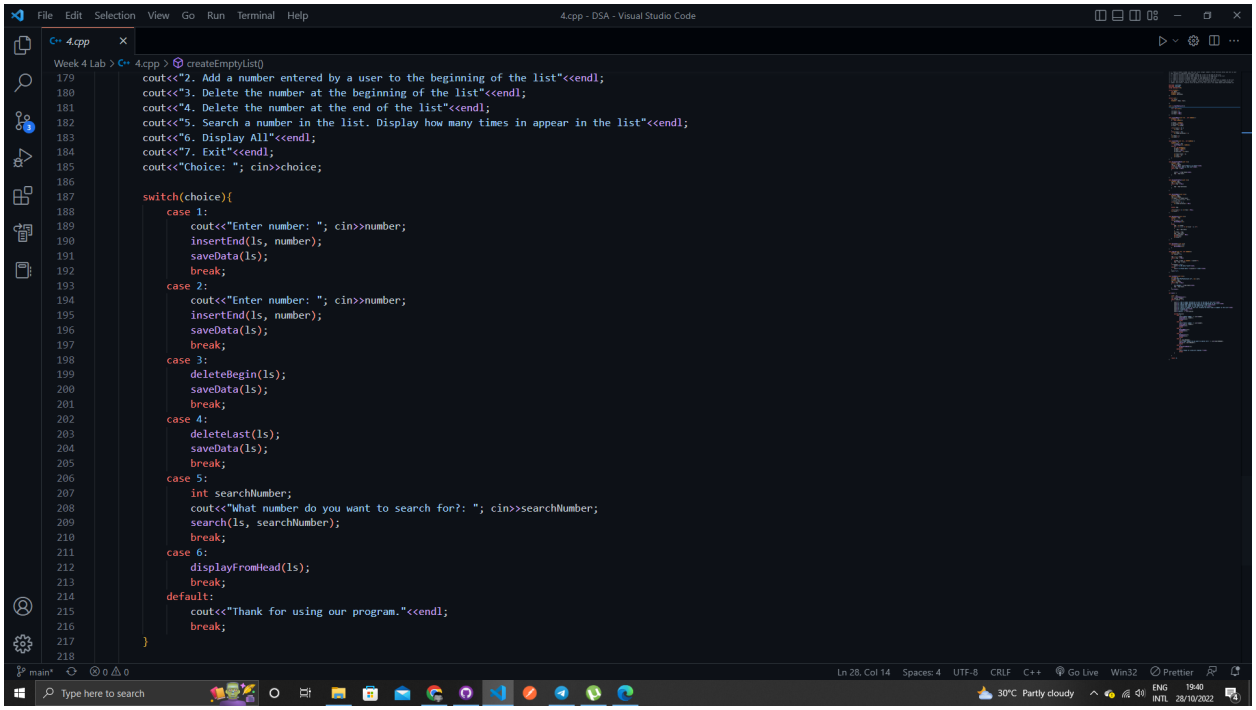
```
95  void deleteBegin(List *ls){
96      Element *tmp;
97      tmp = ls->head;
98      ls->head = ls->head->next;
99      // ls->head->previous = NULL;
100     // delete tmp;
101     if(ls->size >= 2) {
102         ls->head->previous = NULL;
103     }
104
105     delete tmp;
106
107     if(ls->size == 1) ls->tail = NULL;
108     ls->size--;
109 }
110
111
112 void deleteLast(List *ls){
113     Element * tmp;
114     int i;
115     if(ls->size == 1){
116         deleteBegin(ls);
117     }
118     else{
119         tmp = ls->head;
120         for ( i = 1; i <= ls->size - 2; i++)
121         {
122             tmp = tmp->next;
123         }
124         ls->tail = tmp;
125         tmp = tmp->next;
126         ls->tail->next = NULL;
127         delete(tmp);
128         ls->size--;
129     }
130 }
131
132
133 void deleteAll(List *ls){
134     while(ls->size > 0){
```



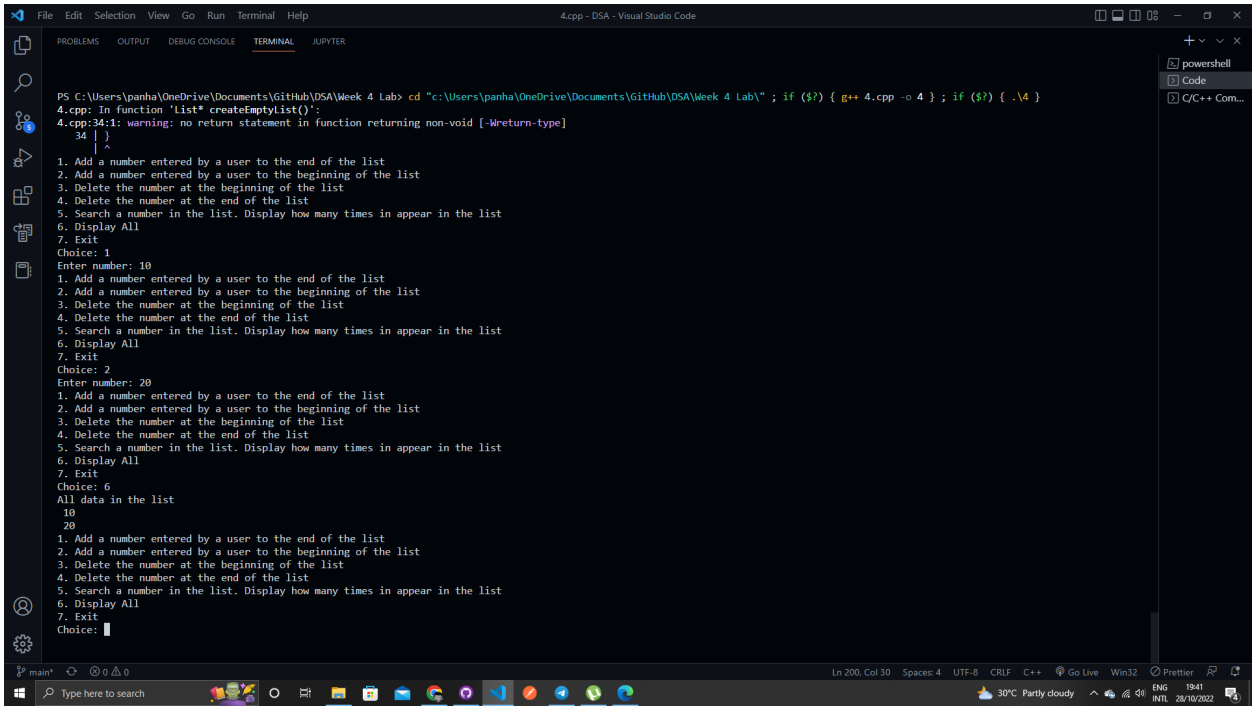
```
132
133 void deleteAll(List *ls){
134     while(ls->size > 0){
135         deleteBegin(ls);
136     }
137 }
138
139 void search(List *ls, int newData){
140     Element *tmp;
141     int counter = 0;
142
143     tmp = ls -> head;
144     while (tmp != NULL)
145     {
146         if(tmp -> data == newData ) counter++;
147         tmp = tmp -> next;
148     }
149     if(counter == 0) {
150         cout<<"\n\tNo data found"<<endl;
151     }else{
152         cout<<"\n\tFound data "<<counter<<" times"<<endl;
153     }
154     cout<<"\n";
155 }
156
157
158 void saveData(List *ls){
159     fstream f1;
160     f1.open("Ex4-PhatPanhareach.txt", ios::out);
161     Element *tmp;
162     tmp = ls->head;
163     while (tmp != NULL)
164     {
165         f1<<"Number: "<<tmp->data<<endl;
166         tmp = tmp->next;
167     }
168     f1.close();
169 }
170
171 int main() {
```



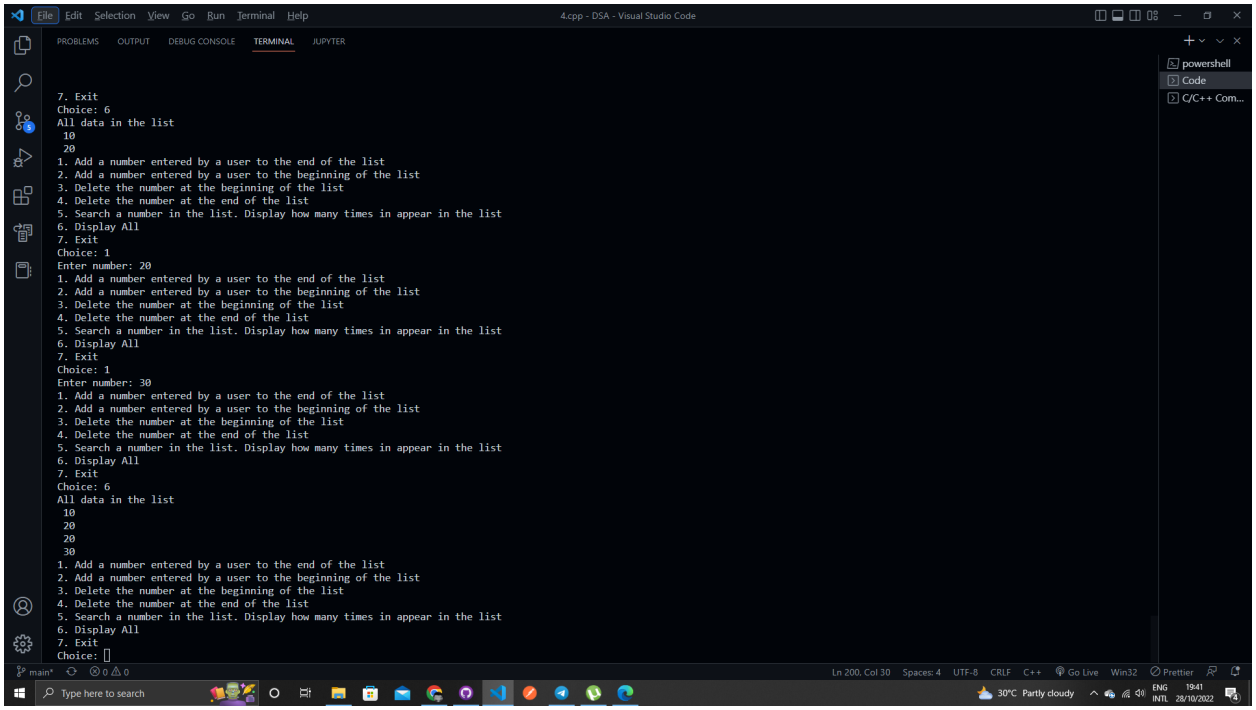
```
171 int main() {
172
173     List * ls;
174     ls = createEmptyList();
175     int choice, number;
176     while (choice != 7)
177     {
178         cout<<"1. Add a number entered by a user to the end of the list"<<endl;
179         cout<<"2. Add a number entered by a user to the beginning of the list"<<endl;
180         cout<<"3. Delete the number at the beginning of the list"<<endl;
181         cout<<"4. Delete the number at the end of the list"<<endl;
182         cout<<"5. Search a number in the list. Display how many times in appear in the list"<<endl;
183         cout<<"6. Display All"<<endl;
184         cout<<"7. Exit"<<endl;
185         cout<<"Choice: "; cin>>choice;
186
187         switch(choice){
188             case 1:
189                 cout<<"Enter number: "; cin>>number;
190                 insertEnd(ls, number);
191                 saveData(ls);
192                 break;
193             case 2:
194                 cout<<"Enter number: "; cin>>number;
195                 insertEnd(ls, number);
196                 saveData(ls);
197                 break;
198             case 3:
199                 deleteBegin(ls);
200                 saveData(ls);
201                 break;
202             case 4:
203                 deleteLast(ls);
204                 saveData(ls);
205                 break;
206             case 5:
207                 int searchNumber;
208                 cout<<"What number do you want to search for?: "; cin>>searchNumber;
209                 search(ls, searchNumber);
210                 break;
```

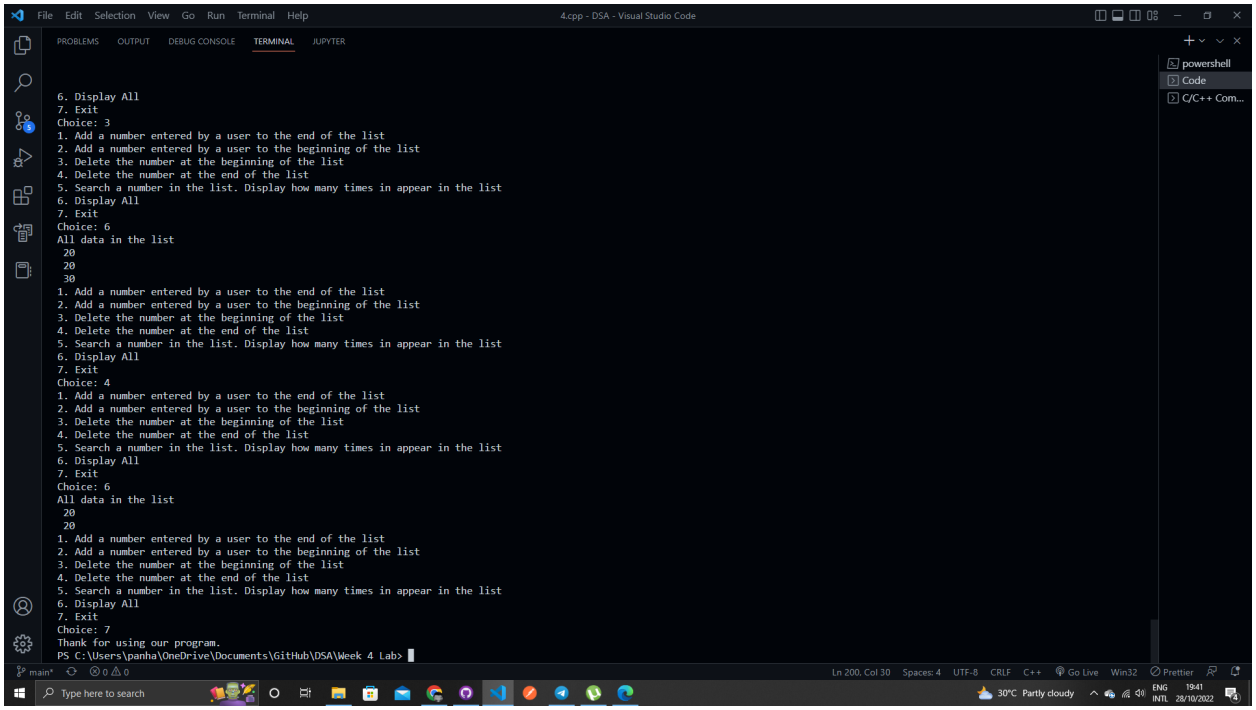


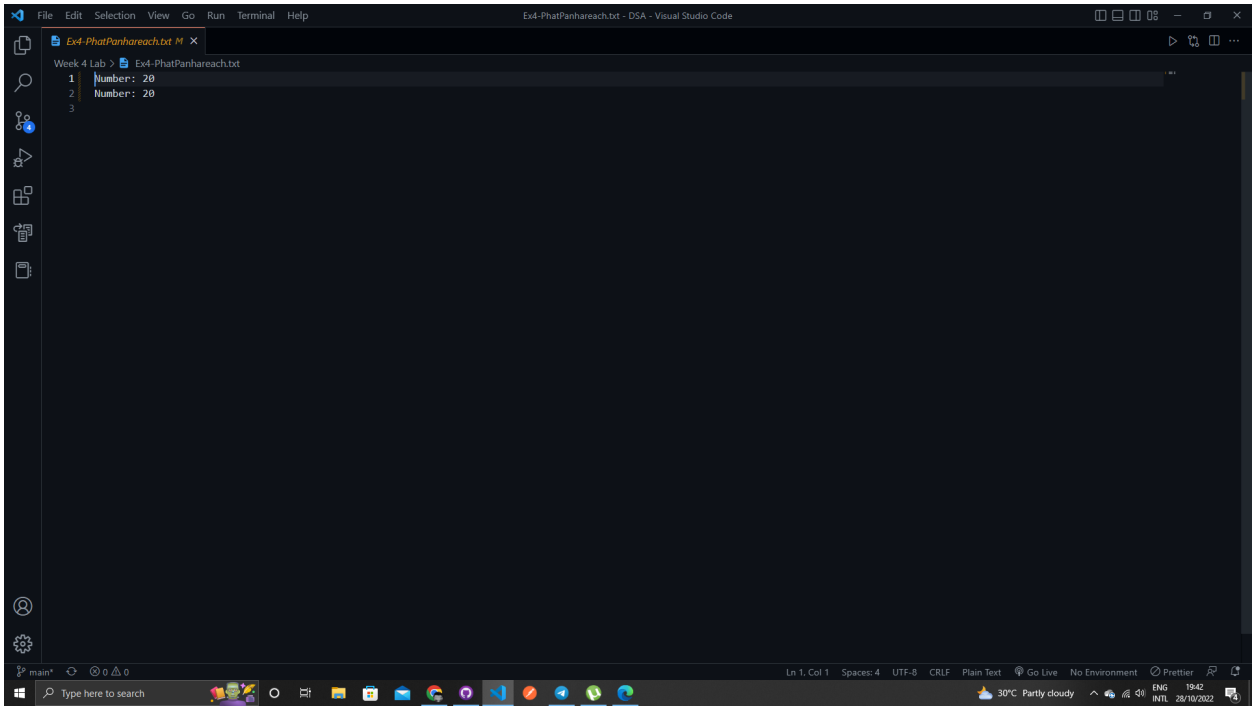
```
179         cout<<"2. Add a number entered by a user to the beginning of the list"<<endl;
180         cout<<"3. Delete the number at the beginning of the list"<<endl;
181         cout<<"4. Delete the number at the end of the list"<<endl;
182         cout<<"5. Search a number in the list. Display how many times in appear in the list"<<endl;
183         cout<<"6. Display All"<<endl;
184         cout<<"7. Exit"<<endl;
185         cout<<"Choice: "; cin>>choice;
186 
187         switch(choice){
188             case 1:
189                 cout<<"Enter number: "; cin>>number;
190                 insertEnd(ls, number);
191                 saveData(ls);
192                 break;
193             case 2:
194                 cout<<"Enter number: "; cin>>number;
195                 insertEnd(ls, number);
196                 saveData(ls);
197                 break;
198             case 3:
199                 deleteBegin(ls);
200                 saveData(ls);
201                 break;
202             case 4:
203                 deleteLast(ls);
204                 saveData(ls);
205                 break;
206             case 5:
207                 int searchNumber;
208                 cout<<"What number do you want to search for?: "; cin>>searchNumber;
209                 search(ls, searchNumber);
210                 break;
211             case 6:
212                 displayFromHead(ls);
213                 break;
214             default:
215                 cout<<"Thank for using our program."<<endl;
216                 break;
217         }
218 
```



```
PS C:\Users\panha\OneDrive\Documents\GitHub\DSA\Week 4 Lab> cd "c:\Users\panha\OneDrive\Documents\GitHub\DSA\Week 4 Lab\" ; if ($?) { g++ 4.cpp -o 4 } ; if ($?) { .\4 }
4.cpp: In function 'List* createEmptyList()':
4.cpp:34:1: warning: no return statement in function returning non-void [-Wreturn-type]
   34 | }
      | ^
1. Add a number entered by a user to the end of the list
2. Add a number entered by a user to the beginning of the list
3. Delete the number at the beginning of the list
4. Delete the number at the end of the list
5. Search a number in the list. Display how many times in appear in the list
6. Display All
7. Exit
Choice: 1
Enter number: 10
1. Add a number entered by a user to the end of the list
2. Add a number entered by a user to the beginning of the list
3. Delete the number at the beginning of the list
4. Delete the number at the end of the list
5. Search a number in the list. Display how many times in appear in the list
6. Display All
7. Exit
Choice: 2
Enter number: 20
1. Add a number entered by a user to the end of the list
2. Add a number entered by a user to the beginning of the list
3. Delete the number at the beginning of the list
4. Delete the number at the end of the list
5. Search a number in the list. Display how many times in appear in the list
6. Display All
7. Exit
Choice: 6
All data in the list
 10
 20
1. Add a number entered by a user to the end of the list
2. Add a number entered by a user to the beginning of the list
3. Delete the number at the beginning of the list
4. Delete the number at the end of the list
5. Search a number in the list. Display how many times in appear in the list
6. Display All
7. Exit
Choice:
```


File Edit Selection View Go Run Terminal Help
4.cpp - DSA - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
powershell
Code
C/C++ Com...
7. Exit
Choice: 6
All data in the list
10
20
1. Add a number entered by a user to the end of the list
2. Add a number entered by a user to the beginning of the list
3. Delete the number at the beginning of the list
4. Delete the number at the end of the list
5. Search a number in the list. Display how many times in appear in the list
6. Display All
7. Exit
Choice: 1
Enter number: 20
1. Add a number entered by a user to the end of the list
2. Add a number entered by a user to the beginning of the list
3. Delete the number at the beginning of the list
4. Delete the number at the end of the list
5. Search a number in the list. Display how many times in appear in the list
6. Display All
7. Exit
Choice: 1
Enter number: 30
1. Add a number entered by a user to the end of the list
2. Add a number entered by a user to the beginning of the list
3. Delete the number at the beginning of the list
4. Delete the number at the end of the list
5. Search a number in the list. Display how many times in appear in the list
6. Display All
7. Exit
Choice: 6
All data in the list
10
20
20
30
1. Add a number entered by a user to the end of the list
2. Add a number entered by a user to the beginning of the list
3. Delete the number at the beginning of the list
4. Delete the number at the end of the list
5. Search a number in the list. Display how many times in appear in the list
6. Display All
7. Exit
Choice:
main*
Ln 200, Col 30 Spaces: 4 UTF-8 CRLF C++ Go Live Win32 Prettier
Type here to search
30°C Partly cloudy
ENG INTL 19:41 28/10/2022


File Edit Selection View Go Run Terminal Help
4.cpp - DSA - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
powershell
Code
C/C++ Com...
6. Display All
7. Exit
Choice: 3
1. Add a number entered by a user to the end of the list
2. Add a number entered by a user to the beginning of the list
3. Delete the number at the beginning of the list
4. Delete the number at the end of the list
5. Search a number in the list. Display how many times in appear in the list
6. Display All
7. Exit
Choice: 6
All data in the list
20
20
30
1. Add a number entered by a user to the end of the list
2. Add a number entered by a user to the beginning of the list
3. Delete the number at the beginning of the list
4. Delete the number at the end of the list
5. Search a number in the list. Display how many times in appear in the list
6. Display All
7. Exit
Choice: 4
1. Add a number entered by a user to the end of the list
2. Add a number entered by a user to the beginning of the list
3. Delete the number at the beginning of the list
4. Delete the number at the end of the list
5. Search a number in the list. Display how many times in appear in the list
6. Display All
7. Exit
Choice: 6
All data in the list
20
20
1. Add a number entered by a user to the end of the list
2. Add a number entered by a user to the beginning of the list
3. Delete the number at the beginning of the list
4. Delete the number at the end of the list
5. Search a number in the list. Display how many times in appear in the list
6. Display All
7. Exit
Choice: 7
Thank for using our program.
PS C:\Users\panha\OneDrive\Documents\GitHub\DSA\Week 4 Lab>
main*
Ln 200, Col 30 Spaces: 4 UTF-8 CRLF C++ Go Live Win32 Prettier
Type here to search
30°C Partly cloudy
ENG INTL 19:41 28/10/2022


File Edit Selection View Go Run Terminal Help
Ex4-PhatPanhareach.txt - DSA - Visual Studio Code
Ex4-PhatPanhareach.txt M
Week 4 Lab > Ex4-PhatPanhareach.txt
1 Number: 20
2 Number: 20
3
main*
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF Plain Text Go Live No Environment Prettier
Type here to search
30°C Partly cloudy
ENG INTL
19:42 28/10/2022