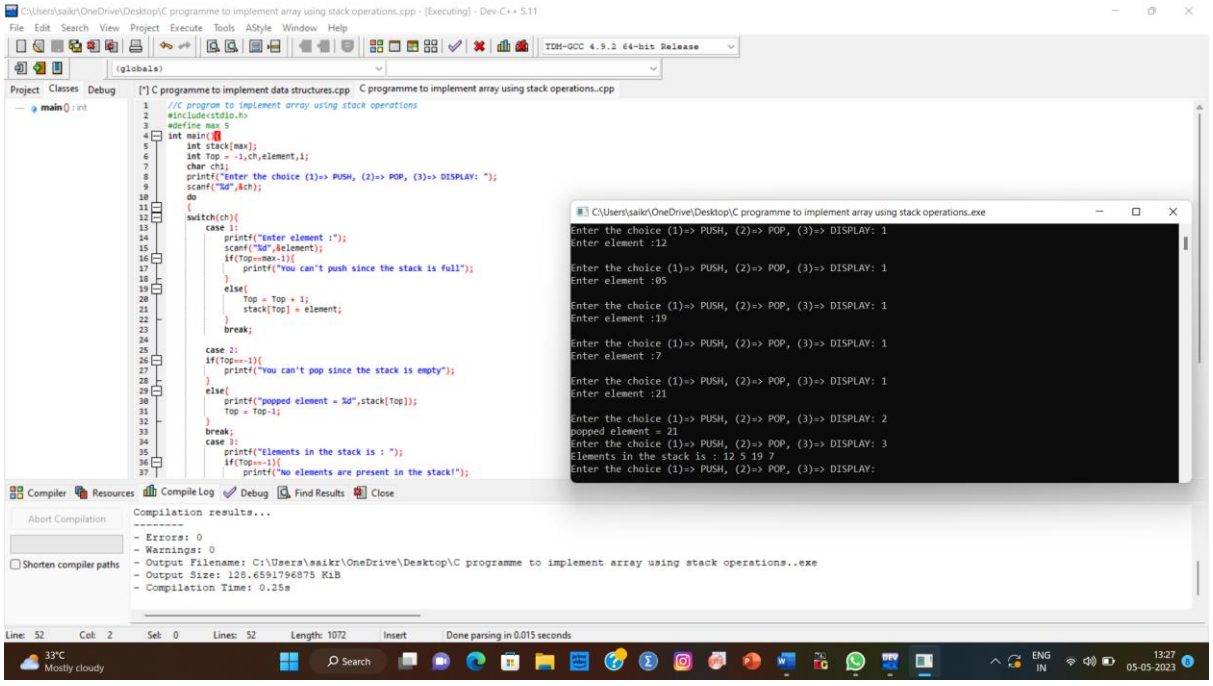
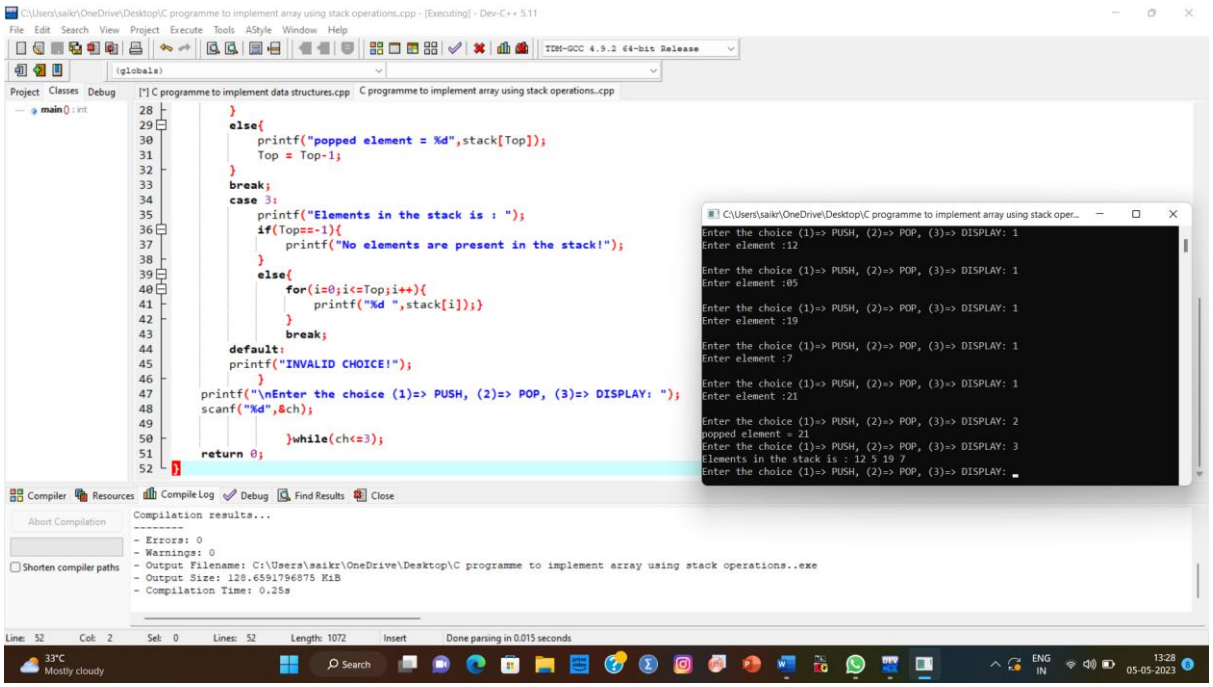
. **DATA STRUCTURES DAY 2**

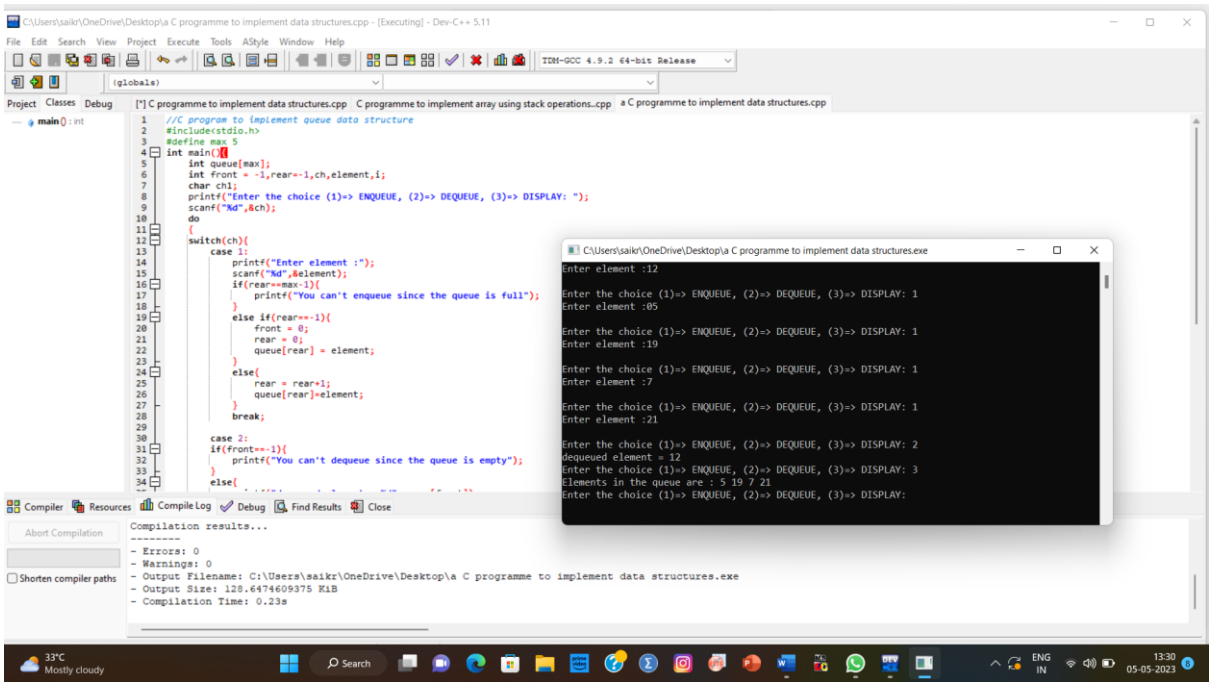
T. Sai krishna(192211870)

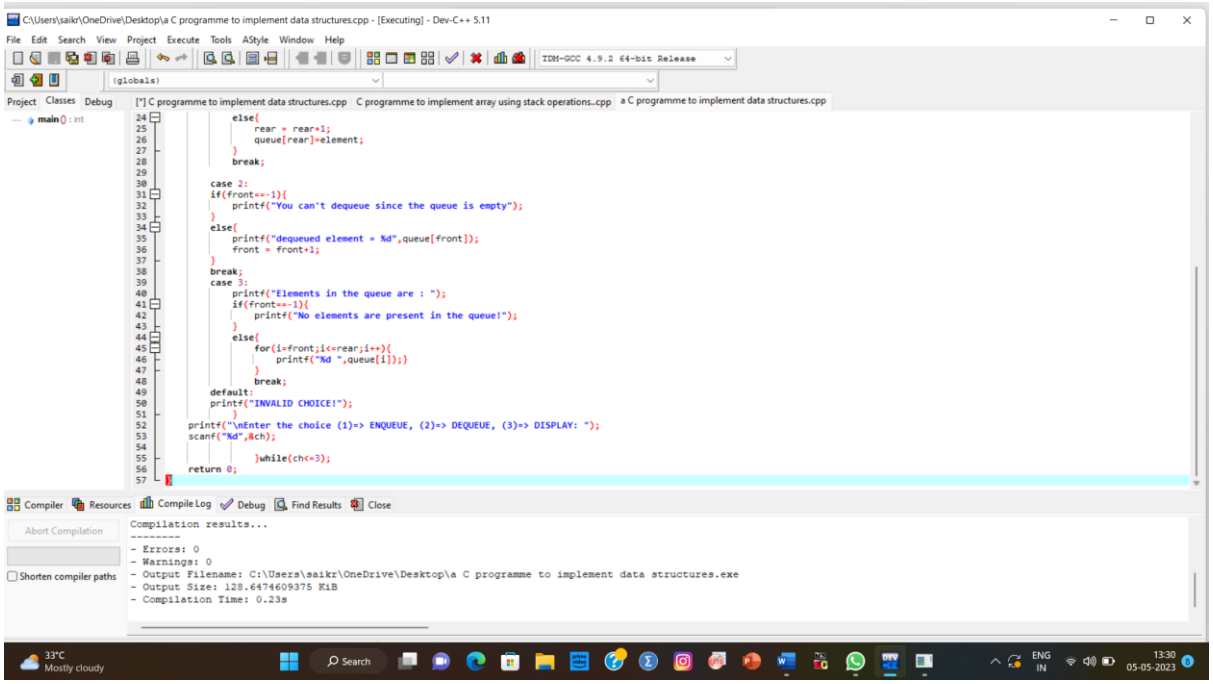
22) Write a C programme to implement array using stack operations.



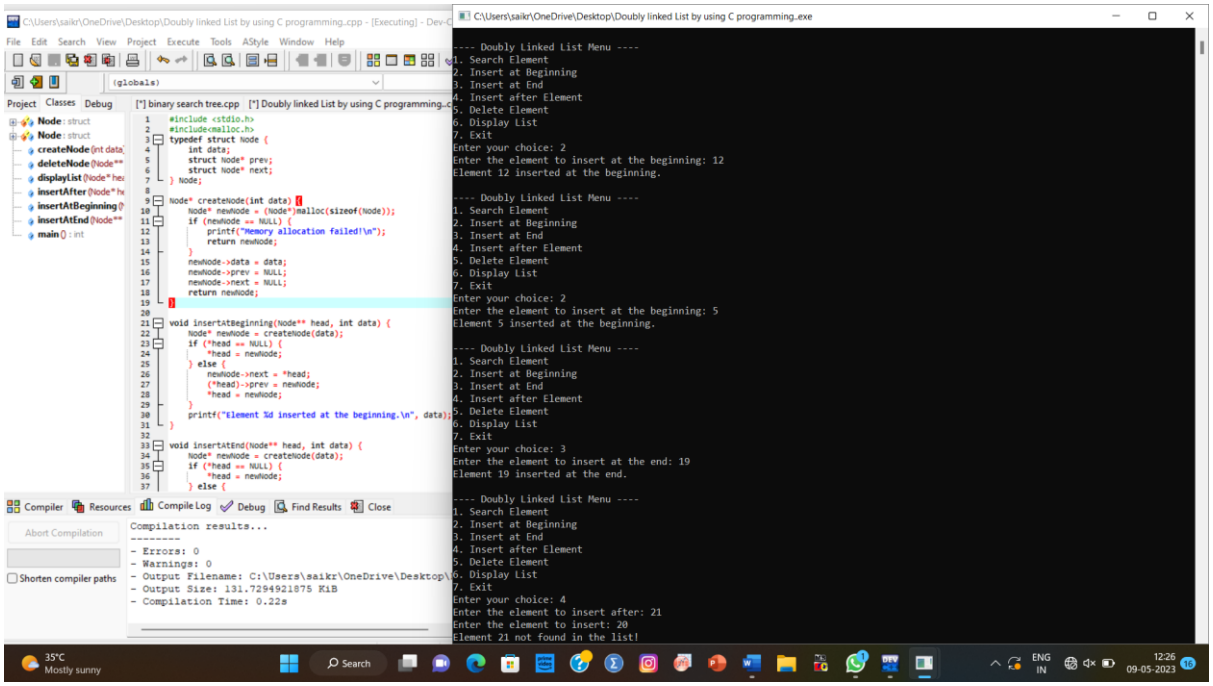


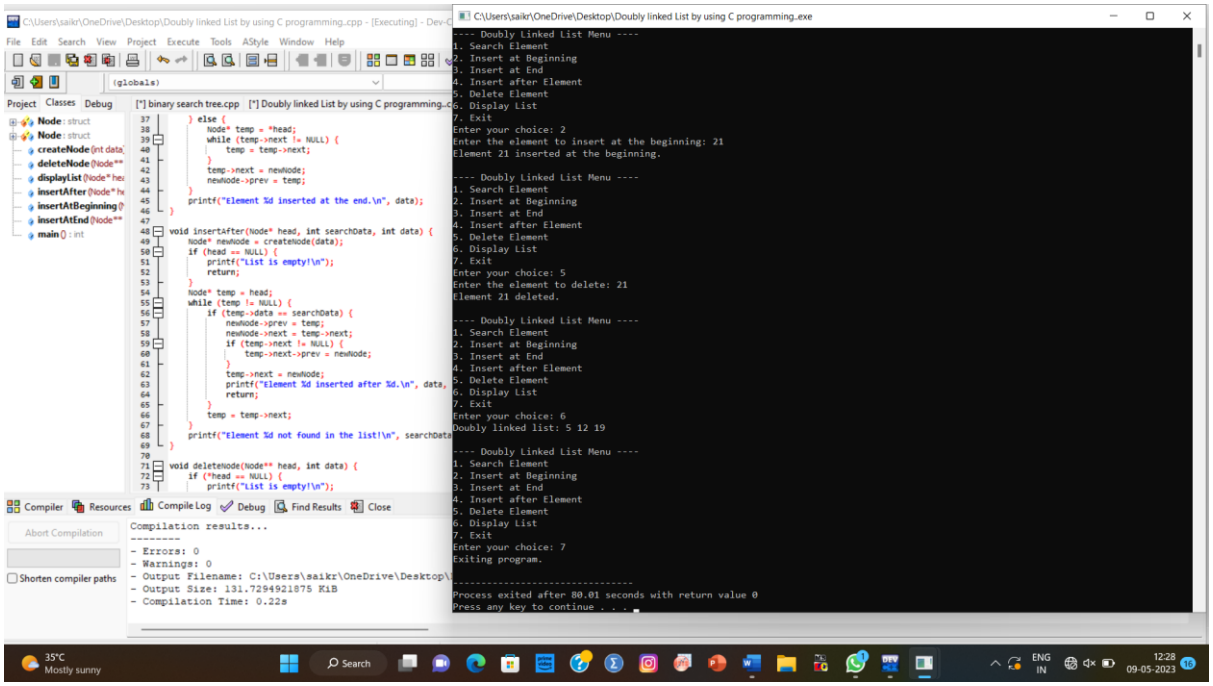
23) Write a C programme to implement data structures.

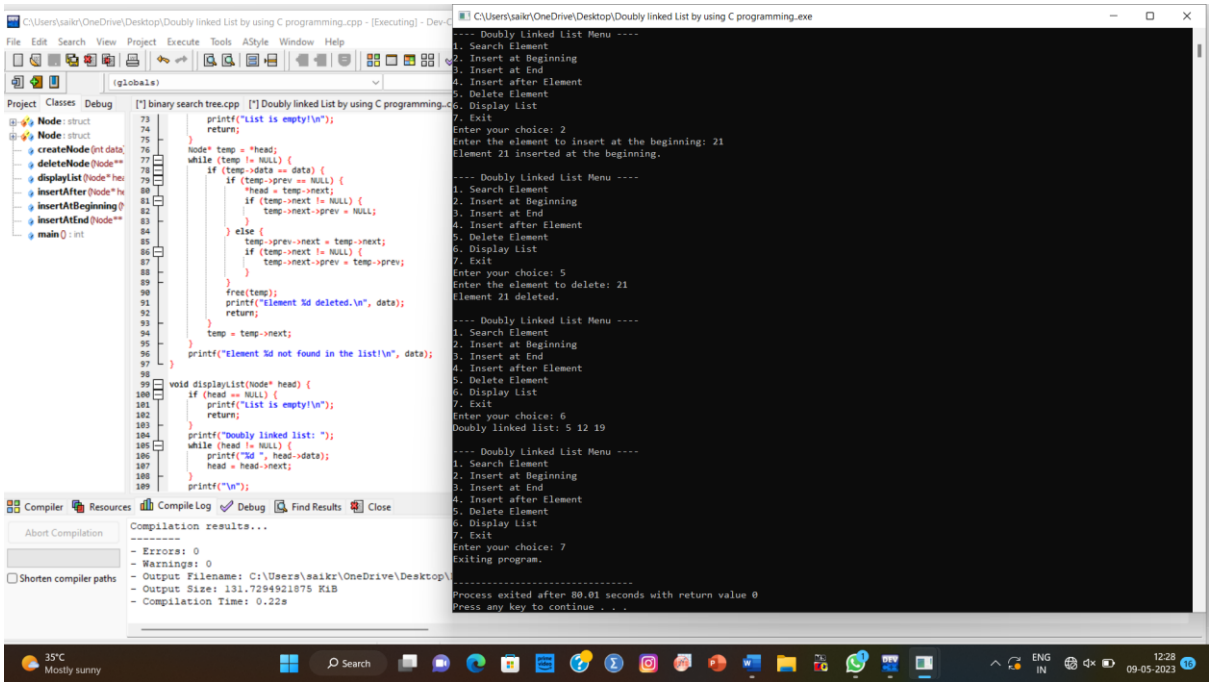


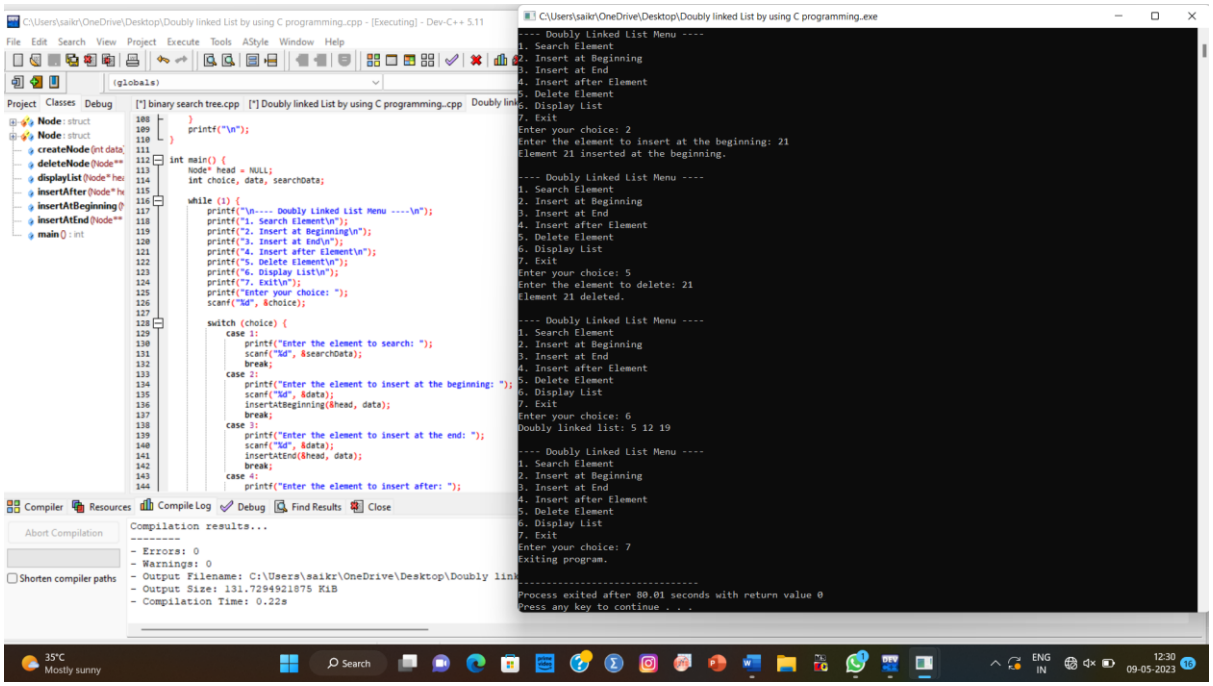


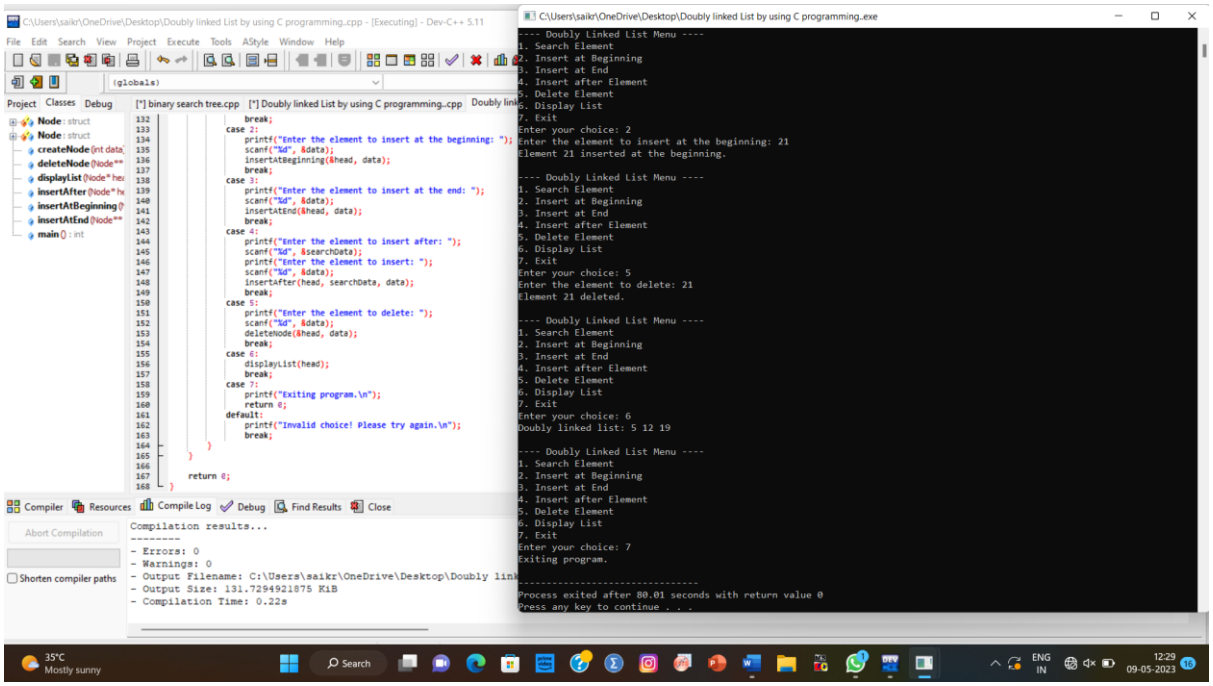
24) Doubly linked List by using C programming.



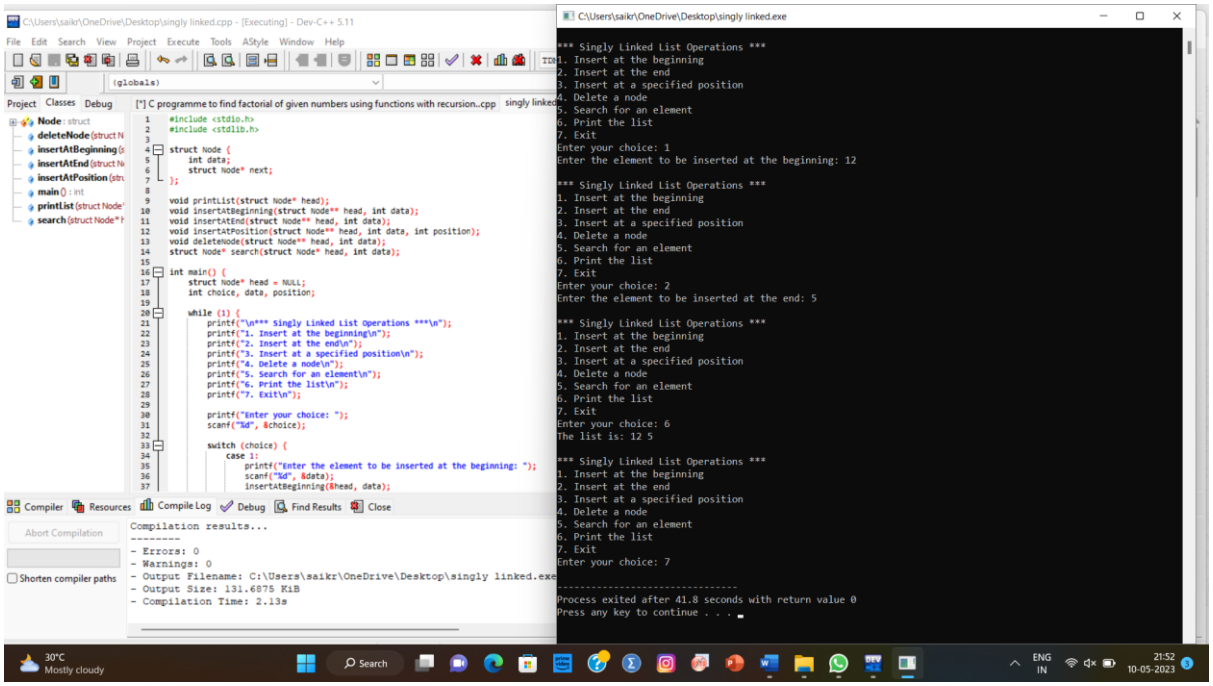


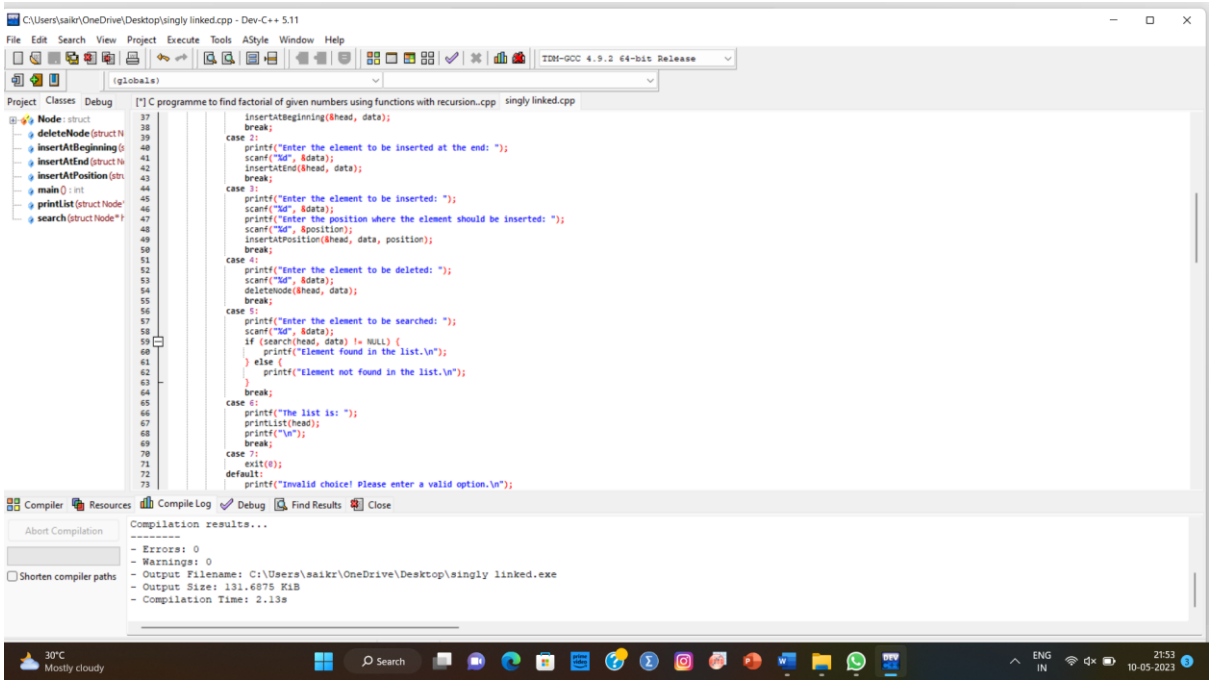

C:\Users\saikr\OneDrive\Desktop\Doubly linked List by using C programming.exe
---- Doubly Linked List Menu ----
1. Search Element
2. Insert at Beginning
3. Insert at End
4. Insert after Element
5. Delete Element
6. Display List
7. Exit
Enter your choice: 2
Enter the element to insert at the beginning: 21
Element 21 inserted at the beginning.
---- Doubly Linked List Menu ----
1. Search Element
2. Insert at Beginning
3. Insert at End
4. Insert after Element
5. Delete Element
6. Display List
7. Exit
Enter your choice: 5
Enter the element to delete: 21
Element 21 deleted.
---- Doubly Linked List Menu ----
1. Search Element
2. Insert at Beginning
3. Insert at End
4. Insert after Element
5. Delete Element
6. Display List
7. Exit
Enter your choice: 6
Doubly linked list: 5 12 19
---- Doubly Linked List Menu ----
1. Search Element
2. Insert at Beginning
3. Insert at End
4. Insert after Element
5. Delete Element
6. Display List
7. Exit
Enter your choice: 7
Exiting program.
Process exited after 80.01 seconds with return value 0
Press any key to continue . . .
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\saikr\OneDrive\Desktop\
- Output Size: 131.7294921875 KiB
- Compilation Time: 0.22s
35°C
Mostly sunny
Search
ENG
IN
12:28
09-05-2023


C:\Users\saikr\OneDrive\Desktop\Doubly linked List by using C programming.exe
---- Doubly Linked List Menu ----
1. Search Element
2. Insert at Beginning
3. Insert at End
4. Insert after Element
5. Delete Element
6. Display List
7. Exit
Enter your choice: 2
Enter the element to insert at the beginning: 21
Element 21 inserted at the beginning.
---- Doubly Linked List Menu ----
1. Search Element
2. Insert at Beginning
3. Insert at End
4. Insert after Element
5. Delete Element
6. Display List
7. Exit
Enter your choice: 5
Enter the element to delete: 21
Element 21 deleted.
---- Doubly Linked List Menu ----
1. Search Element
2. Insert at Beginning
3. Insert at End
4. Insert after Element
5. Delete Element
6. Display List
7. Exit
Enter your choice: 6
Doubly linked list: 5 12 19
---- Doubly Linked List Menu ----
1. Search Element
2. Insert at Beginning
3. Insert at End
4. Insert after Element
5. Delete Element
6. Display List
7. Exit
Enter your choice: 7
Exiting program.
Process exited after 80.01 seconds with return value 0
Press any key to continue . . .
Compilation results...
- Errors: 0
- Warnings: 0
- Output Size: 131.7294921875 KiB
- Compilation Time: 0.22s
35°C
Mostly sunny
Search
ENG
IN
12:30
09-05-2023

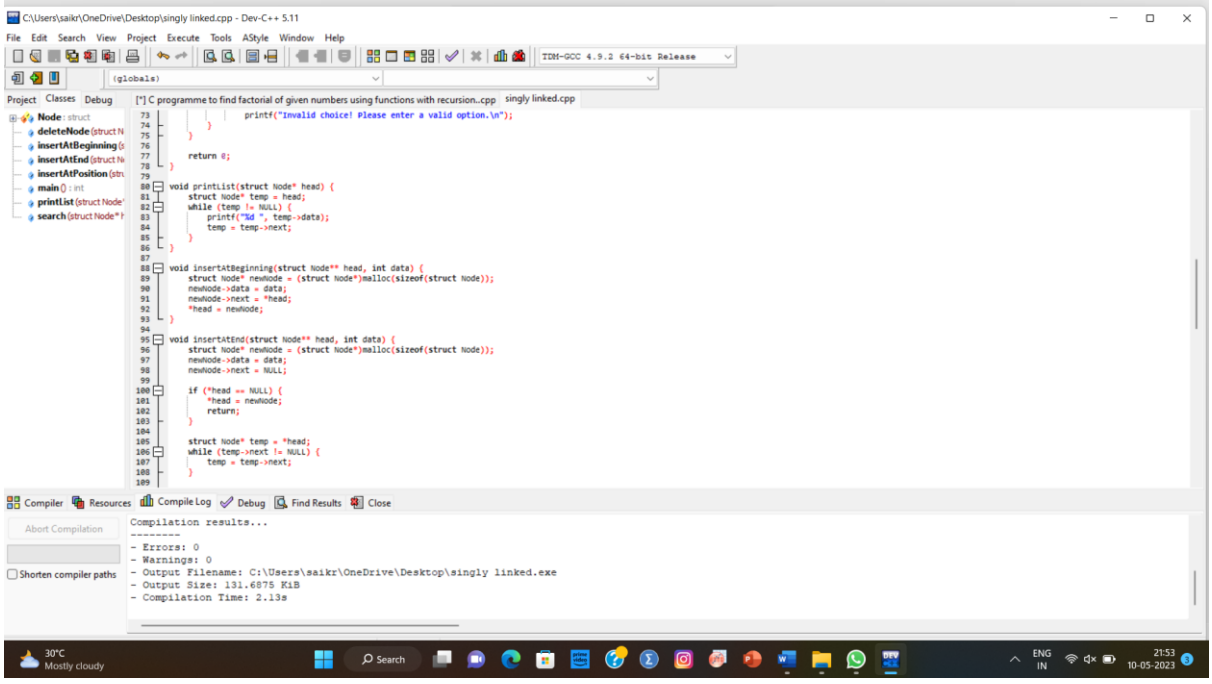


25)Singly linked List by using C programming.

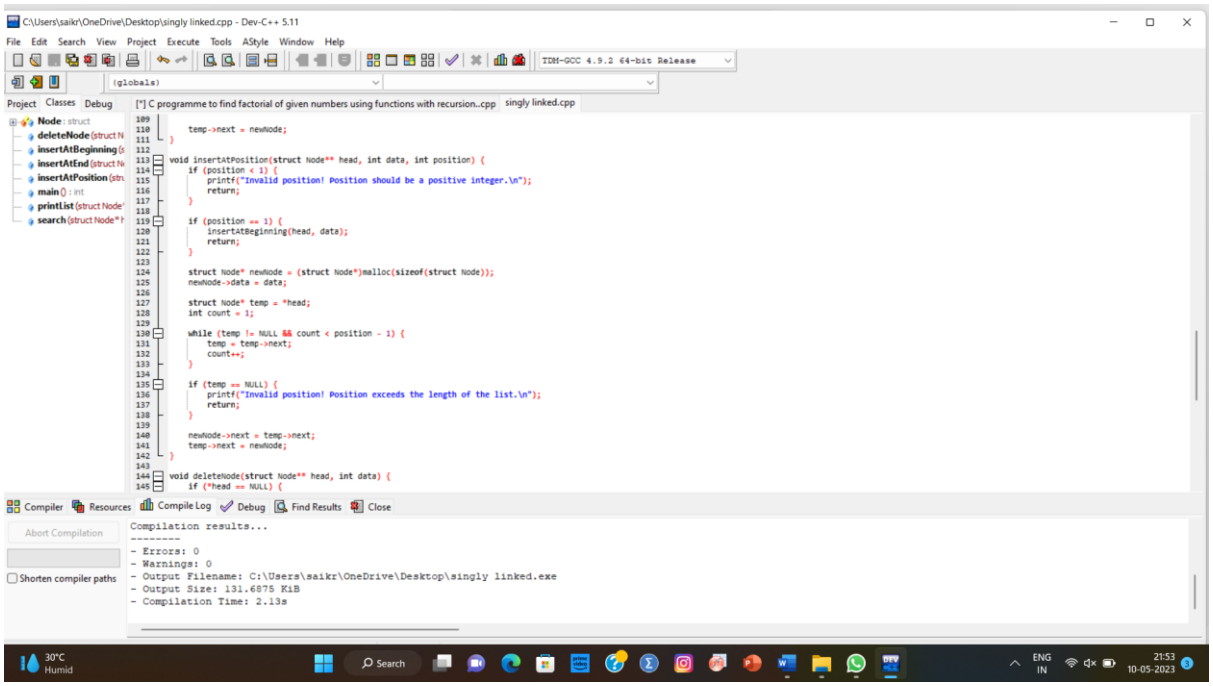


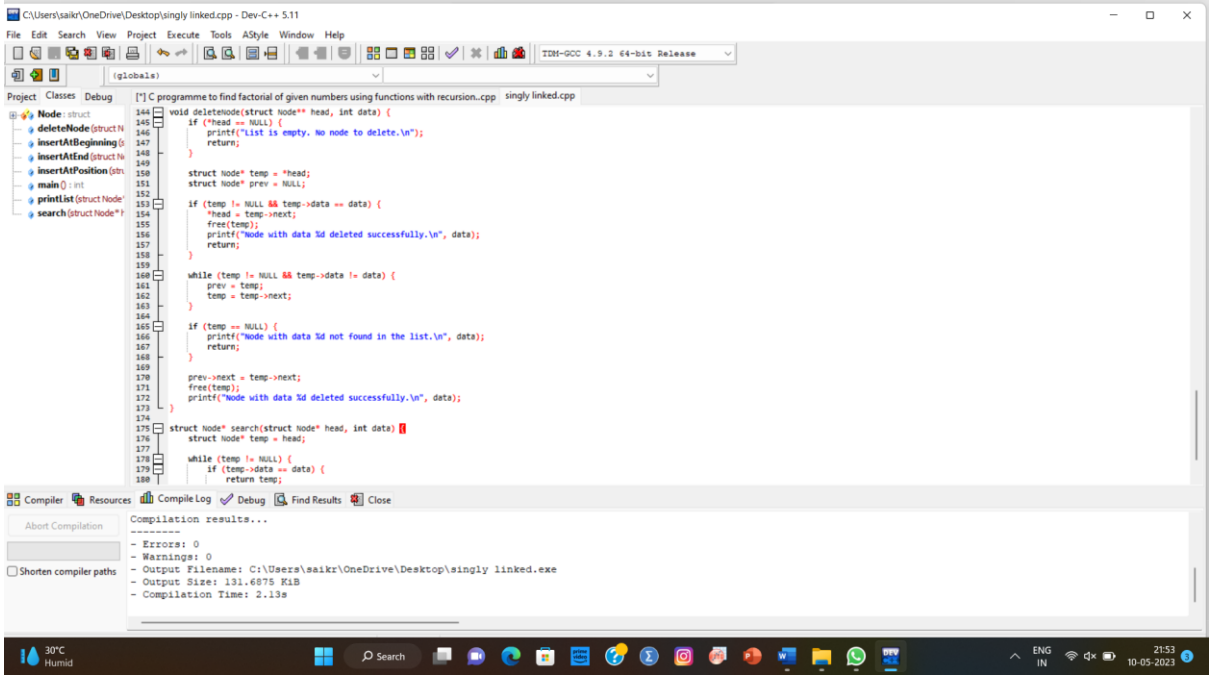

C:\Users\saikr\OneDrive\Desktop\singly linked.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
[*] C programme to find factorial of given numbers using functions with recursion..cpp
singly linked.cpp
Node : struct
deleteNode (struct N
insertAtBeginning (s
insertAtEnd (struct N
insertAtPosition (str
main () : int
printList (struct Node*
search (struct Node* h
insertatBeginning(&head, data);
break;
case 2:
printf("Enter the element to be inserted at the end: ");
scanf("%d", &data);
insertAtEnd(&head, data);
break;
case 3:
printf("Enter the element to be inserted: ");
scanf("%d", &data);
printf("Enter the position where the element should be inserted: ");
scanf("%d", &position);
insertAtPosition(&head, data, position);
break;
case 4:
printf("Enter the element to be deleted: ");
scanf("%d", &data);
deleteNode(&head, data);
break;
case 5:
printf("Enter the element to be searched: ");
scanf("%d", &data);
if (search(head, data) != NULL) {
printf("Element found in the list.\n");
} else {
printf("Element not found in the list.\n");
}
break;
case 6:
printf("The list is: ");
printList(head);
printf("\n");
break;
case 7:
exit(0);
default:
printf("Invalid choice! Please enter a valid option.\n");
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Shorten compiler paths
Compilation results...

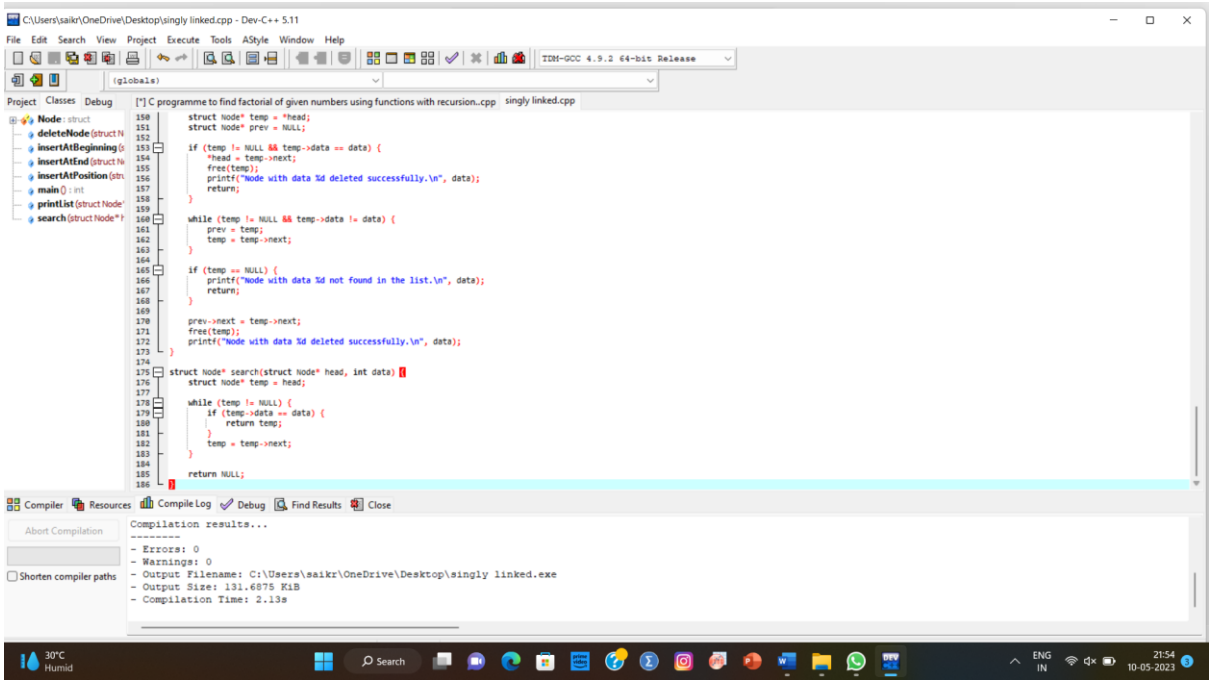
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\saikr\OneDrive\Desktop\singly linked.exe
- Output Size: 131.6875 KiB
- Compilation Time: 2.13s
30°C
Mostly cloudy
Search
ENG
IN
21:53
10-05-2023


C:\Users\saikr\OneDrive\Desktop\singly linked.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 Debug
[*] C programme to find factorial of given numbers using functions with recursion..cpp
singly linked.cpp
Node : struct
deleteNode (struct N
insertAtBeginning (s
insertAtEnd (struct N
insertAtPosition (str
main () : int
printList (struct Node*
search (struct Node* h
printf("Invalid choice! Please enter a valid option.\n");
}
}
return 0;
}
void printList(struct Node* head) {
struct Node* temp = head;
while (temp != NULL) {
printf("%d ", temp->data);
temp = temp->next;
}
}
void insertAtBeginning(struct Node** head, int data) {
struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
newNode->data = data;
newNode->next = *head;
*head = newNode;
}
void insertAtEnd(struct Node** head, int data) {
struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
newNode->data = data;
newNode->next = NULL;
if (*head == NULL) {
*head = newNode;
return;
}
struct Node* temp = *head;
while (temp->next != NULL) {
temp = temp->next;
}
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Shorten compiler paths
Compilation results...

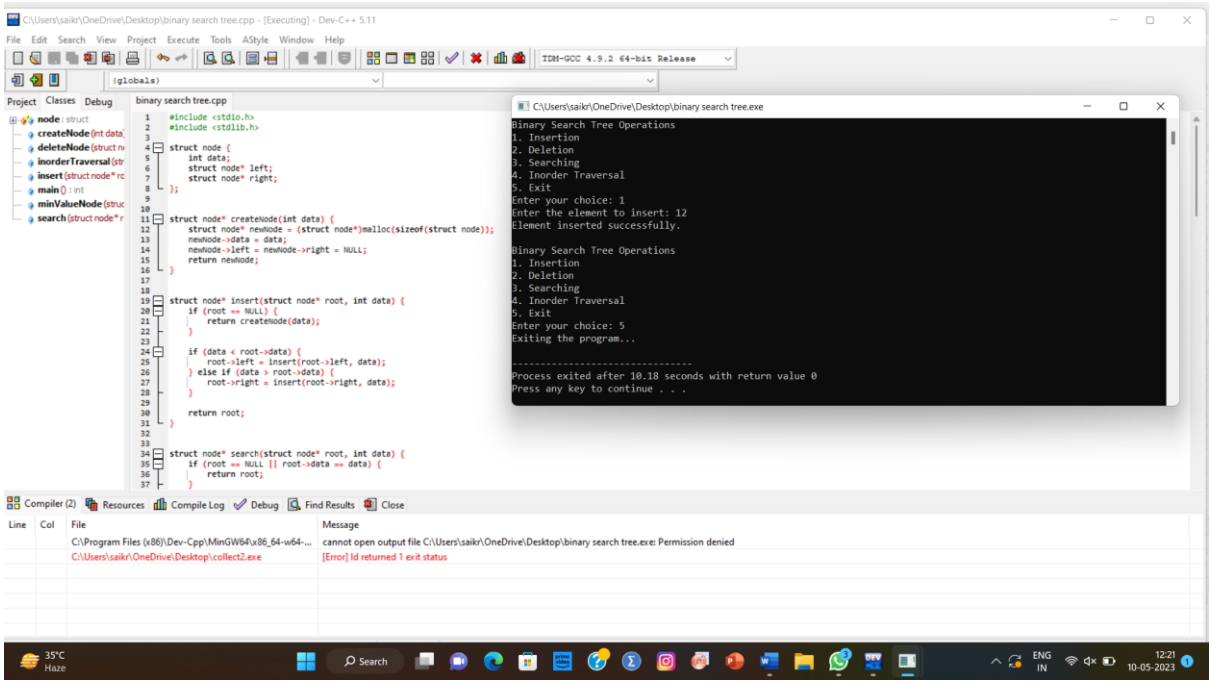
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\saikr\OneDrive\Desktop\singly linked.exe
- Output Size: 131.6875 KiB
- Compilation Time: 2.13s
30°C
Mostly cloudy
Search
ENG
IN
21:53
10-05-2023

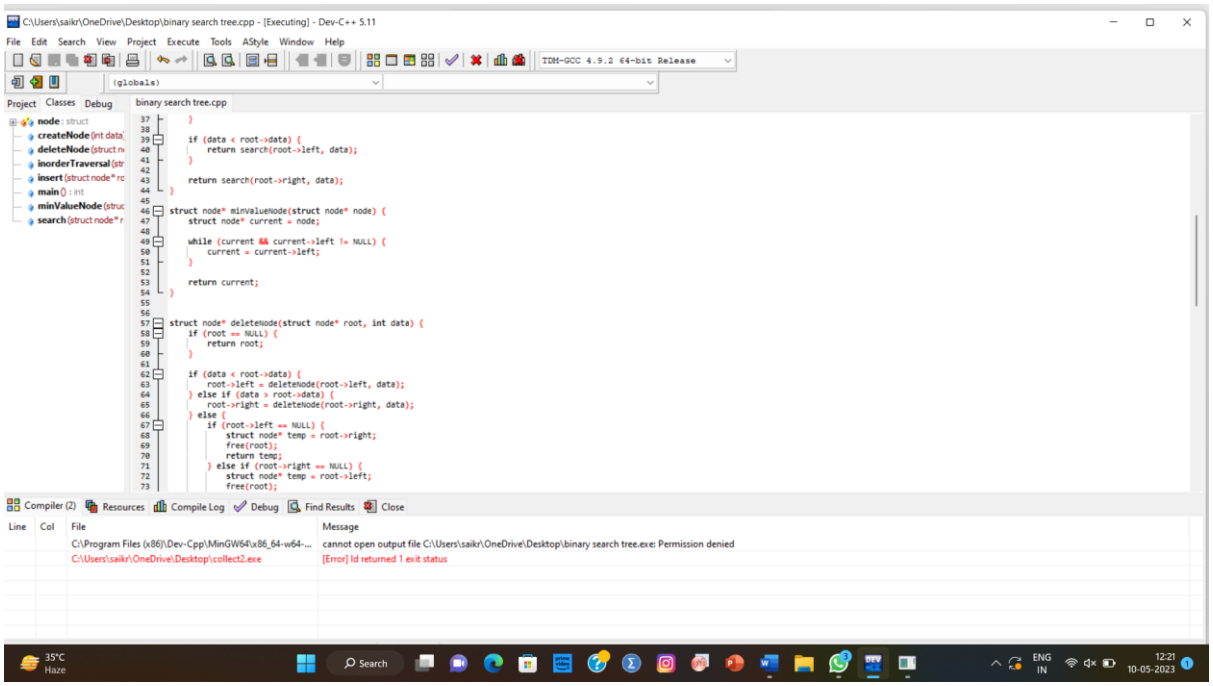

C:\Users\saikr\OneDrive\Desktop\singly linked.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
[*] C programme to find factorial of given numbers using functions with recursion..cpp
singly linked.cpp
Node : struct
deleteNode (struct N
insertAtBeginning (s
insertAtEnd (struct N
insertAtPosition (stru
main () : int
printList (struct Node*
search (struct Node*)
109 temp->next = newNode;
110 }
111 void insertAtPosition(struct Node** head, int data, int position) {
112 if (position < 1) {
113 printf("Invalid position! Position should be a positive integer.\n");
114 return;
115 }
116 if (position == 1) {
117 insertAtBeginning(head, data);
118 return;
119 }
120 struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
121 newNode->data = data;
122 struct Node* temp = *head;
123 int count = 1;
124 while (temp != NULL && count < position - 1) {
125 temp = temp->next;
126 count++;
127 }
128 if (temp == NULL) {
129 printf("Invalid position! Position exceeds the length of the list.\n");
130 return;
131 }
132 newNode->next = temp->next;
133 temp->next = newNode;
134 }
135 void deleteNode(struct Node** head, int data) {
136 if (*head == NULL) {
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Shorten compiler paths
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\saikr\OneDrive\Desktop\singly linked.exe
- Output Size: 131.6875 KiB
- Compilation Time: 2.13s
30°C Humid
Search
ENG IN
21:53 10-05-2023

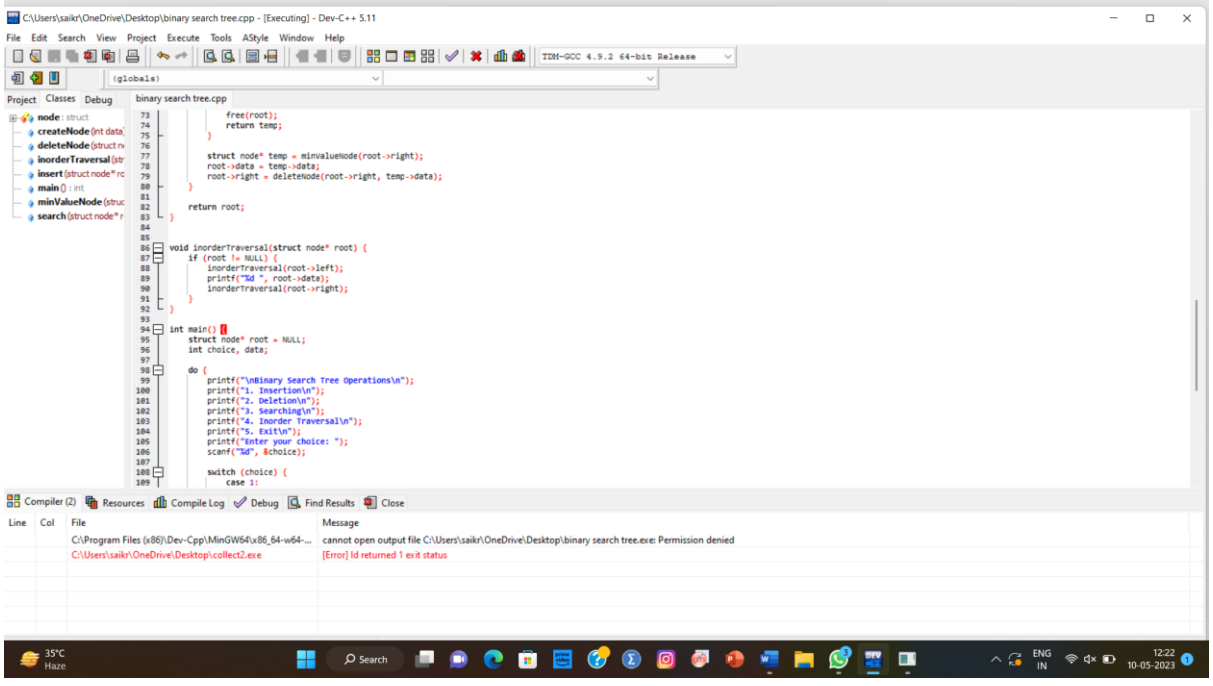

C:\Users\saikr\OneDrive\Desktop\singly linked.cpp - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
[*] C programme to find factorial of given numbers using functions with recursion..cpp
singly linked.cpp
Node : struct
deleteNode (struct N
insertAtBeginning (s
insertAtEnd (struct N
insertAtPosition (stru
main () : int
printList (struct Node*
search (struct Node*)
144 void deleteNode(struct Node** head, int data) {
145 if (*head == NULL) {
146 printf("List is empty. No node to delete.\n");
147 return;
148 }
149 struct Node* temp = *head;
150 struct Node* prev = NULL;
151 if (temp != NULL && temp->data == data) {
152 *head = temp->next;
153 free(temp);
154 printf("Node with data %d deleted successfully.\n", data);
155 return;
156 }
157 while (temp != NULL && temp->data != data) {
158 prev = temp;
159 temp = temp->next;
160 }
161 if (temp == NULL) {
162 printf("Node with data %d not found in the list.\n", data);
163 return;
164 }
165 prev->next = temp->next;
166 free(temp);
167 printf("Node with data %d deleted successfully.\n", data);
168 }
169 struct Node* search(struct Node* head, int data) {
170 struct Node* temp = head;
171 while (temp != NULL) {
172 if (temp->data == data) {
173 return temp;
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Shorten compiler paths
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\saikr\OneDrive\Desktop\singly linked.exe
- Output Size: 131.6875 KiB
- Compilation Time: 2.13s
30°C Humid
Search
ENG IN
21:53 10-05-2023

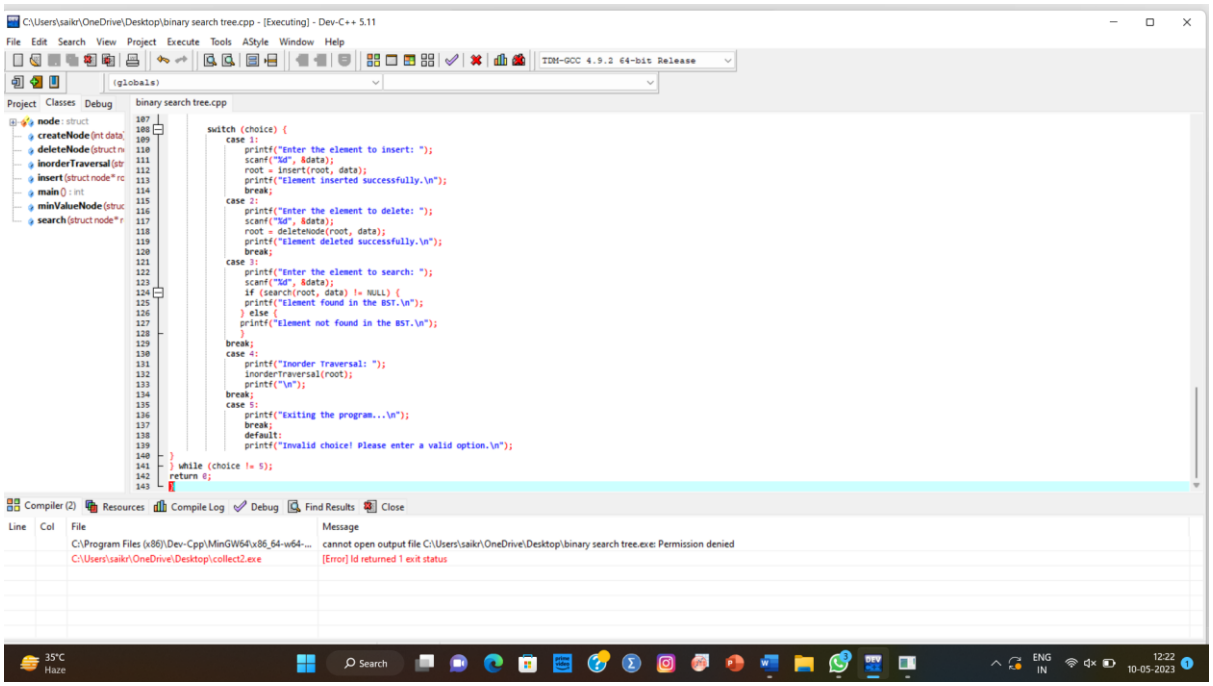


26) Binary search tree by using C programming.

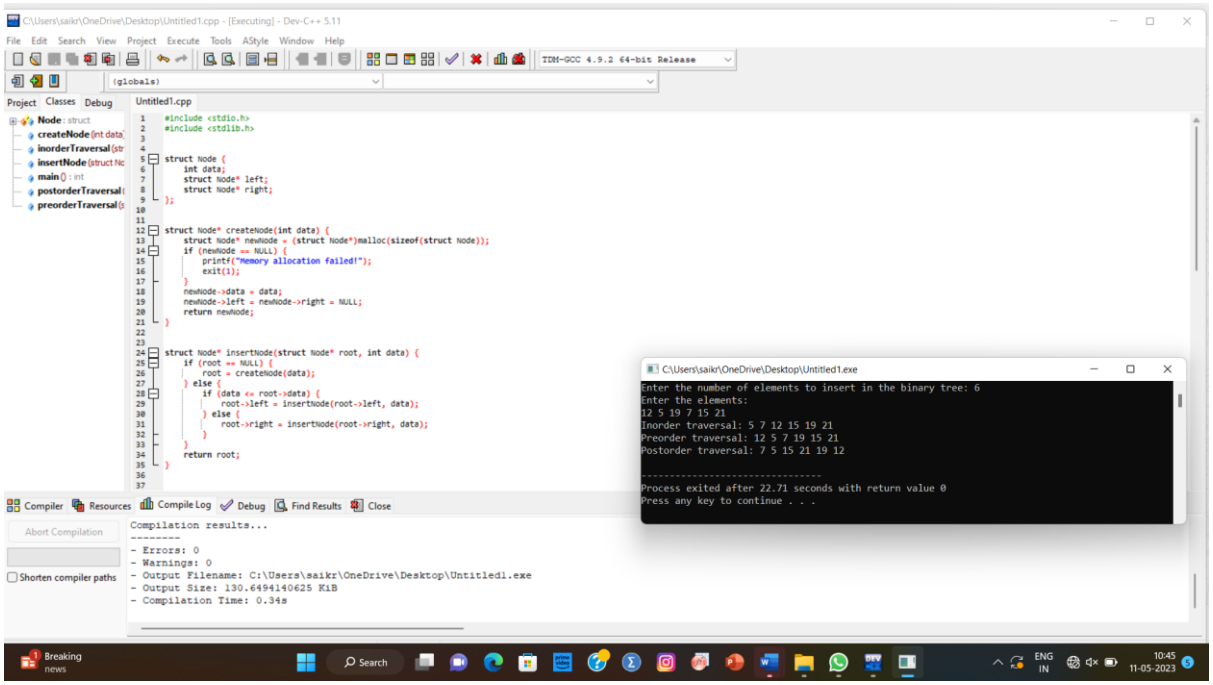


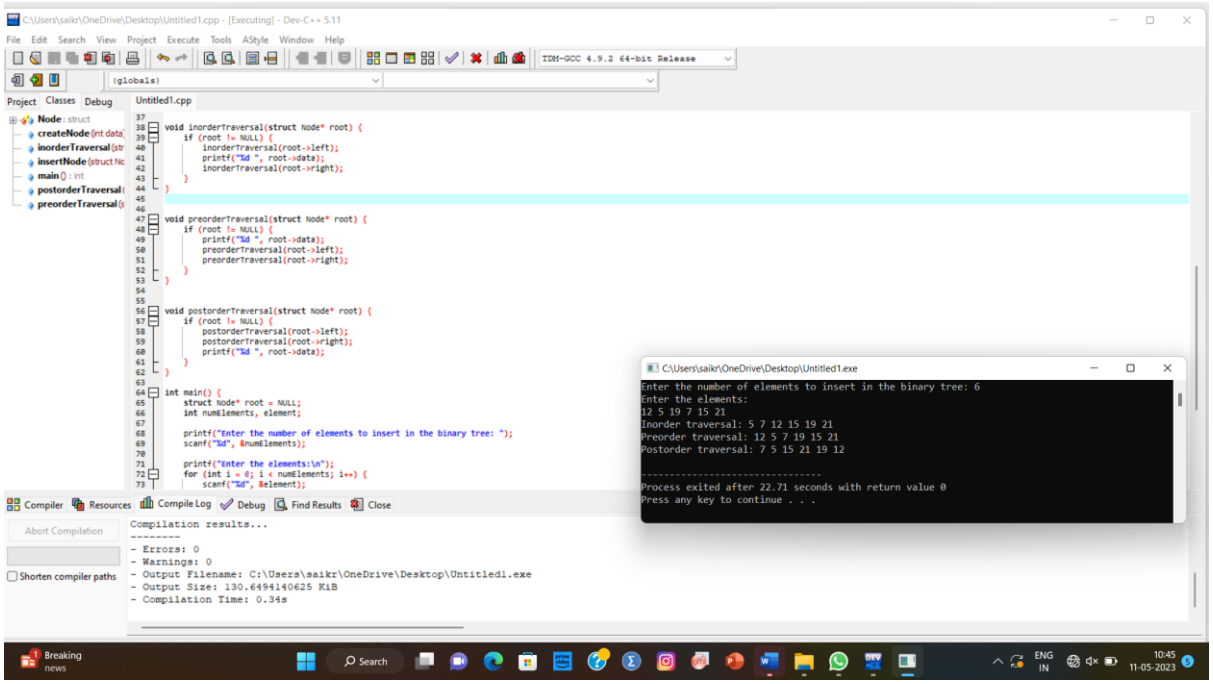

C:\Users\saikr\OneDrive\Desktop\binary search tree.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
binary search tree.cpp
node : struct
createNode(int data)
deleteNode(struct n
inorderTraversal(str
insert(struct node* rc
main() : int
minValueNode(struc
search(struct node* r
}
if (data < root->data) {
return search(root->left, data);
}
return search(root->right, data);
}
struct node* minValueNode(struct node* node) {
struct node* current = node;
while (current && current->left != NULL) {
current = current->left;
}
return current;
}
struct node* deleteNode(struct node* root, int data) {
if (root == NULL) {
return root;
}
if (data < root->data) {
root->left = deleteNode(root->left, data);
} else if (data > root->data) {
root->right = deleteNode(root->right, data);
} else {
if (root->left == NULL) {
struct node* temp = root->right;
free(root);
return temp;
} else if (root->right == NULL) {
struct node* temp = root->left;
free(root);
Compiler (2)
Resources
Compile Log
Debug
Find Results
Close
Line
Col
File
Message
C:\Program Files (x86)\Dev-Cpp\MinGW64\x86_64-w64-...
cannot open output file C:\Users\saikr\OneDrive\Desktop\binary search tree.exe: Permission denied
C:\Users\saikr\OneDrive\Desktop\collect2.exe
[Error] ld returned 1 exit status
35°C
Haze
Search
ENG
IN
12:21
10-05-2023

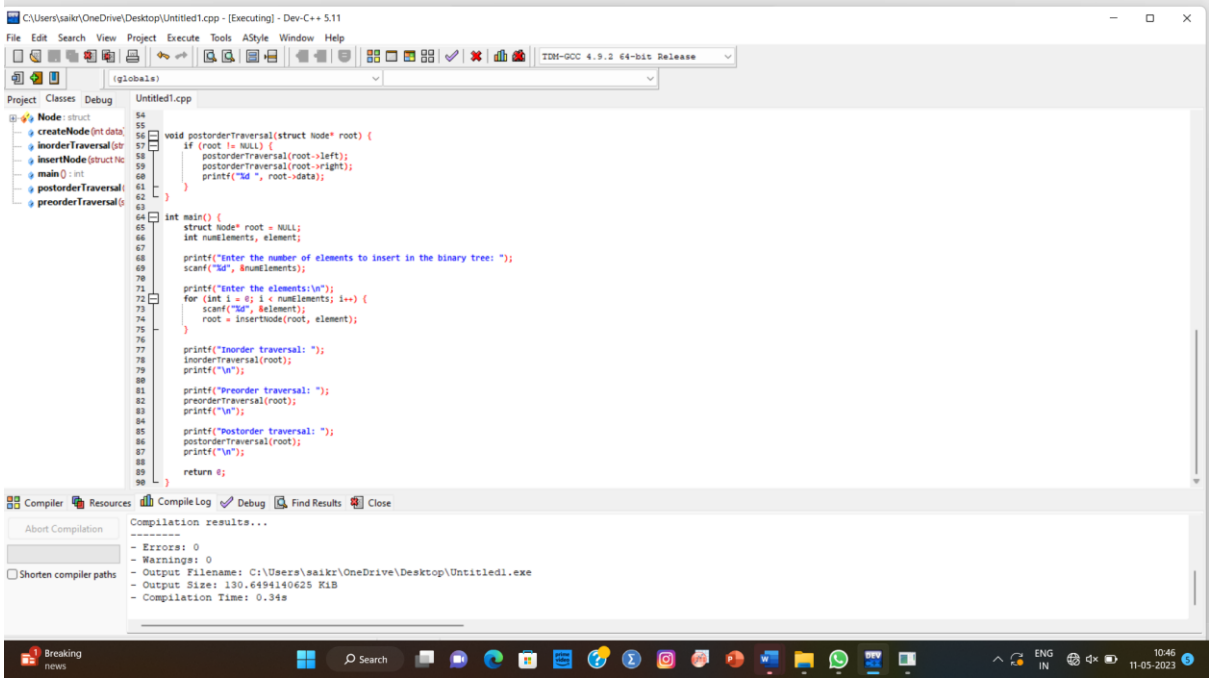

C:\Users\saikr\OneDrive\Desktop\binary search tree.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
binary search tree.cpp
node : struct
createNode(int data)
deleteNode(struct n
inorderTraversal(str
insert(struct node* rc
main() : int
minValueNode(struc
search(struct node* r
free(root);
return temp;
}
struct node* temp = minValueNode(root->right);
root->data = temp->data;
root->right = deleteNode(root->right, temp->data);
}
return root;
}
void inorderTraversal(struct node* root) {
if (root != NULL) {
inorderTraversal(root->left);
printf("%d ", root->data);
inorderTraversal(root->right);
}
}
int main() {
struct node* root = NULL;
int choice, data;
do {
printf("\nBinary Search Tree Operations\n");
printf("1. Insertion\n");
printf("2. Deletion\n");
printf("3. Searching\n");
printf("4. Inorder Traversal\n");
printf("5. Exit\n");
printf("Enter your choice: ");
scanf("%d", &choice);
switch (choice) {
case 1:
Compiler (2)
Resources
Compile Log
Debug
Find Results
Close
Line
Col
File
Message
C:\Program Files (x86)\Dev-Cpp\MinGW64\x86_64-w64-...
cannot open output file C:\Users\saikr\OneDrive\Desktop\binary search tree.exe: Permission denied
C:\Users\saikr\OneDrive\Desktop\collect2.exe
[Error] ld returned 1 exit status
35°C
Haze
Search
ENG
IN
12:22
10-05-2023



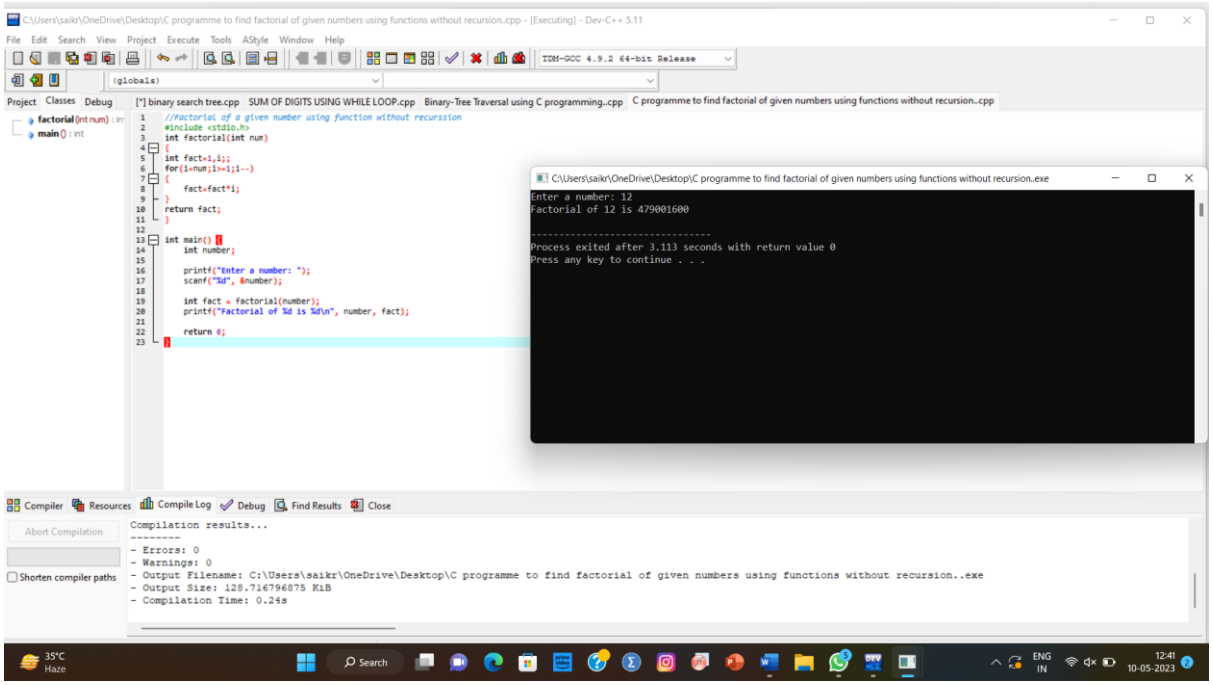
27)Binary-Tree Traversal using C programming.



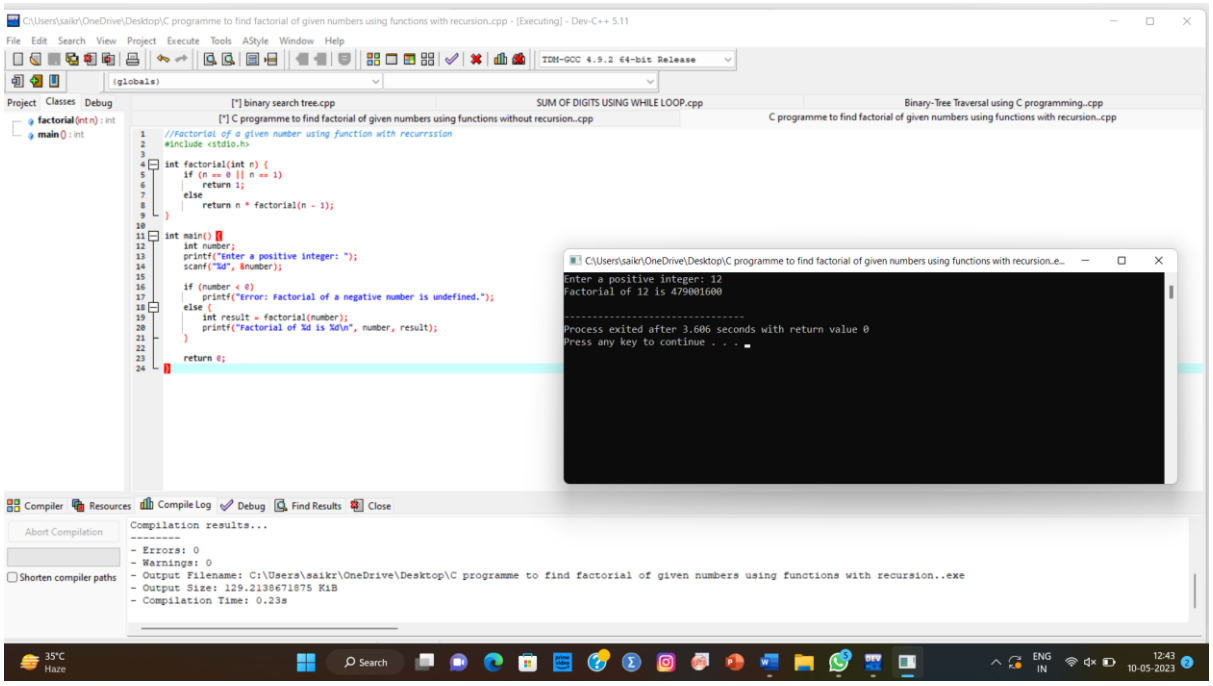

C:\Users\saikr\OneDrive\Desktop\Untitled1.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Project Classes Debug
Untitled1.cpp
Node : struct
createNode (int data)
inorderTraversal (str
insertNode (struct No
main () : int
postorderTraversal (
preorderTraversal (s
void inorderTraversal(struct Node* root) {
if (root != NULL) {
inorderTraversal(root->left);
printf("%d ", root->data);
inorderTraversal(root->right);
}
}
void preorderTraversal(struct Node* root) {
if (root != NULL) {
printf("%d ", root->data);
preorderTraversal(root->left);
preorderTraversal(root->right);
}
}
void postorderTraversal(struct Node* root) {
if (root != NULL) {
postorderTraversal(root->left);
postorderTraversal(root->right);
printf("%d ", root->data);
}
}
int main() {
struct Node* root = NULL;
int numElements, element;
printf("Enter the number of elements to insert in the binary tree: ");
scanf("%d", &numElements);
printf("Enter the elements:\n");
for (int i = 0; i < numElements; i++) {
scanf("%d", &element);
C:\Users\saikr\OneDrive\Desktop\Untitled1.exe
Enter the number of elements to insert in the binary tree: 6
Enter the elements:
12 5 19 7 15 21
Inorder traversal: 5 7 12 15 19 21
Preorder traversal: 12 5 7 19 15 21
Postorder traversal: 7 5 15 21 19 12
Process exited after 22.71 seconds with return value 0
Press any key to continue . . .
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Shorten compiler paths
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\saikr\OneDrive\Desktop\Untitled1.exe
- Output Size: 130.64941406250 KiB
- Compilation Time: 0.34s
Breaking news
Search
ENG IN
10:45 11-05-2023


C:\Users\saikr\OneDrive\Desktop\Untitled1.cpp - [Executing] - 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 Debug
Untitled1.cpp
Node : struct
createNode (int data)
inorderTraversal (str
insertNode (struct No
main () : int
postorderTraversal (
preorderTraversal (s
void postorderTraversal(struct Node* root) {
if (root != NULL) {
postorderTraversal(root->left);
postorderTraversal(root->right);
printf("%d ", root->data);
}
}
int main() {
struct Node* root = NULL;
int numElements, element;
printf("Enter the number of elements to insert in the binary tree: ");
scanf("%d", &numElements);
printf("Enter the elements:\n");
for (int i = 0; i < numElements; i++) {
scanf("%d", &element);
root = insertNode(root, element);
}
printf("Inorder traversal: ");
inorderTraversal(root);
printf("\n");
printf("Preorder traversal: ");
preorderTraversal(root);
printf("\n");
printf("Postorder traversal: ");
postorderTraversal(root);
printf("\n");
return 0;
}
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Shorten compiler paths
Compilation results...
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\saikr\OneDrive\Desktop\Untitled1.exe
- Output Size: 130.64941406250 KiB
- Compilation Time: 0.34s
Breaking news
Search
ENG IN
10:46 11-05-2023

28)Write C programme to find factorial of given numbers using functions without recursion.



29) Write C programme to find factorial of given numbers using functions with recursion.



30) Write a C programme to generate Fibonacci using function with recursion