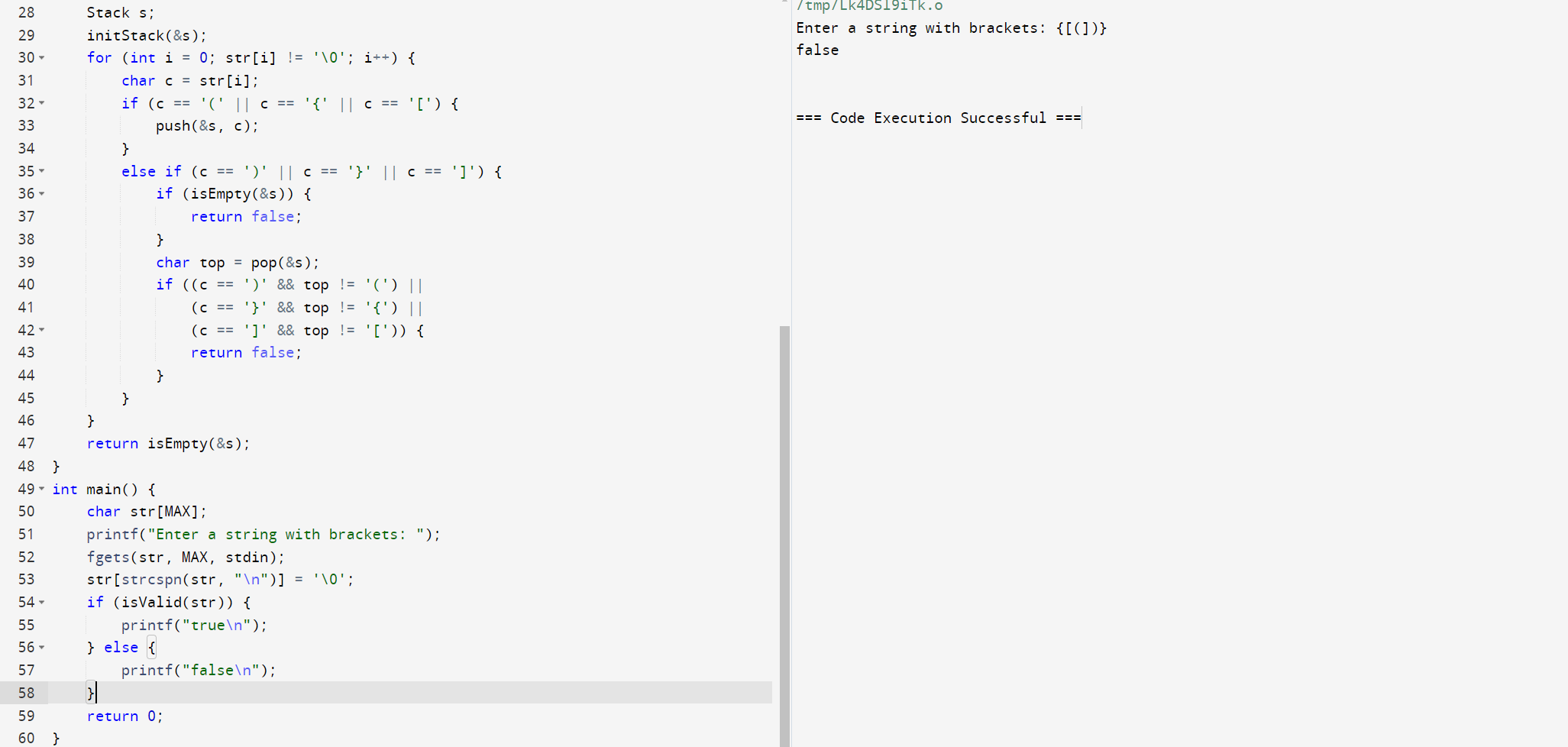
61. Write a C program to check if a given string is valid or not. The string contains the

characters '(', ')', '{', '}', '[' and ']'. The string is valid if the open brackets must be

closed with the same type of bracket and in the correct order.

Expected Output: false





62. Write a program in C to create a singly linked list of n nodes and count the

number of nodes.

Test Data :

Input the number of nodes : 3

Input data for node 1 : 5

Input data for node 2 : 6

Input data for node 3 : 7

Expected Output :

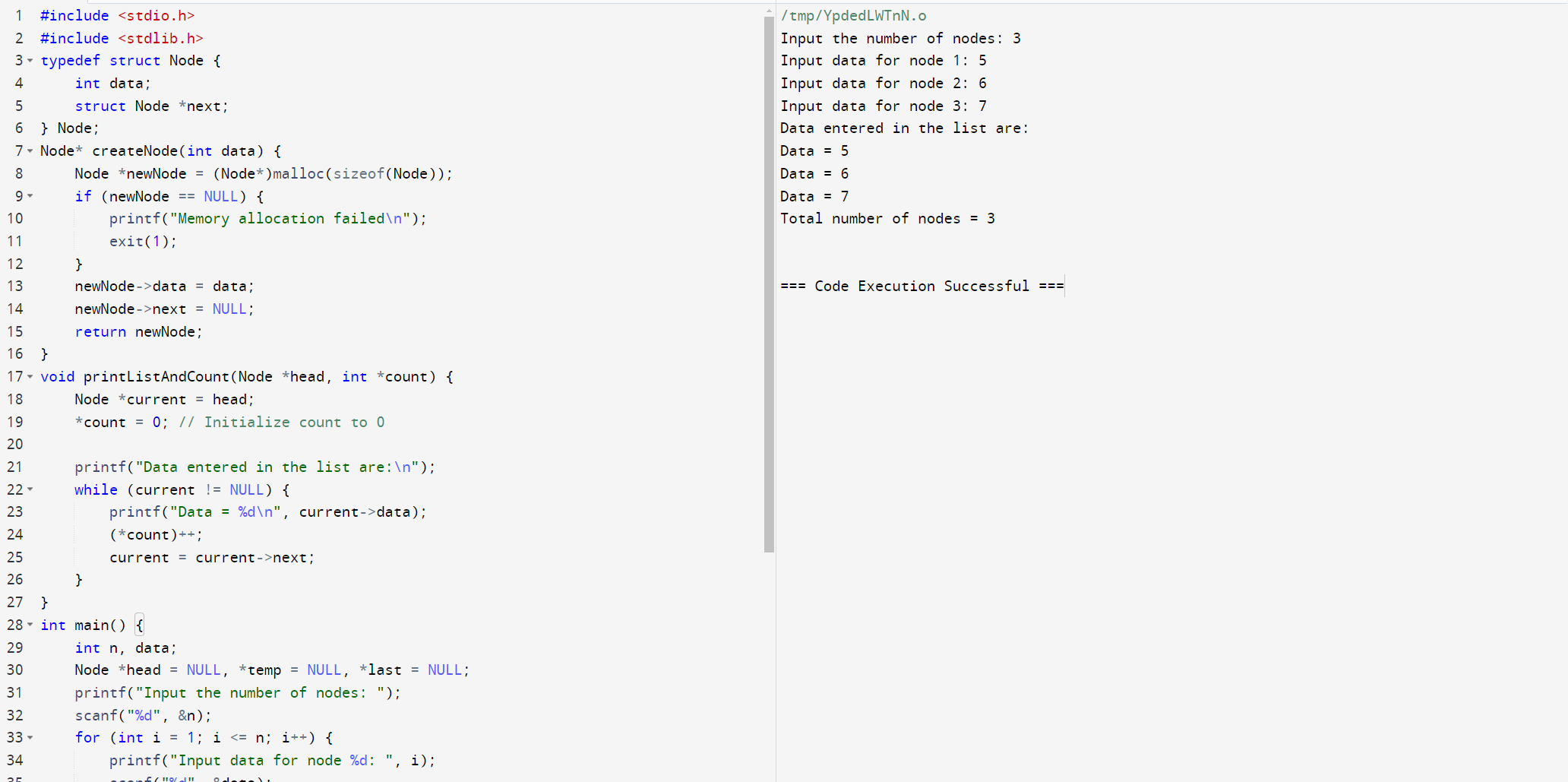
Data entered in the list are :

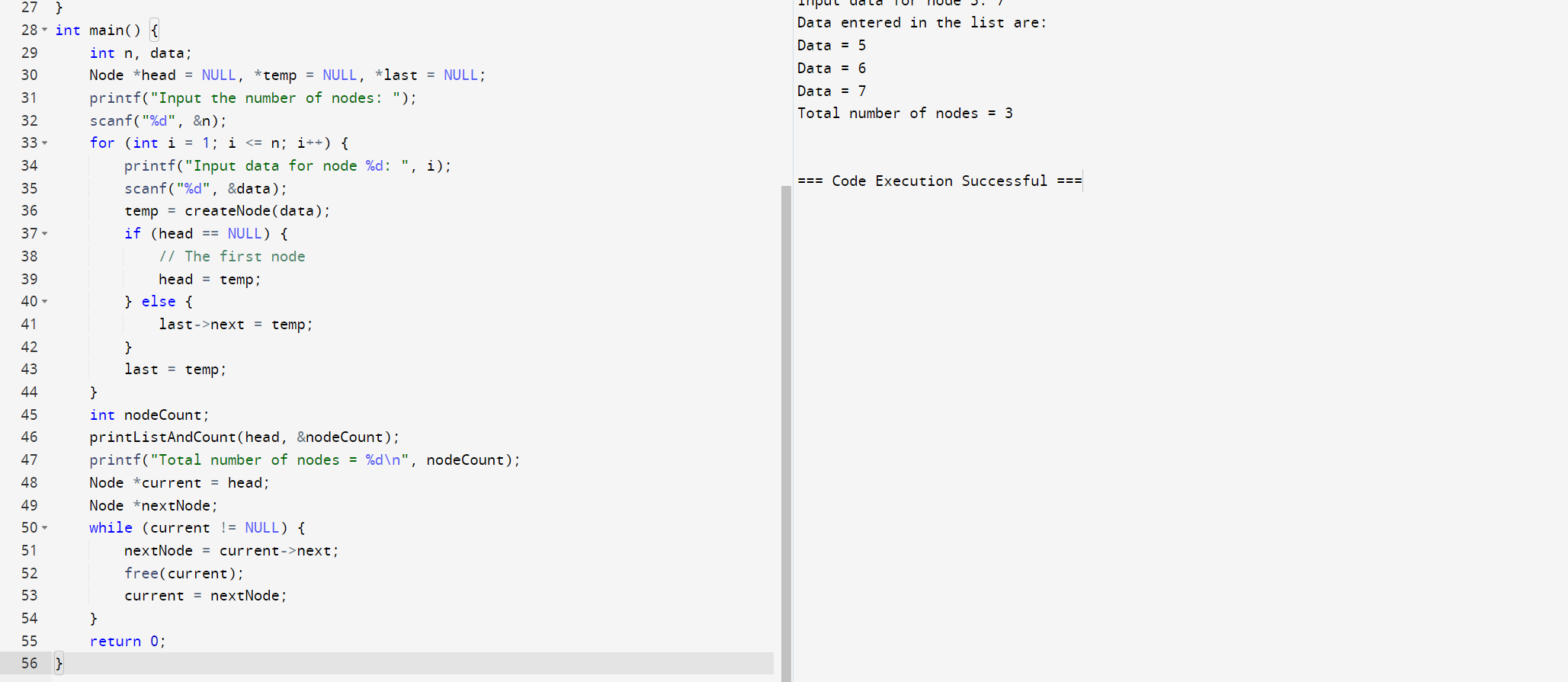
Data = 5

Data = 6

Data = 7

Total number of nodes = 3

****

****

63. Write a C program to remove duplicates from a given array of integers.. >

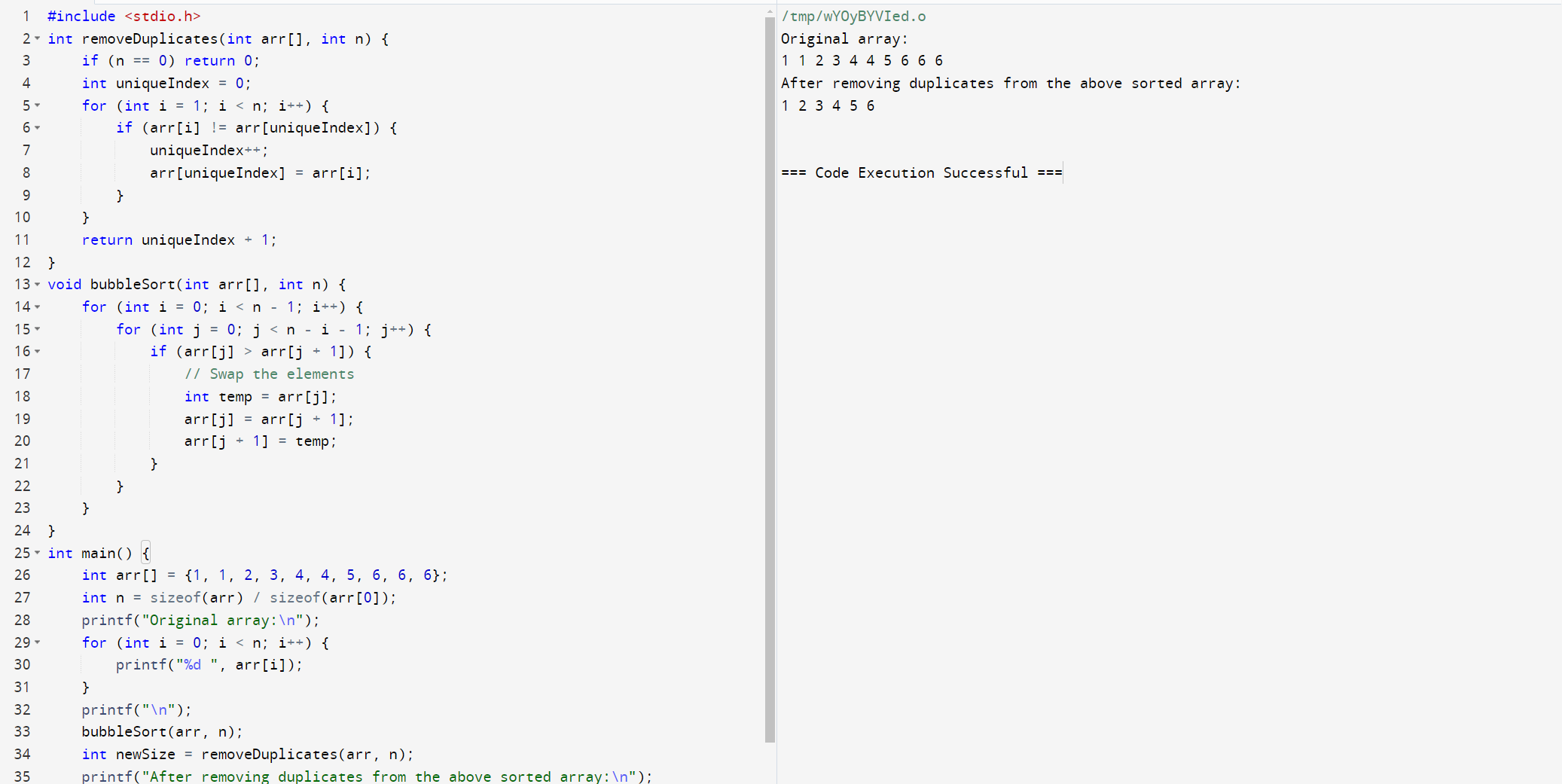
Expected Output:

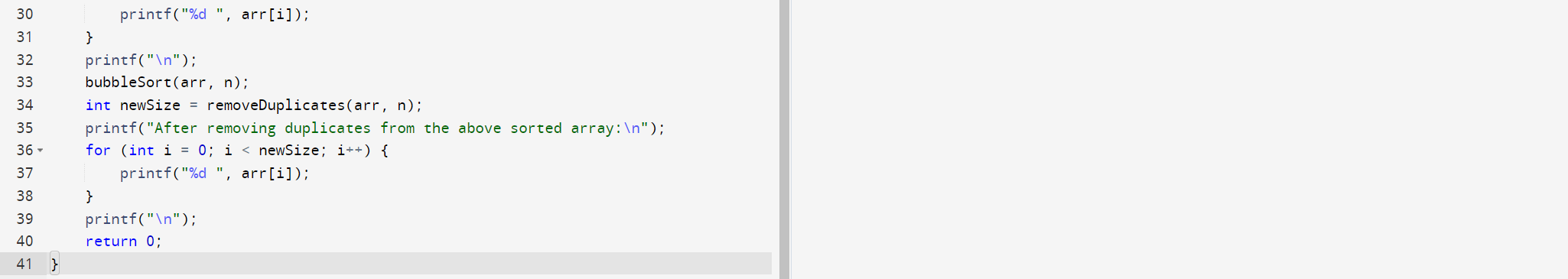
Original array:

1 1 2 3 4 4 5 6 6 6

After removing duplicates from the above sorted array:

1 2 3 4 5 6

****

****

**64.** **Write a C program to merge two sorted singly linked lists into a single sorted**

**linked list.Test Data and Expected Output :**

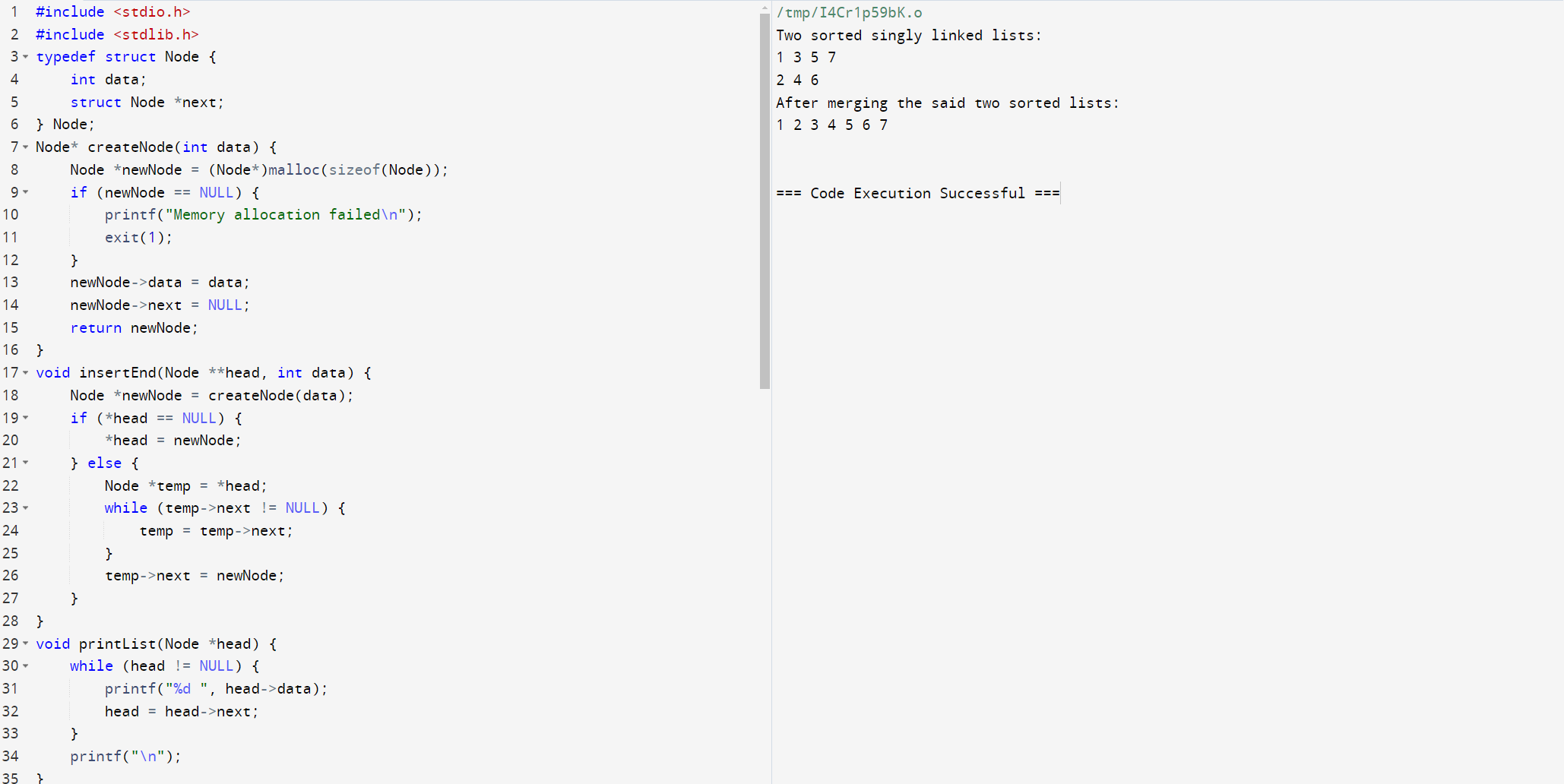
**Two sorted singly linked lists:**

**1 3 5 7**

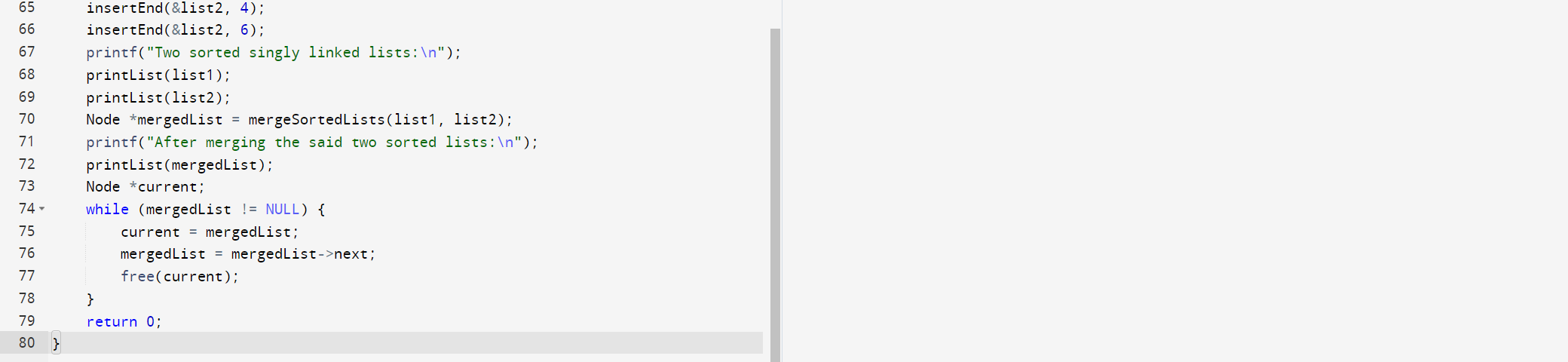
**2 4 6**

**After merging the said two sorted lists:**

**1 2 3 4 5 6 7**

****

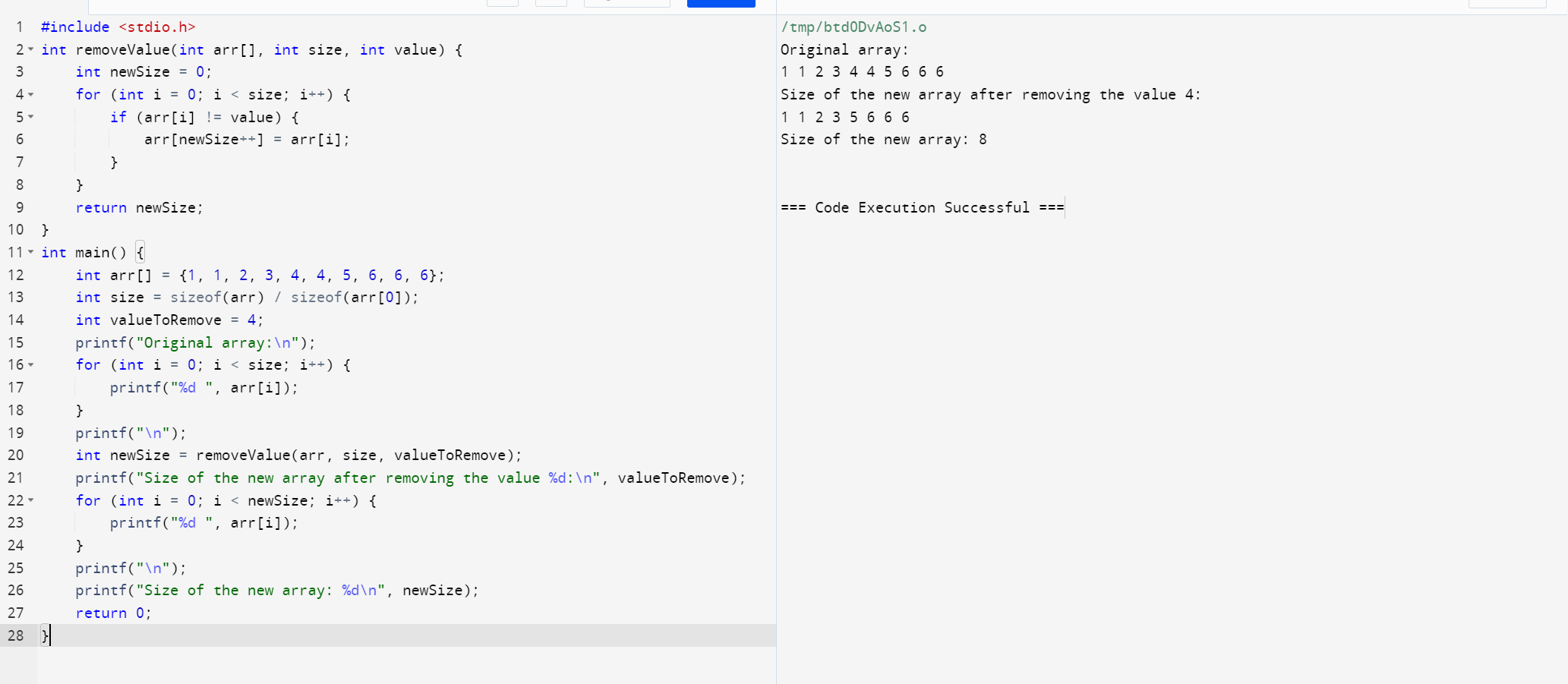
****

****

**65.** Write a C program to remove all instances of a given value from a given array of integers and return the length of the updated array. >

Expected Output:Original array: 1 1 2 3 4 4 5 6 6 6

Size of the new array after removing the value 4: 8

****

66. Write a C program to remove duplicates from a single unsorted linked list.

Test Data and Expected Output :

Original Singly List:

1 2 3 3 4

After removing duplicate elements from the said singly list:

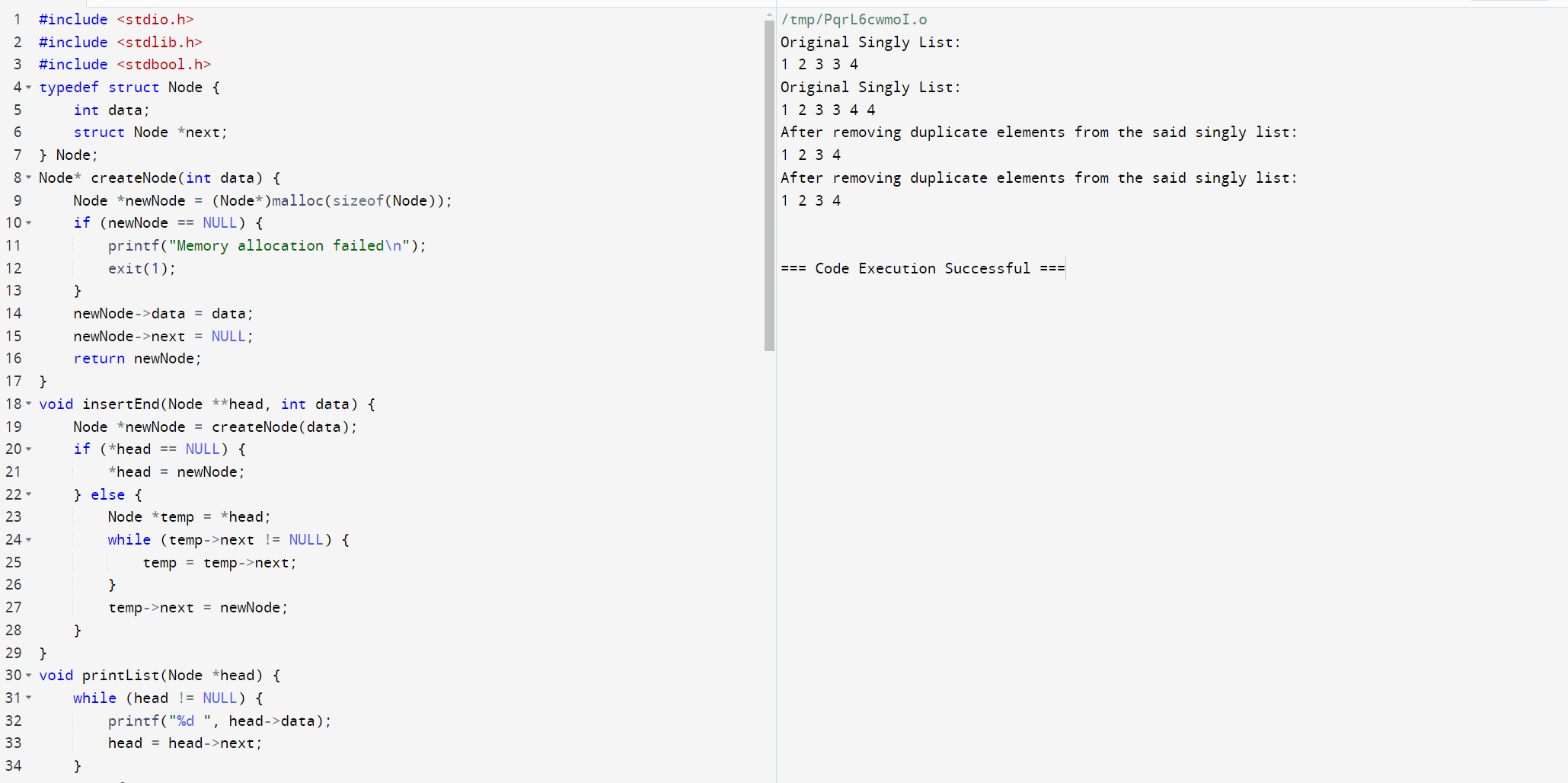
1 2 3 4

Original Singly List:

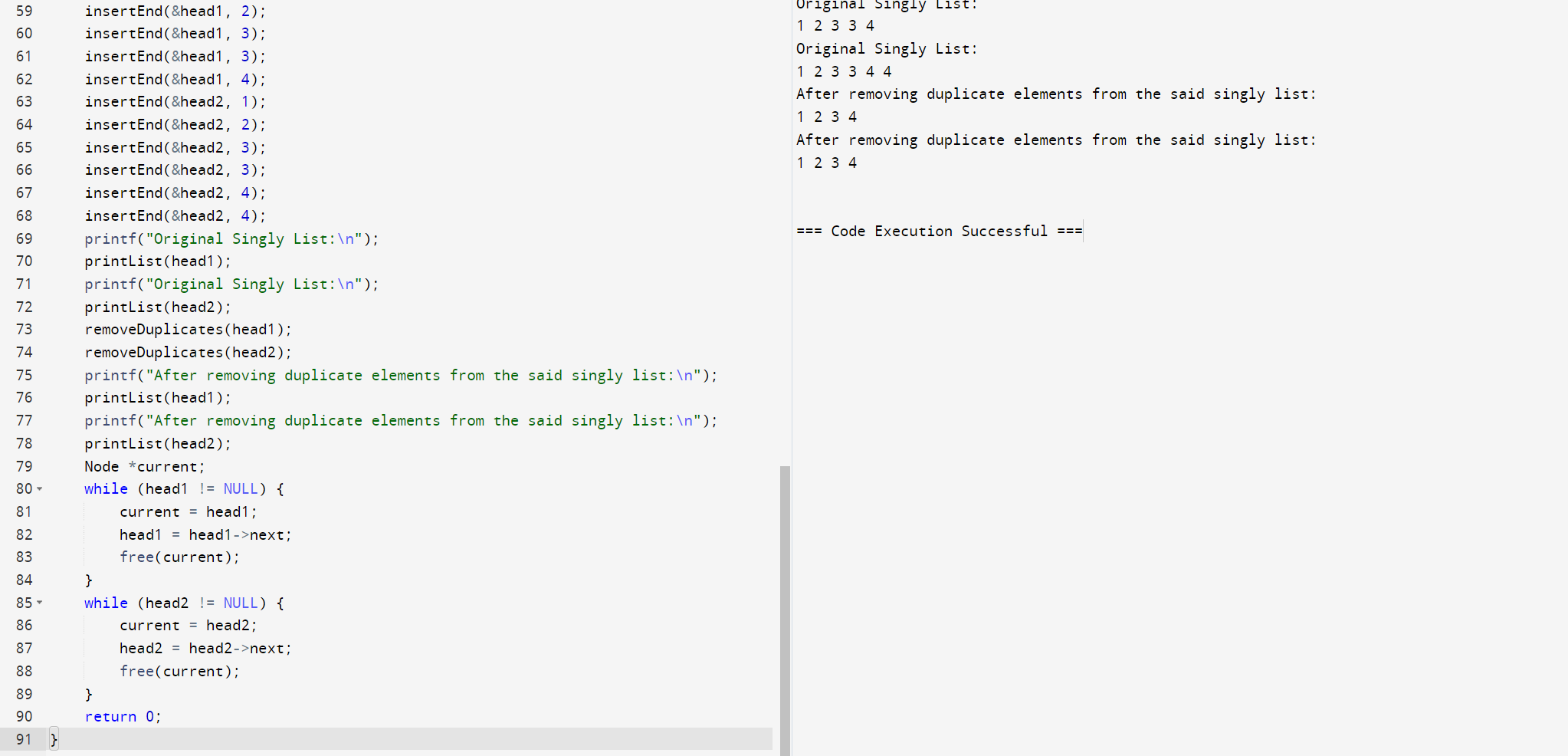
1 2 3 3 4 4

After removing duplicate elements from the said singly list:

1 2 3 4







67. Write a C program to find the intersection of two singly linked lists.

Test Data and Expected Output :

Original lists:

1 2 3 4

5 3 4

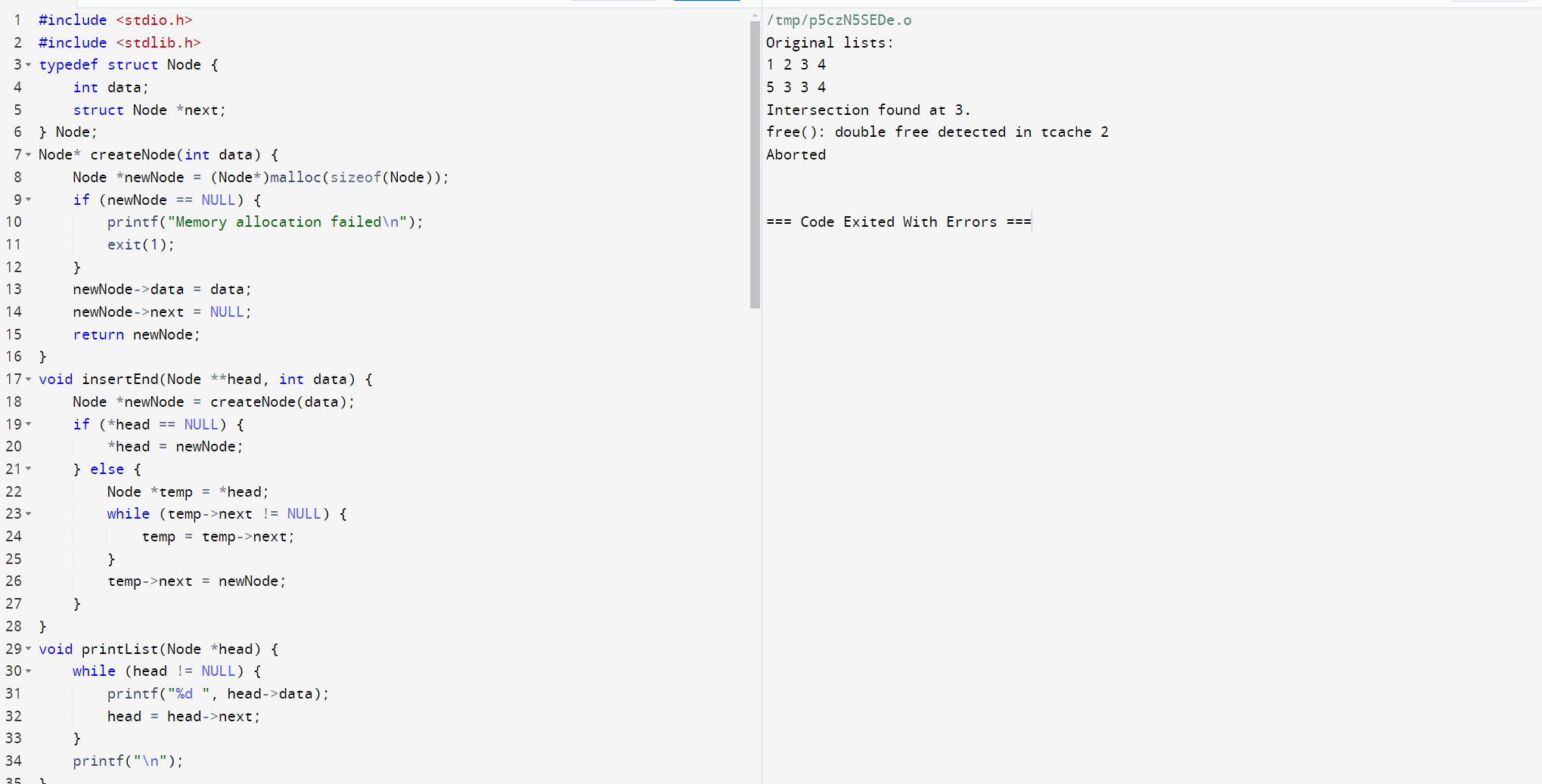
Intersection found at 3.

Original lists:

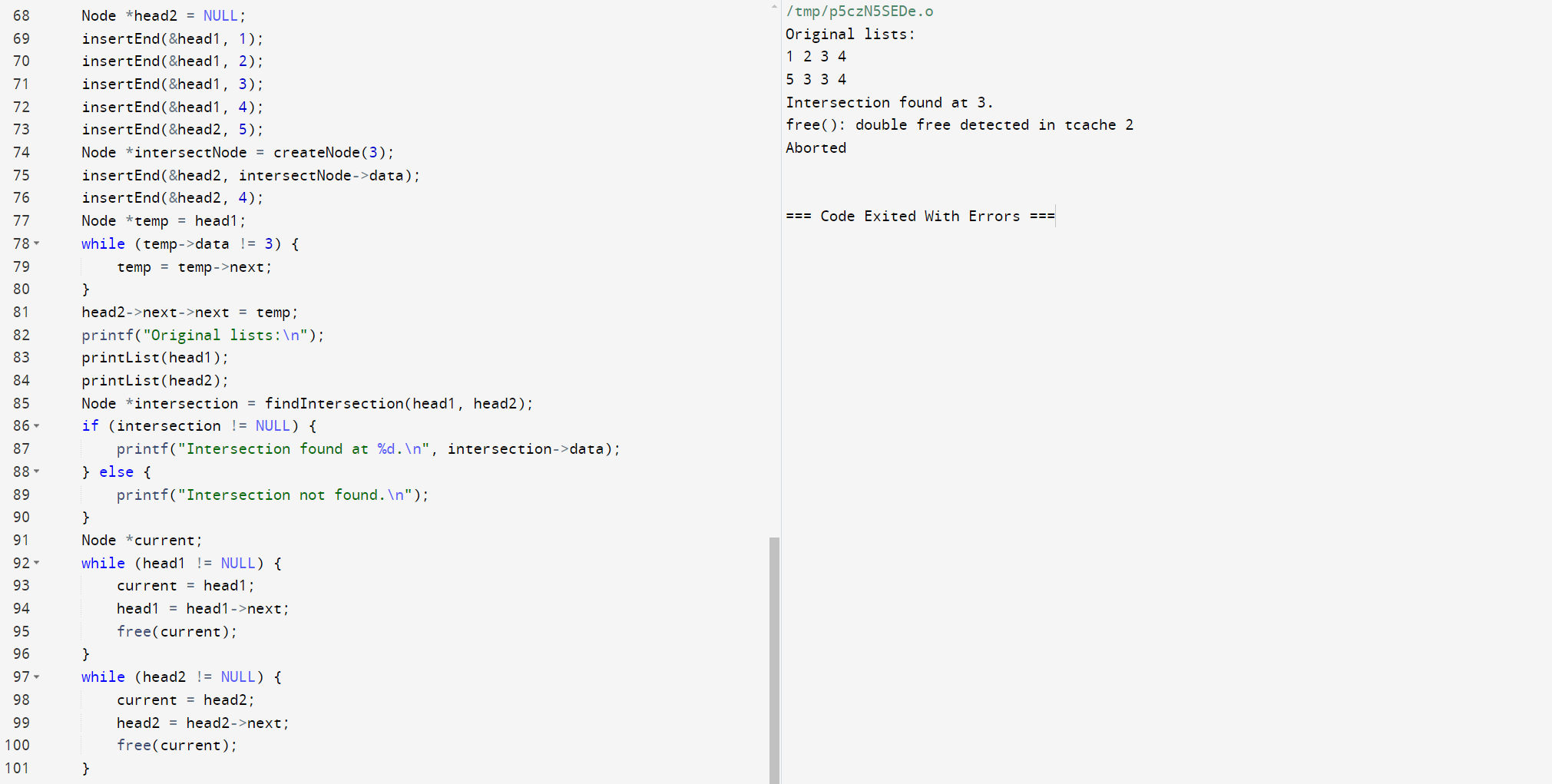
1 2 3 4

5 3 4

Intersection not found.









68. Write a C program to reverse the elements of a queue. >

Expected Output:

Queue elements are:

1 2 3 4 5

Reverse Queue, elements are:

5 4 3 2 1

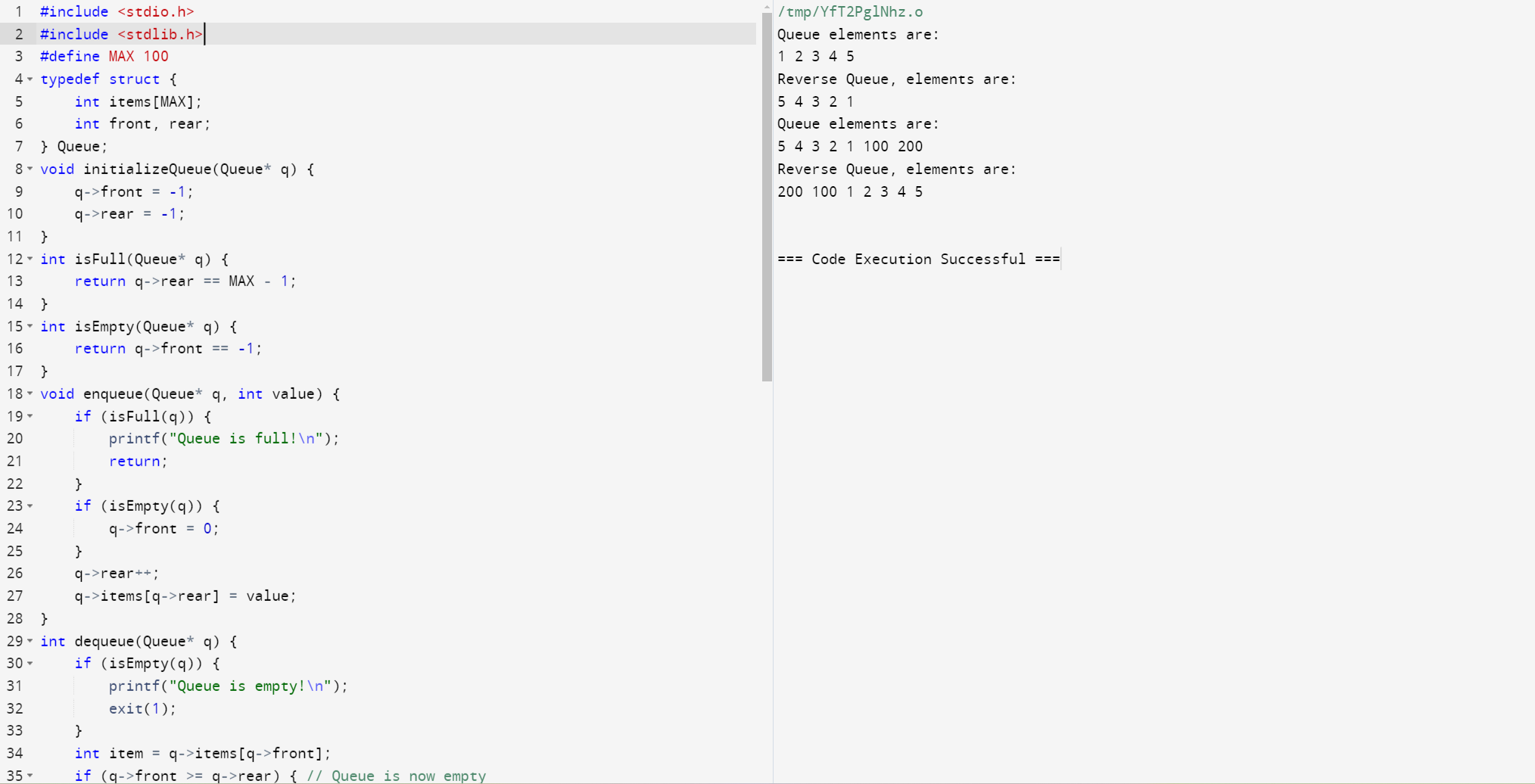
Add two elements to the said queue:

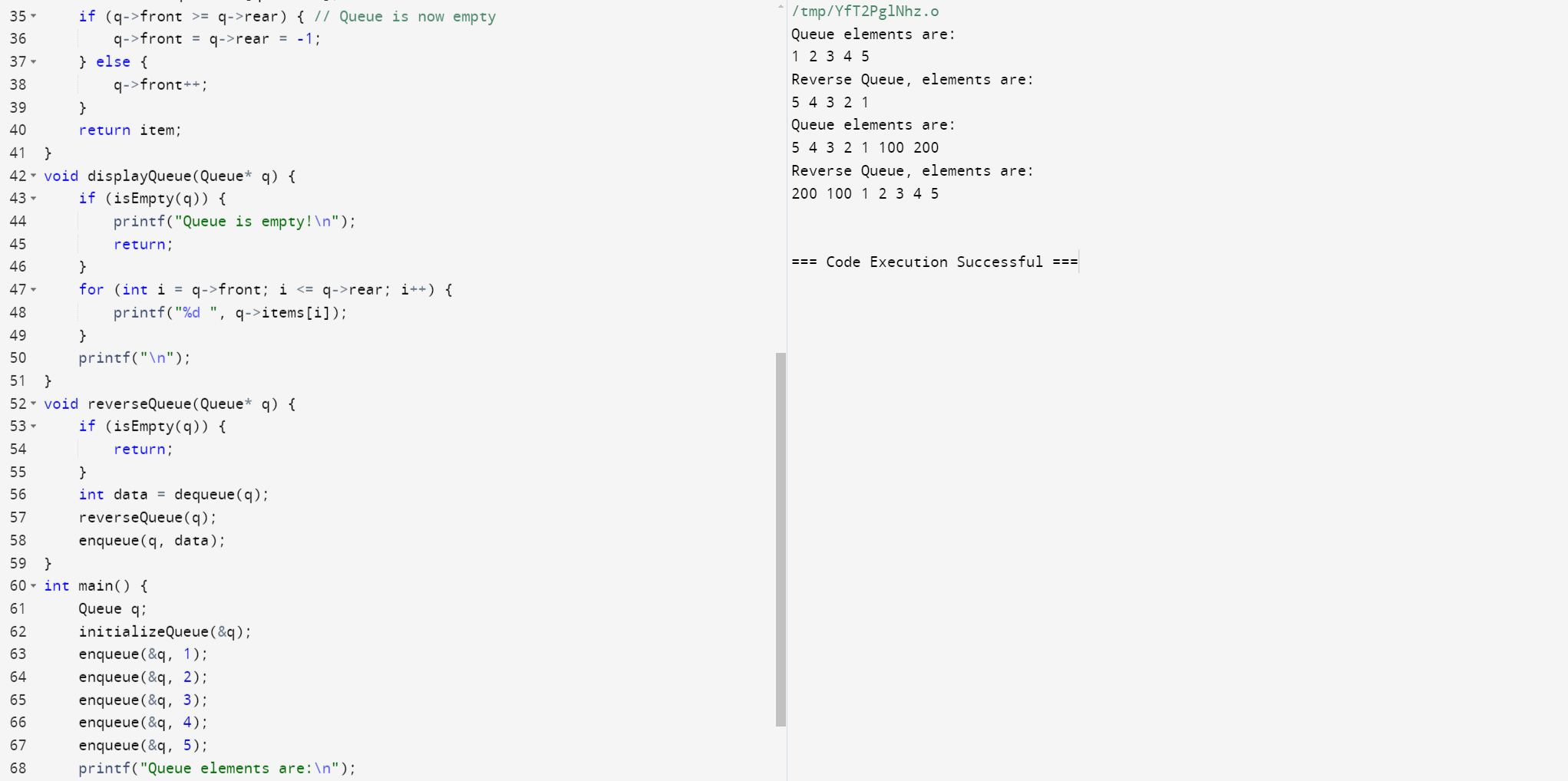
Queue elements are:

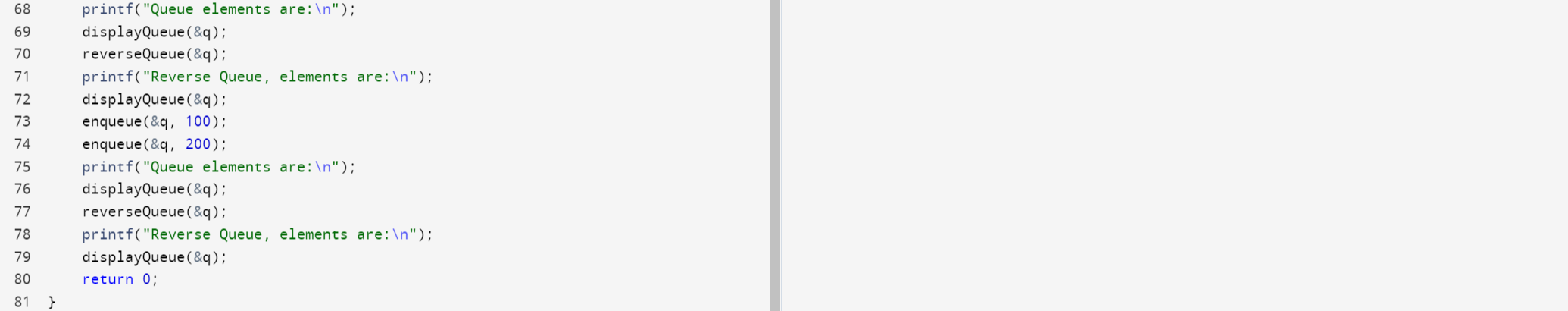
5 4 3 2 1 100 200

Reverse Queue, elements are:

200 100 1 2 3 4 5







69. Write a C program to find the maximum element in a queue. >

Expected Output:

Queue elements are: 1 2 3 4 5

Maximum value in the queue is: 5

Remove 2 elements from the said queue:

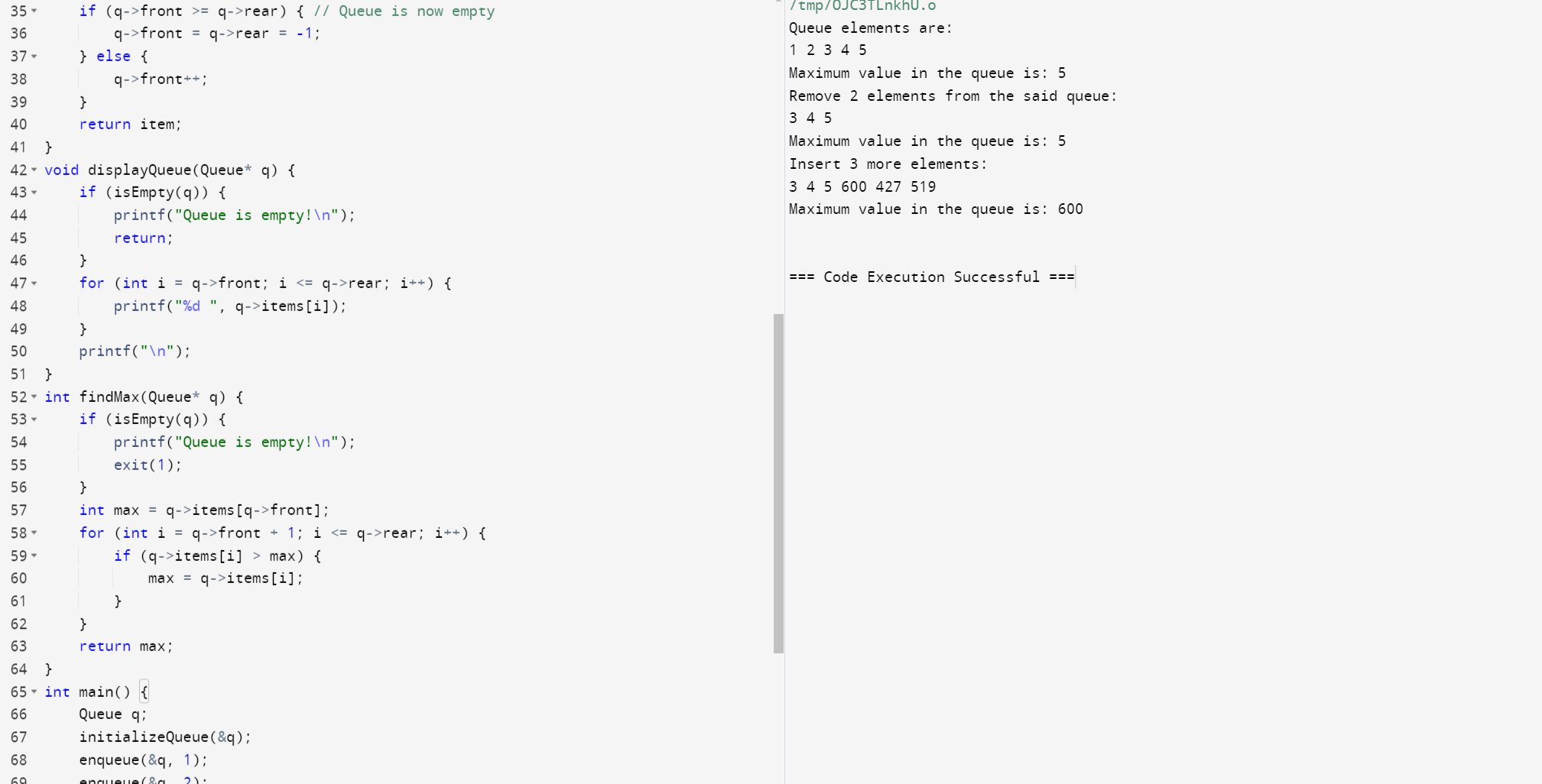
Queue elements are: 3 4 5

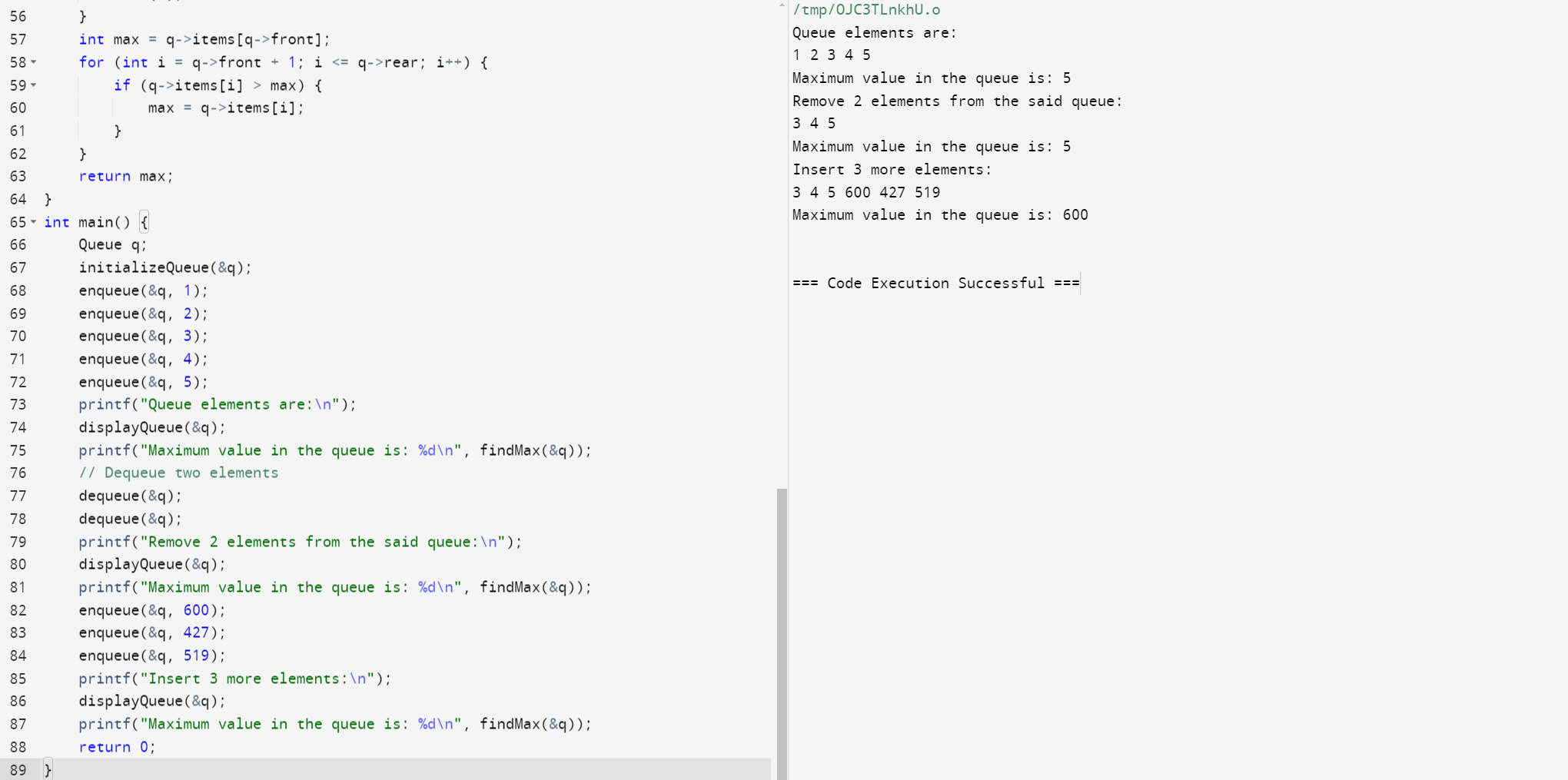
Maximum value in the queue is: 5Insert 3 more elements:

Queue elements are: 3 4 5 600 427 519

Maximum value in the queue is: 600







70. Write a C program to sort the elements of a queue in ascending order. >

Expected Output:

Input some elements into the queue:

Elements of the queue:

4 2 7 5 1

Sort the said queue:

Elements of the sorted queue in ascending order:

1 2 4 5 7

Input two more elements into the queue:

Elements of the queue:

1 2 4 5 7 -1 3

Sort the said queue:

Elements of the sorted queue in ascending order:

-1 1 2

