



Green University of Bangladesh
Department of Computer Science and Engineering(CSE)
Faculty of Sciences and Engineering
Semester: (Summer, Year:2022), B.Sc. in CSE (Day)

LAB REPORT NO: 05
Course Title: Data Structure Lab
Course Code: CSE 106 Section: DB

Lab Experiment Name: Linked List

Student Details

Name		ID
1.	Shariful Islam Emon	213902056

Lab Date : 17/08/2022

Submission Date : 23/08/2022

Course Teacher's Name : Farhana Akter Sunny

[For Teachers use only: **Don't Write Anything inside this box**]

<u>Lab Report Status</u>	
Marks:	Signature:.....
Comments:.....	Date:.....

1. TITLE OF THE LAB EXPERIMENT

" Linked List"

2. IMPLEMENTATION

Answer to the problem no: 1

Problem Statement: Implement a BST and traverse the tree using Pre-order, in-order and post-order (by taking user choice) Traversal using linked list

Code:

```
#include <stdio.h>
#include <stdlib.h>
//Shariful islam emon 213902056
struct node {
    int value;
    struct node* left;
    struct node* right;
};

void InOrder(struct node* root) {
    if (root == NULL) return;
    InOrder(root->left);
    printf("%d ", root->value);
    InOrder(root->right);
}
```

```

void PreOrder(struct node* root) {
    if (root == NULL) return;
    printf("%d ", root->value);
    PreOrder(root->left);
    PreOrder(root->right);
}

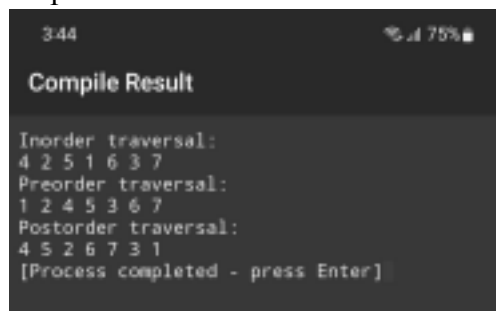
void PostOrder(struct node* root) {
    if (root == NULL) return;
    PostOrder(root->left);
    PostOrder(root->right);
    printf("%d ", root->value);
}

struct node* createNode(int value) {
    struct node* newNode = malloc(sizeof(struct node));
    newNode->value = value;
    newNode->left = NULL;
    newNode->right = NULL;
    return newNode;
}

int main() {
    struct node* root = createNode(1);
    root->left = createNode(2);
    root->right = createNode(3);
    root->left->left = createNode(4);
    root->left->right = createNode(5);
    root->right->left = createNode(6);
    root->right->right = createNode(7);
    printf("Inorder traversal:\t \n");
    InOrder(root);
    printf("\nPreorder traversal:\t \n");
    PreOrder(root);
    printf("\nPostorder traversal:\t \n");
    PostOrder(root);
}

```

Output:



```

344 75%
Compile Result
Inorder traversal:
4 2 5 1 6 3 7
Preorder traversal:
1 2 4 5 3 6 7
Postorder traversal:
4 5 2 6 7 3 1
[Process completed - press Enter]

```