Sample Test Cases for the Assignment 2 programs:

Question 1

TC1

GC_List = {Ali, Usman, Haider, Maryam, Masooma, Urooj}
FAS_List = {Ashiq, Manika, Ali, Masooma, Akbar, Urooj}
FDS_List = {Masooma, Bilal, Amna, Madiha, Rohail, Urooj}
Output

List_of_all_student (string array) = {Ali, Usman, Haider, Maryam, Masooma, Urooj, Ashiq, Manika, Akbar, Bilal, Amna, Madhiha, Rohail}
List_of_common_student (string array): {Masooma, Urooj}

TC2

GC_List = {Ali, Usman, Haider}
FAS_List = {Ashiq, Manika, Ali, Masooma }
FDS_List = {Masooma, Bilal, Amna , Rohail, Urooj}
Output

List_of_all_student (string array) = {Ali, Usman, Haider, Ashiq, Manika, Masoona, Bilal, Amna, Rohail, Urooj}
List_of_common_student (string array): There are no common students

TC3

GC_List = {Ali, Usman, Haider}
FAS_List = null
FDS_List = null

Output

List_of_all_student (string array) = {Ali, Usman, Haider}
List_of_common_student (string array): There are no common students

TC4

GC_List = {Ali, Usman, Haider}
FAS_List = {Masooma }
FDS_List = null

Output

List_of_all_student (string array) = {Ali, Usman, Haider, Masooma}
: List_of_common_student (string array): There are no common students

Question 2

Input List: 12 ➔11 ➔13 ➔16 ➔14➔22➔11➔12

Find Number 16

If not sorted, then sort

11 ➔11➔12➔12➔13➔14➔16➔22

The find mid

No of elements / 2 = 8/2 = 4

So mid will be at 4 (12). Shift head.

Now again find mid 4/2, (4 is the size of remaining list and last mid) new mid will be 2 so move mid to 14 and so on.

If element found, print result found else print result not found

Question 3

Command	Output
Go to floor 4 (**lift_operating_system**)	Current = 4. Lift is at 4th floor
Skip floor 3 (**skip_floor**)	Floor 3 will skip
Go to floor 3 (**lift_operating_system**)	Cannot go to floor 3
Halt llife	Lift has been halted
Go to floor 2 (**lift_operating_system**)	Lift is halted
Go to floor 3 (**lift_operating_system**)	Lift is halted
Un Halt lift	Lift is functional again
Go to floor 3	Cannot go to floor 3
Make floor 3 operatioal again (**make_floor_operational)**	You can go to floor 3 now
Go to floor -1 (**lift_operating_system**)	Lift is at basement
Add Floor	A new floor has been added at the top
Go to floor -2 (**lift_operating_system**)	Invalid floor
Go to floor -1 (**lift_operating_system**	You are already at basement
Exit	Close the program

Question 4 - String Manipulation

1. **int Calculate_length (StringList *head)**

I		a	m		a		b	o	y	.

The length of the string is 11 character.

2. **bool substring (StringList *head, String str)**

StringList = “I am taking the DS Class”

Str = “DS”

The function should return true as the DS exist in the above string.

Another example

StringList = “I am taking the DS Class”

Str = “taking”

The function should return true as it exists in the above string.

Another example

StringList = “I am taking the DS Class”

Str = “that DS”

The function should return false as it does not exist in the above string.

Another example

StringList = “I am taking the DS Class”

Str = “Taking the DS class and doing the assignment”

The function should return false as it does not exist in the above string.

3. **int substring_position (StringList *head, String str)**

The above function returns the index of the main string where the sub string starts.

For example

StringList = “I am taking DS course”

Str = “taking”

The function will return 5 as substring exist and starts at 5^{th} index of **StringList**. In case the substring does not exist, you will return -1.

4. **void replaceString (StringList *head, String find, String replace)**

This function will allow the user to replace a given string with existing string. For example

StringList: I am taking DS class.

FindText: a

ReplaceText: Y

The resultant string is: I Ym tYking DS clYss

Another example

StringList: I am taking DS class.

FindText: data

ReplaceText: Y

Error: Provided text does not exist

5. **void appendText (StringList *head, String appends, int index)**

This function will allow the user to append a string in stringllist, For example.

StringList: I am taking DS class.

Index = 5

Append text: not

Resulting = I am not taking DS class

6. **void deleteText (StringList *head, String delText)**

This function will allow the user to delete a string in stringllist, For example.

StringList: I am taking DS class.

Delete text: taking D

Resulting = I am not S class

If the deleting text does not exist, show error that the text cannot be deleted