CSE 2003

DATA STRUCTURES AND ALGORITHMS

Lab Assessment – 5

L19+L20 | SJT317

FALL SEMESTER 2020-21

by

SHARADINDU ADHIKARI

19BCE2105

Question 1

Problem:

WAP to implement doubly linked list having facilities to insert a node at any position and to delete a node with particular information.

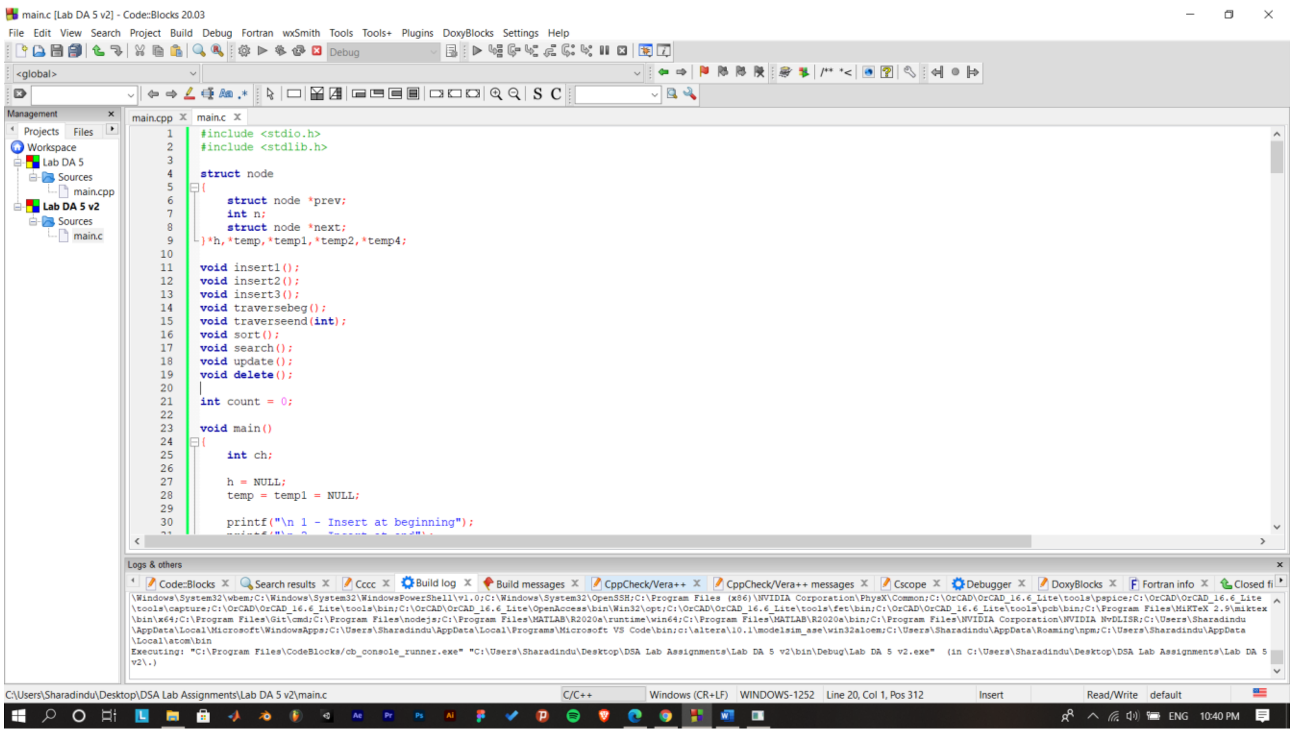
Code, SS & CMD:

```
#include <stdio.h>
#include <stdlib.h>

struct node
{
    struct node *prev;
    int n;
    struct node *next;
}*h,*temp,*temp1,*temp2,*temp4;

void insert1();
void insert2();
void insert3();
void traversebeg();
void traverseend(int);
void sort();
void search();
void update();
void delete();

int count = 0;

void main()
{
    int ch;

    h = NULL;
    temp = temp1 = NULL;

    printf("\n 1 - Insert at beginning");
    printf("\n 2 - Insert at end");
    printf("\n 3 - Insert at position i");
    printf("\n 4 - Delete at i");
    printf("\n 5 - Display from beginning");
    printf("\n 6 - Display from end");
    printf("\n 7 - Search for element");
    printf("\n 8 - Sort the list");
    printf("\n 9 - Update an element");
    printf("\n 10 - Exit");

    while (1)
    {
        printf("\n Enter choice : ");
        scanf("%d", &ch);
        switch (ch)
        {
        case 1:
            insert1();
            break;
            case 2:
                insert2();
                break;
            case 3:
                insert3();
                break;
            case 4:
                delete();
                break;
            case 5:
                traversebeg();
                break;
            case 6:
                temp2 = h;
                if (temp2 == NULL)
                    printf("\n Error : List empty to display ");
                else
                {
                    printf("\n Reverse order of linked list is : ");
                    traverseend(temp2->n);
                }
                break;
            case 7:
                search();
                break;
            case 8:
                sort();
                break;
            case 9:
                update();
                break;
            case 10:
                exit(0);
            default:
                printf("\n Wrong choice menu");
            }
        }
    }

    /* TO create an empty node */
    void create()
    {
        int data;

        temp =(struct node *)malloc(1*sizeof(struct node));
        temp->prev = NULL;
        temp->next = NULL;
        printf("\n Enter value to node : ");
        scanf("%d", &data);
        temp->n = data;
        count++;
    }

    /*  TO insert at beginning */
    void insert1()
    {
        if (h == NULL)
        {
            create();
            h = temp;
            temp1 = h;
        }
        else
        {
            create();
            temp->next = h;
            h->prev = temp;
            h = temp;
```

```c
        }
    }

    /* To insert at end */
    void insert2()
    {
        if (h == NULL)
        {
            create();
            h = temp;
            temp1 = h;
        }
        else
        {
            create();
            temp1->next = temp;
            temp->prev = temp1;
            temp1 = temp;
        }
    }

    /* To insert at any position */
    void insert3()
    {
        int pos, i = 2;

        printf("\n Enter position to be inserted : ");
        scanf("%d", &pos);
        temp2 = h;

        if ((pos < 1) || (pos >= count + 1))
        {
            printf("\n Position out of range to insert");
            return;
        }
        if ((h == NULL) && (pos != 1))
        {
            printf("\n Empty list cannot insert other than 1st position");
            return;
        }
        if ((h == NULL) && (pos == 1))
        {
            create();
            h = temp;
            temp1 = h;
            return;
        }
        else
        {
            while (i < pos)
            {
                temp2 = temp2->next;
                i++;
            }
            create();
            temp->prev = temp2;
            temp->next = temp2->next;
            temp2->next->prev = temp;
            temp2->next = temp;
        }
    }

    /* To delete an element */
    void delete()
    {
        int i = 1, pos;

        printf("\n Enter position to be deleted : ");
        scanf("%d", &pos);
        temp2 = h;

        if ((pos < 1) || (pos >= count + 1))
        {
            printf("\n Error : Position out of range to delete");
            return;
        }
        if (h == NULL)
        {
            printf("\n Error : Empty list no elements to delete");
            return;
        }
        else
        {
            while (i < pos)
            {
                temp2 = temp2->next;
                i++;
            }
            if (i == 1)
            {
                if (temp2->next == NULL)
                {
                    printf("Node deleted from list");
                    free(temp2);
                    temp2 = h = NULL;
                    return;
                }
            }
            if (temp2->next == NULL)
            {
                temp2->prev->next = NULL;
                free(temp2);
                printf("Node deleted from list");
                return;
            }
            temp2->next->prev = temp2->prev;
            if (i != 1)
                temp2->prev->next = temp2->next;    /* Might not need this statement if i
    == 1 check */
            if (i == 1)
                h = temp2->next;
            printf("\n Node deleted");
            free(temp2);
        }
        count--;
    }

    /* Traverse from beginning */
    void traversebeg()
    {
        temp2 = h;

        if (temp2 == NULL)
        {
            printf("List empty to display \n");
            return;
        }
        printf("\n Linked list elements from begining : ");

        while (temp2->next != NULL)
        {
            printf(" %d ", temp2->n);
            temp2 = temp2->next;
        }
        printf(" %d ", temp2->n);
    }
```

```
/* To traverse from end recursively */
void traverseend(int i)
{
    if (temp2 != NULL)
    {
        i = temp2->n;
        temp2 = temp2->next;
        traverseend(i);
        printf(" %d ", i);
    }
}

/* To search for an element in the list */
void search()
{
    int data, count = 0;
    temp2 = h;

    if (temp2 == NULL)
    {
        printf("\n Error : List empty to search for data");
        return;
    }
    printf("\n Enter value to search : ");
    scanf("%d", &data);
    while (temp2 != NULL)
    {
        if (temp2->n == data)
        {
            printf("\n Data found in %d position",count + 1);
            return;
        }
        else
             temp2 = temp2->next;
            count++;
    }
    printf("\n Error : %d not found in list", data);
}

/* To update a node value in the list */
void update()
{
    int data, data1;

    printf("\n Enter node data to be updated : ");
    scanf("%d", &data);
    printf("\n Enter new data : ");
    scanf("%d", &data1);
    temp2 = h;
    if (temp2 == NULL)
    {
        printf("\n Error : List empty no node to update");
        return;
    }
    while (temp2 != NULL)
    {
        if (temp2->n == data)
        {

            temp2->n = data1;
            traversebeg();
            return;
        }
        else
            temp2 = temp2->next;
    }
```

```c
    printf("\n Error : %d not found in list to update", data);
}

/* To sort the linked list */
void sort()
{
    int i, j, x;

    temp2 = h;
    temp4 = h;

    if (temp2 == NULL)
    {
        printf("\n List empty to sort");
        return;
    }

    for (temp2 = h; temp2 != NULL; temp2 = temp2->next)
    {
        for (temp4 = temp2->next; temp4 != NULL; temp4 = temp4->next)
        {
            if (temp2->n > temp4->n)
            {
                x = temp2->n;
                temp2->n = temp4->n;
                temp4->n = x;
            }
        }
    }
    traversebeg();
}
```

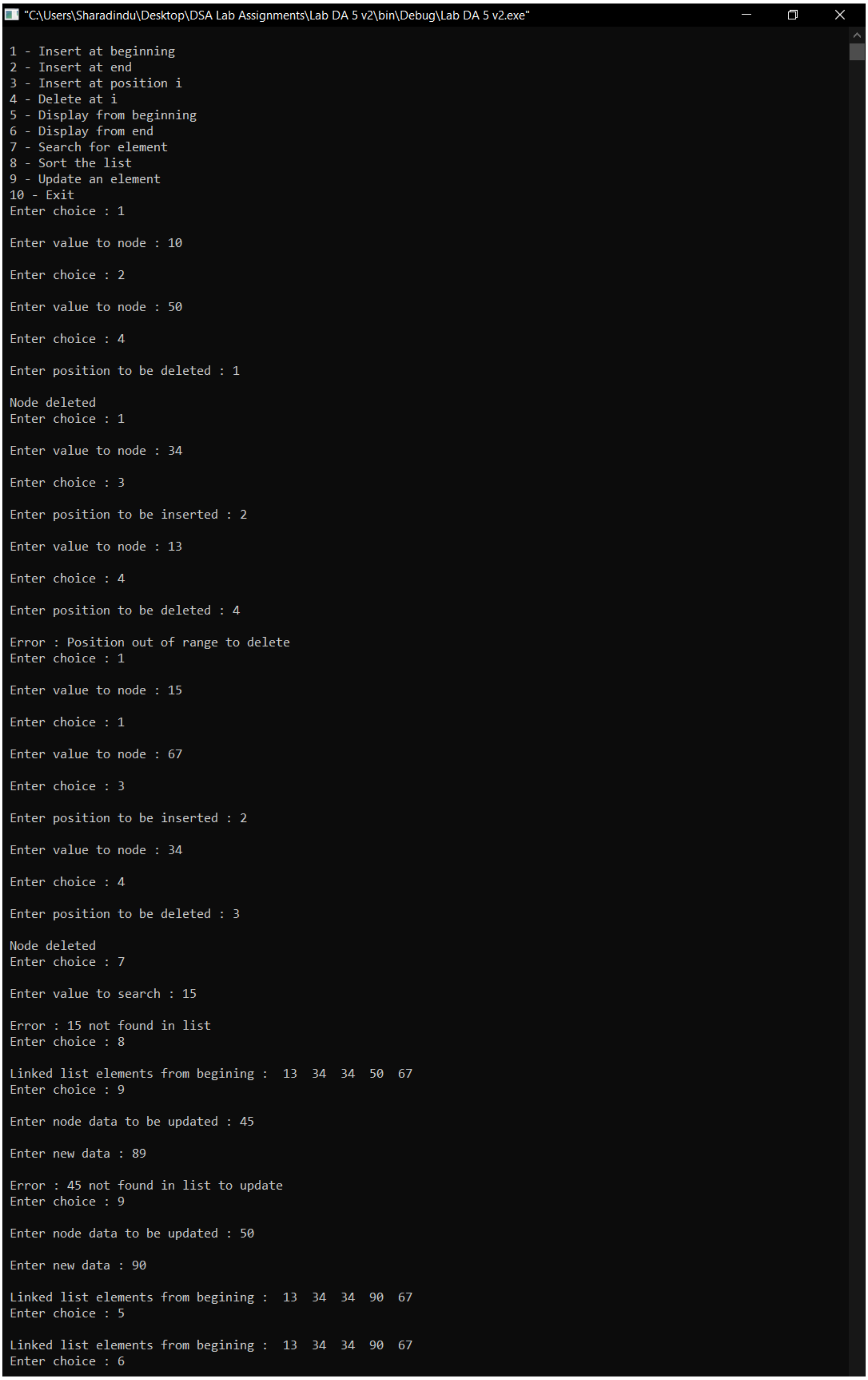


main.c [Lab DA 5 v2] - Code::Blocks 20.03

```c
#include <stdio.h>
#include <stdlib.h>

struct node
{
    struct node *prev;
    int n;
    struct node *next;
}*h,*temp,*temp1,*temp2,*temp4;

void insert1();
void insert2();
void insert3();
void traversebeg();
void traverseend(int);
void sort();
void search();
void update();
void delete();

int count = 0;

void main()
{
    int ch;

    h = NULL;
    temp = temp1 = NULL;

    printf("\n 1 - Insert at beginning");
```

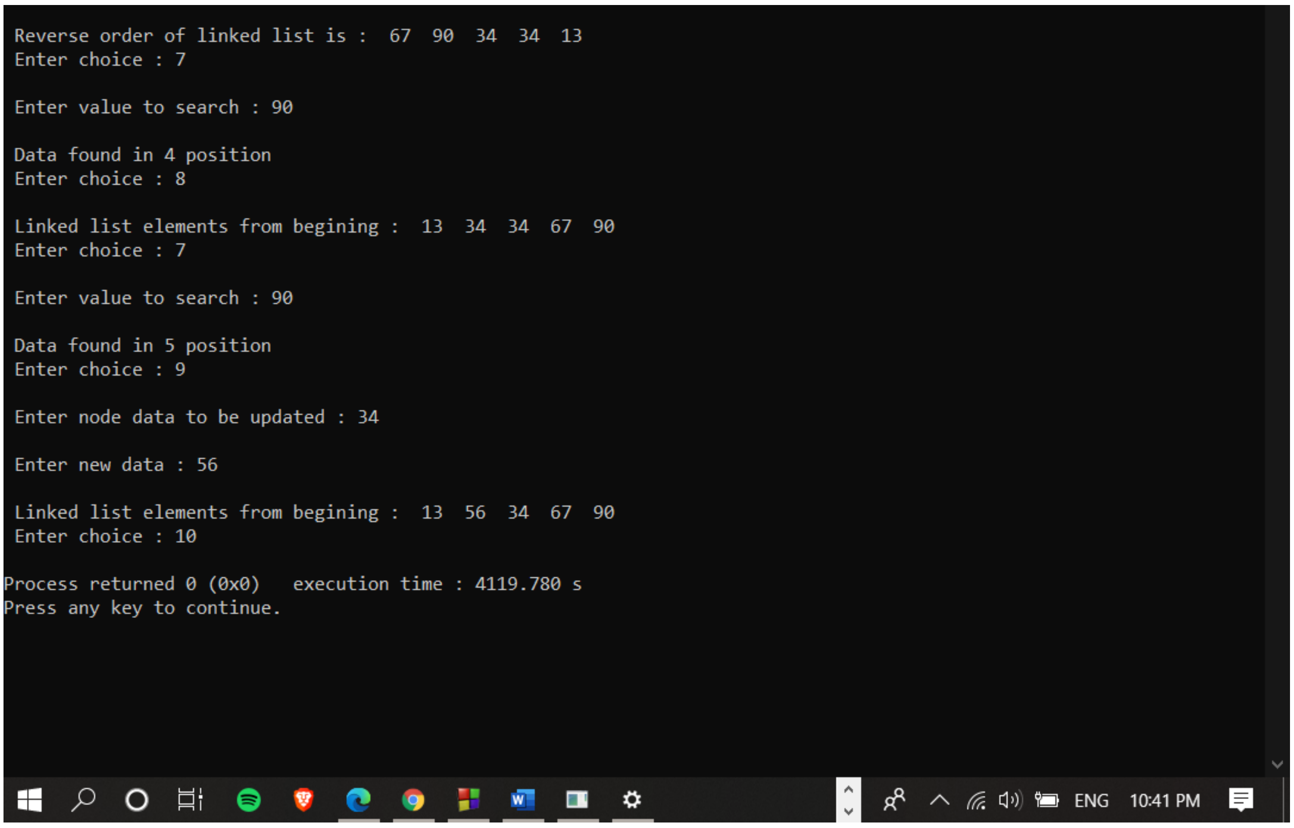
```
\Windows\System32\wbem;C:\Windows\System32\WindowsPowerShell\v1.0;C:\Windows\System32\OpenSSH;C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common;C:\OrCAD\OrCAD_16.6_Lite\tools\pspice;C:\OrCAD\OrCAD_16.6_Lite
\tools\capture;C:\OrCAD\OrCAD_16.6_Lite\tools\bin;C:\OrCAD\OrCAD_16.6_Lite\OpenAccess\bin\Win32\opt;C:\OrCAD\OrCAD_16.6_Lite\tools\fet\bin;C:\OrCAD\OrCAD_16.6_Lite\tools\pcb\bin;C:\Program Files\MiKTeX 2.9\miktex
\bin\x64;C:\Program Files\Git\cmd;C:\Program Files\nodejs;C:\Program Files\MATLAB\R2020a\runtime\win64;C:\Program Files\MATLAB\R2020a\bin;C:\Program Files\NVIDIA Corporation\NVIDIA NvDLISR;C:\Users\Sharadindu
\AppData\Local\Microsoft\WindowsApps;C:\Users\Sharadindu\AppData\Local\Programs\Microsoft VS Code\bin;c:\altera\10.1\modelsim_ase\win32aloem;C:\Users\Sharadindu\AppData\Roaming\npm;C:\Users\Sharadindu\AppData
\Local\atom\bin
Executing: "C:\Program Files\CodeBlocks/cb_console_runner.exe" "C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5 v2\bin\Debug\Lab DA 5 v2.exe"  (in C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5
v2\.)
```

C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5 v2\main.c | C/C++ | Windows (CR+LF) | WINDOWS-1252 | Line 20, Col 1, Pos 312 | Insert | Read/Write | default



"C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5 v2\bin\Debug\Lab DA 5 v2.exe"

```
1 - Insert at beginning
2 - Insert at end
3 - Insert at position i
4 - Delete at i
5 - Display from beginning
6 - Display from end
7 - Search for element
8 - Sort the list
9 - Update an element
10 - Exit
Enter choice : 1

Enter value to node : 10

Enter choice : 2

Enter value to node : 50

Enter choice : 4

Enter position to be deleted : 1

Node deleted
Enter choice : 1

Enter value to node : 34

Enter choice : 3

Enter position to be inserted : 2

Enter value to node : 13

Enter choice : 4

Enter position to be deleted : 4

Error : Position out of range to delete
Enter choice : 1

Enter value to node : 15

Enter choice : 1

Enter value to node : 67

Enter choice : 3

Enter position to be inserted : 2

Enter value to node : 34

Enter choice : 4

Enter position to be deleted : 3

Node deleted
Enter choice : 7

Enter value to search : 15

Error : 15 not found in list
Enter choice : 8

Linked list elements from begining :  13  34  34  50  67
Enter choice : 9

Enter node data to be updated : 45

Enter new data : 89

Error : 45 not found in list to update
Enter choice : 9

Enter node data to be updated : 50

Enter new data : 90

Linked list elements from begining :  13  34  34  90  67
Enter choice : 5

Linked list elements from begining :  13  34  34  90  67
Enter choice : 6
```



```
 Reverse order of linked list is :  67  90  34  34  13
 Enter choice : 7

 Enter value to search : 90

 Data found in 4 position
 Enter choice : 8

 Linked list elements from begining :  13  34  34  67  90
 Enter choice : 7

 Enter value to search : 90

 Data found in 5 position
 Enter choice : 9

 Enter node data to be updated : 34

 Enter new data : 56

 Linked list elements from begining :  13  56  34  67  90
 Enter choice : 10

Process returned 0 (0x0)   execution time : 4119.780 s
Press any key to continue.
```

Question 2

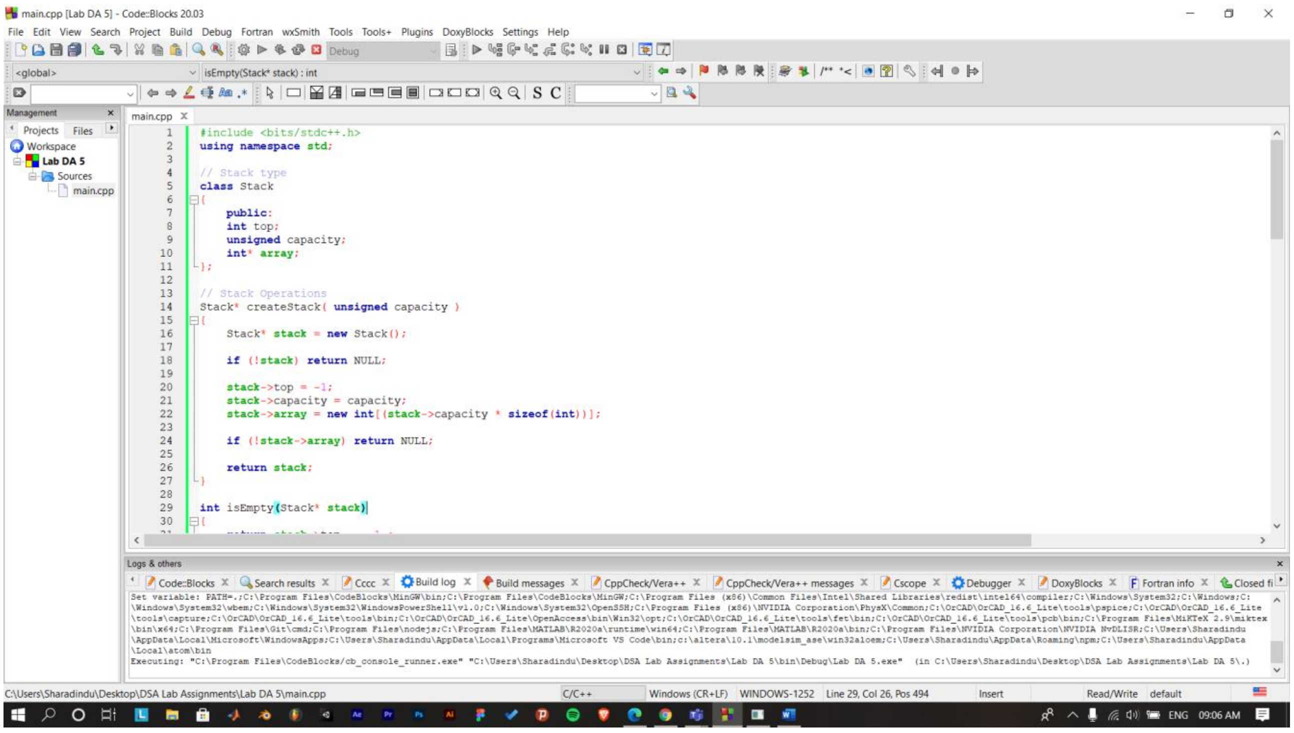
Problem:

WAP to evaluate a postfix expression using a linked stack implementation.

Code, SS & CMD:

```
#include <bits/stdc++.h>
using namespace std;

// Stack type
class Stack
{
    public:
    int top;
    unsigned capacity;
    int* array;
};

// Stack Operations
Stack* createStack( unsigned capacity )
{
    Stack* stack = new Stack();

    if (!stack) return NULL;

    stack->top = -1;
    stack->capacity = capacity;
    stack->array = new int[(stack->capacity * sizeof(int))];

    if (!stack->array) return NULL;

    return stack;
}

int isEmpty(Stack* stack)
{
    return stack->top == -1 ;
}

int peek(Stack* stack)
{
    return stack->array[stack->top];
}

int pop(Stack* stack)
{
    if (!isEmpty(stack))
        return stack->array[stack->top--] ;
    return '$';
}

void push(Stack* stack,int op)
{
    stack->array[++stack->top] = op;
}
```

```
// The main function that returns value
// of a given postfix expression
int evaluatePostfix(char* exp)
{
    // Create a stack of capacity equal to expression size
    Stack* stack = createStack(strlen(exp));
    int i;

    // See if stack was created successfully
    if (!stack) return -1;

    // Scan all characters one by one
    for (i = 0; exp[i]; ++i)
    {
        //if the character is blank space then continue
        if(exp[i] == ' ')continue;

        // If the scanned character is an
        // operand (number here),extract the full number
        // Push it to the stack.
        else if (isdigit(exp[i]))
        {
            int num=0;

            //extract full number
            while(isdigit(exp[i]))
            {
            num = num * 10 + (int)(exp[i] - '0');
                i++;
            }
            i--;

            //push the element in the stack
            push(stack,num);
        }

        // If the scanned character is an operator, pop two
        // elements from stack apply the operator
        else
        {
            int val1 = pop(stack);
            int val2 = pop(stack);

            switch (exp[i])
            {
            case '+': push(stack, val2 + val1); break;
            case '-': push(stack, val2 - val1); break;
            case '*': push(stack, val2 * val1); break;
            case '/': push(stack, val2/val1); break;

            }
        }
    }
    return pop(stack);
}

// Driver code
int main()
{
    char exp[] = "100 200 + 2 / 5 * 7 +";
    cout << evaluatePostfix(exp);
    return 0;
}
```

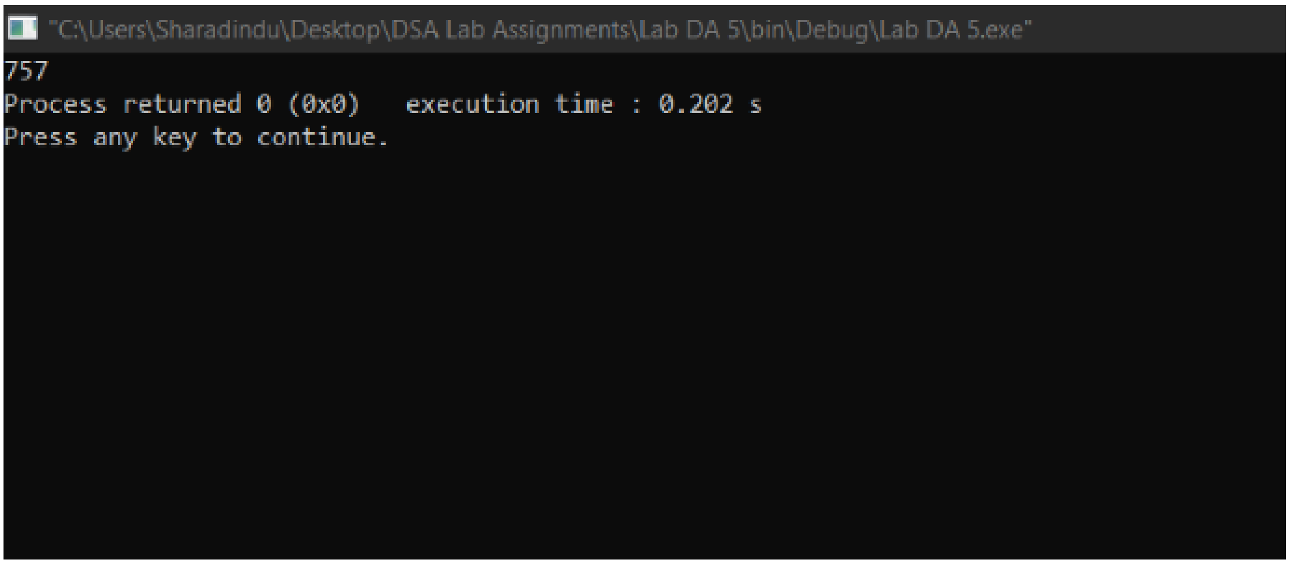


main.cpp [Lab DA 5] - Code::Blocks 20.03

```cpp
#include <bits/stdc++.h>
using namespace std;

// Stack type
class Stack
{
    public:
    int top;
    unsigned capacity;
    int* array;
};

// Stack Operations
Stack* createStack( unsigned capacity )
{
    Stack* stack = new Stack();

    if (!stack) return NULL;

    stack->top = -1;
    stack->capacity = capacity;
    stack->array = new int[(stack->capacity * sizeof(int))];

    if (!stack->array) return NULL;

    return stack;
}

int isEmpty(Stack* stack)
{
```

```
Set variable: PATH=.;C:\Program Files\CodeBlocks\MinGW\bin;C:\Program Files\CodeBlocks\MinGW;C:\Program Files (x86)\Common Files\Intel\Shared Libraries\redist\intel64\compiler;C:\Windows\System32;C:\Windows;C:
\Windows\System32\wbem;C:\Windows\System32\WindowsPowerShell\v1.0;C:\Windows\System32\OpenSSH;C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common;C:\OrCAD\OrCAD_16.6_Lite\tools\pspice;C:\OrCAD\OrCAD_16.6_Lite
\tools\capture;C:\OrCAD\OrCAD_16.6_Lite\tools\bin;C:\OrCAD\OrCAD_16.6_Lite\OpenAccess\bin\Win32\opt;C:\OrCAD\OrCAD_16.6_Lite\tools\fet\bin;C:\OrCAD\OrCAD_16.6_Lite\tools\pcb\bin;C:\Program Files\MiKTeX 2.9\miktex
\bin\x64;C:\Program Files\Git\cmd;C:\Program Files\nodejs;C:\Program Files\MATLAB\R2020a\runtime\win64;C:\Program Files\MATLAB\R2020a\bin;C:\Program Files\NVIDIA Corporation\NVIDIA NvDLISR;C:\Users\Sharadindu
\AppData\Local\Microsoft\WindowsApps;C:\Users\Sharadindu\AppData\Local\Programs\Microsoft VS Code\bin;c:\altera\10.1\modelsim_ase\win32aloem;C:\Users\Sharadindu\AppData\Roaming\npm;C:\Users\Sharadindu\AppData
\Local\atom\bin
Executing: "C:\Program Files\CodeBlocks/cb_console_runner.exe" "C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5\bin\Debug\Lab DA 5.exe"  (in C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5\.)
```

C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5\main.cpp | C/C++ | Windows (CR+LF) | WINDOWS-1252 | Line 29, Col 26, Pos 494 | Insert | Read/Write | default



"C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5\bin\Debug\Lab DA 5.exe"

```
757
Process returned 0 (0x0)   execution time : 0.202 s
Press any key to continue.
```

Question 3

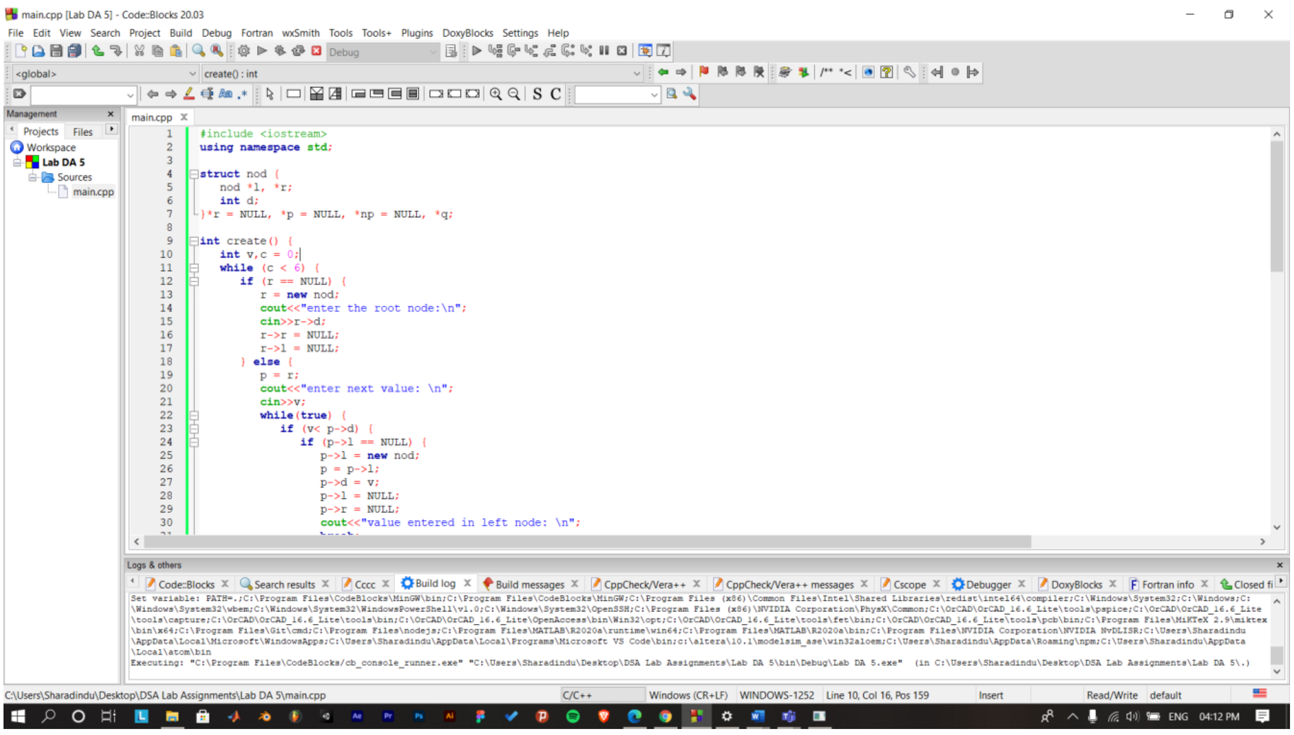
Problem:

WAP for creation of binary tree and traverse it in Pre, Post and Inorder. Show the traversal output.

Code:

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

struct nod {
   nod *l, *r;
   int d;
}*r = NULL, *p = NULL, *np = NULL, *q;

int create() {
   int v,c = 0;
   while (c < 6) {
      if (r == NULL) {
         r = new nod;
         cout<<"enter the root node:\n";
         cin>>r->d;
         r->r = NULL;
         r->l = NULL;
      } else {
         p = r;
         cout<<"enter next value: \n";
         cin>>v;
         while(true) {
            if (v< p->d) {
               if (p->l == NULL) {
                  p->l = new nod;
                  p = p->l;
                  p->d = v;
                  p->l = NULL;
                  p->r = NULL;
                  cout<<"value entered in left node: \n";
                  break;
               } else if (p->l != NULL) {
                  p = p->l;
               }
            } else if (v >p->d) {
               if (p->r == NULL) {
                  p->r = new nod;
                  p = p->r;
                  p->d = v;
                  p->l = NULL;
                  p->r = NULL;
                  cout<<"value entered in right node: \n";
                  break;
               } else if (p->r != NULL) {
                  p = p->r;
               }
            }
         }
      }
      c++;
   }
}

void inorder(nod *p) {
   if (p != NULL) {
      inorder(p->l);
      cout<<p->d<<endl;
      inorder(p->r);
   }
}

void preorder(nod *p) {
   if (p != NULL) {
      cout<<p->d<<endl;
      preorder(p->l);
      preorder(p->r);
   }
}

void postorder(nod *p) {
   if (p != NULL) {
      postorder(p->l);
      postorder(p->r);
      cout<<p->d<<endl;
   }
}

int main() {
   create();
   cout<<" Inorder traversal\n";
   inorder(r);
   cout<<" Preorder traversal\n";
   preorder(r);
   cout<<" Postorder traversal \n";
   postorder(r);
}
```

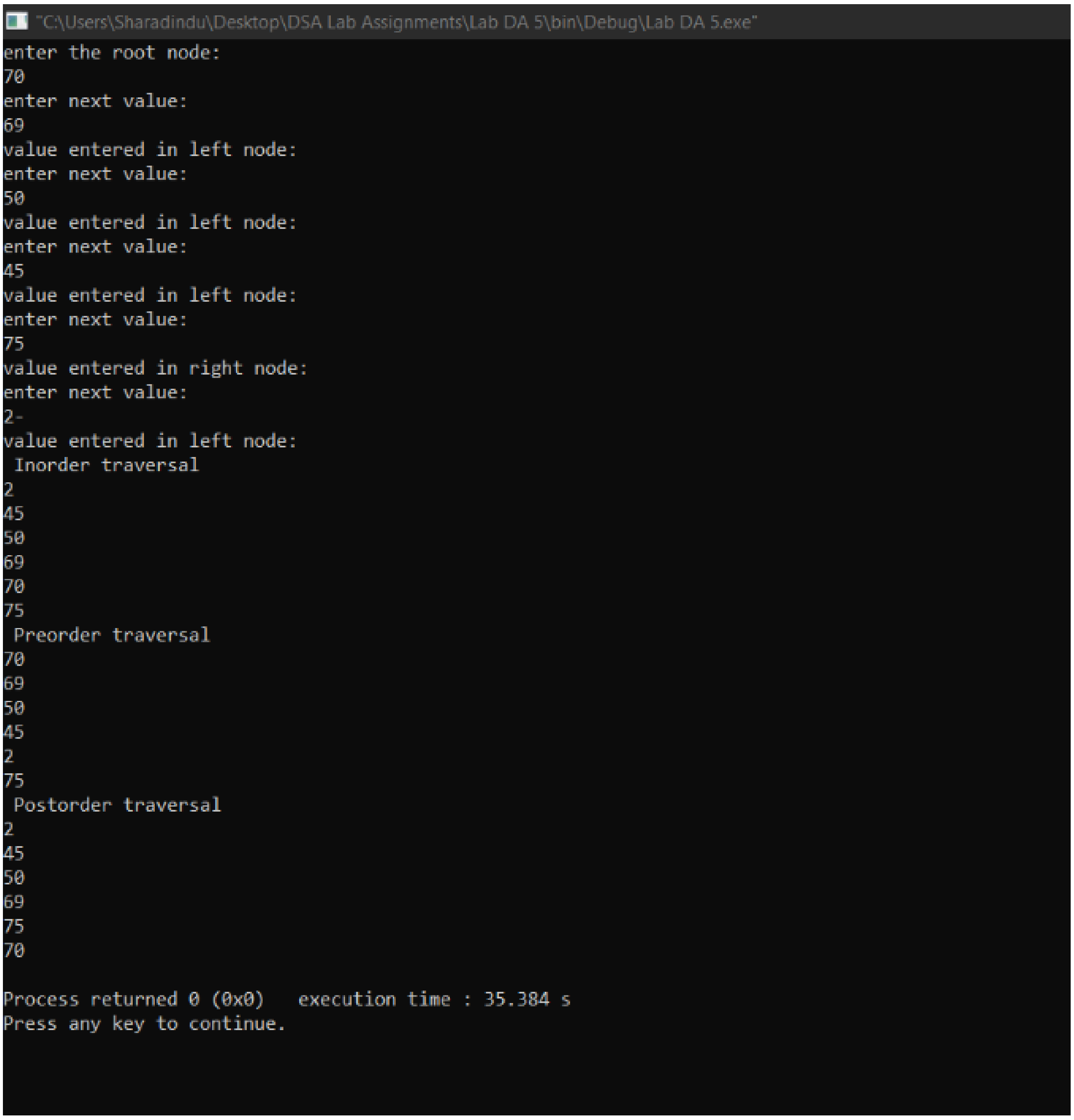
Screenshot of Code & CMD:



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

struct nod {
   nod *l, *r;
   int d;
}*r = NULL, *p = NULL, *np = NULL, *q;

int create() {
   int v,c = 0;
   while (c < 6) {
      if (r == NULL) {
         r = new nod;
         cout<<"enter the root node:\n";
         cin>>r->d;
         r->r = NULL;
         r->l = NULL;
      } else {
         p = r;
         cout<<"enter next value: \n";
         cin>>v;
         while(true) {
            if (v< p->d) {
               if (p->l == NULL) {
                  p->l = new nod;
                  p = p->l;
                  p->d = v;
                  p->l = NULL;
                  p->r = NULL;
                  cout<<"value entered in left node: \n";
```

```
Set variable: PATH=.;C:\Program Files\CodeBlocks\MinGW\bin;C:\Program Files\CodeBlocks\MinGW;C:\Program Files (x86)\Common Files\Intel\Shared Libraries\redist\intel64\compiler;C:\Windows\System32;C:\Windows;C:
\Windows\System32\wbem;C:\Windows\System32\WindowsPowerShell\v1.0;C:\Windows\System32\OpenSSH;C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common;C:\OrCAD\OrCAD_16.6_Lite\tools\pspice;C:\OrCAD\OrCAD_16.6_Lite
\tools\capture;C:\OrCAD\OrCAD_16.6_Lite\tools\bin;C:\OrCAD\OrCAD_16.6_Lite\OpenAccess\bin\Win32\opt;C:\OrCAD\OrCAD_16.6_Lite\tools\fet\bin;C:\OrCAD\OrCAD_16.6_Lite\tools\pcb\bin;C:\Program Files\MiKTeX 2.9\miktex
\bin\x64;C:\Program Files\Git\cmd;C:\Program Files\nodejs;C:\Program Files\MATLAB\R2020a\runtime\win64;C:\Program Files\MATLAB\R2020a\bin;C:\Program Files\NVIDIA Corporation\NVIDIA NvDLISR;C:\Users\Sharadindu
\AppData\Local\Microsoft\WindowsApps;C:\Users\Sharadindu\AppData\Local\Programs\Microsoft VS Code\bin;c:\altera\10.1\modelsim_ase\win32aloem;C:\Users\Sharadindu\AppData\Roaming\npm;C:\Users\Sharadindu\AppData
\Local\atom\bin
Executing: "C:\Program Files\CodeBlocks/cb_console_runner.exe" "C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5\bin\Debug\Lab DA 5.exe"  (in C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5\.)
```

main.cpp [Lab DA 5] - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Debug
<global>
create() : int
Management
Projects Files
Workspace
Lab DA 5
Sources
main.cpp
Logs & others
Code::Blocks Search results Cccc Build log Build messages CppCheck/Vera++ CppCheck/Vera++ messages Cscope Debugger DoxyBlocks Fortran info Closed fi
C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5\main.cpp
C/C++ Windows (CR+LF) WINDOWS-1252 Line 10, Col 16, Pos 159 Insert Read/Write default
ENG 04:12 PM



"C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5\bin\Debug\Lab DA 5.exe"

```
enter the root node:
70
enter next value:
69
value entered in left node:
enter next value:
50
value entered in left node:
enter next value:
45
value entered in left node:
enter next value:
75
value entered in right node:
enter next value:
2-
value entered in left node:
 Inorder traversal
2
45
50
69
70
75
 Preorder traversal
70
69
50
45
2
75
 Postorder traversal
2
45
50
69
75
70

Process returned 0 (0x0)   execution time : 35.384 s
Press any key to continue.
```

Question 4

Problem:

WAP to construct an expression tree for a given arithmetic expression, check its correctness by traversing it in Inorder.

Code & Input:

ALGORITHM EXPLANATION:

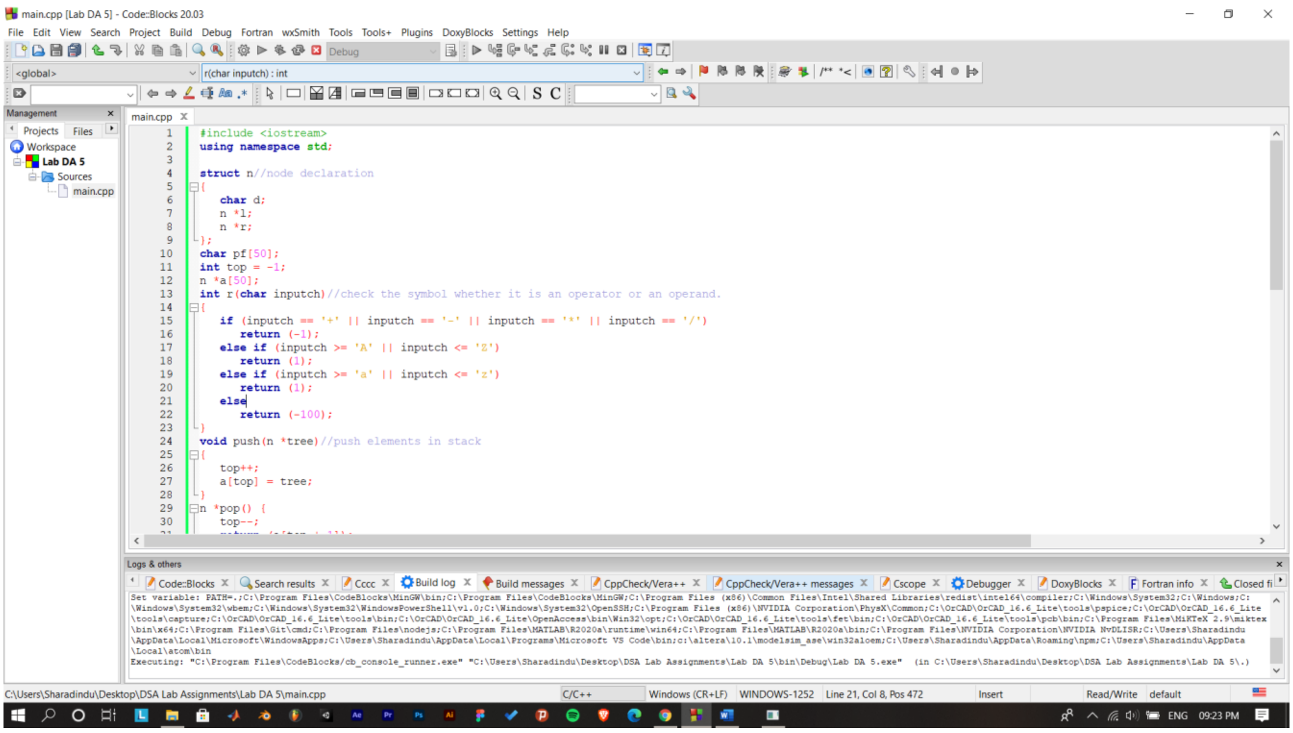
```
Construction of Expression Tree
We consider that a postfix expression is given as an input for constructing an
expression tree. Following are the step to construct an expression tree:
1.    Read one symbol at a time from the postfix expression.
2.    Check if the symbol is an operand or operator.
3.    If the symbol is an operand, create a one node tree and pushed a pointer
onto a stack
4.    If the symbol is an operator, pop two pointer from the stack namely T1 & T2
and form a new tree with root as the operator, T1 & T2 as a left and right child
5.    A pointer to this new tree is pushed onto the stack


So based on the above algorithm we have the following code where we first get the
expression tree and then from the expression tree we perform inorder traversal to
get the infix expression as a result.
```

So the code for the above is:

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

struct n//node declaration
{
   char d;
   n *l;
   n *r;
};
char pf[50];
int top = -1;
n *a[50];
int r(char inputch)//check the symbol whether it is an operator or an operand.
{
   if (inputch == '+' || inputch == '-' || inputch == '*' || inputch == '/')
      return (-1);
   else if (inputch >= 'A' || inputch <= 'Z')
      return (1);
   else if (inputch >= 'a' || inputch <= 'z')
      return (1);
   else
      return (-100);
}
void push(n *tree)//push elements in stack
{
   top++;
   a[top] = tree;
}
```

```
n *pop() {
   top--;
   return (a[top + 1]);
}
void construct_expression_tree(char *suffix) {
   char s;
   n *newl, *p1, *p2;
   int flag;
   s = suffix[0];
   for (int i = 1; s!= 0; i++) {
      flag = r(s);
      if (flag == 1) {
         newl = new n;
         newl->d = s;
         newl->l = NULL;
         newl->r = NULL;
         push(newl);
      } else {
         p1 = pop();
         p2 = pop();
         newl = new n;
         newl->d = s;
         newl->l = p2;
         newl->r = p1;
         push(newl);
      }
      s = suffix[i];
   }
}

void inOrder(n *tree)//perform inorder traversal
{
   if (tree != NULL) {
      inOrder(tree->l);
      cout << tree->d;
      inOrder(tree->r);
   }
}

int main(int argc, char **argv) {
   cout << "Enter Postfix Expression : ";
   cin >>pf;
   construct_expression_tree(pf);
   cout << "\nInfix Expression : ";
   inOrder(a[0]);
   return 0;
}
```

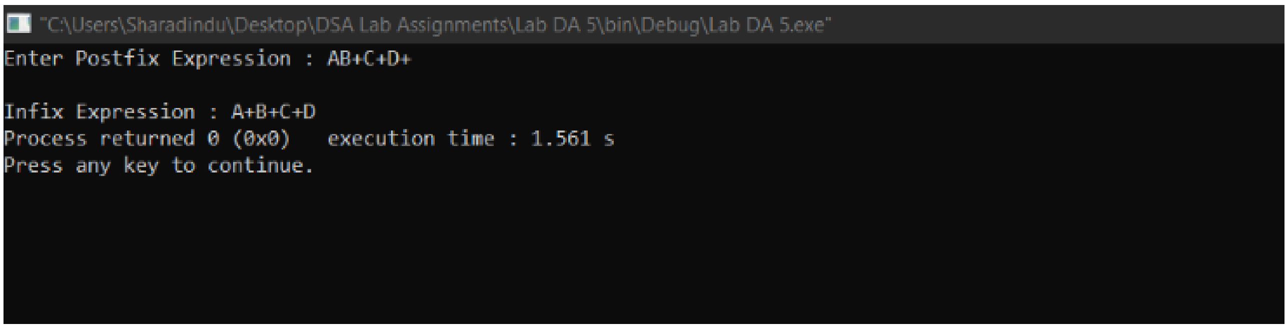
Screenshot of Code & Output:



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

struct n//node declaration
{
    char d;
    n *l;
    n *r;
};
char pf[50];
int top = -1;
n *a[50];
int r(char inputch)//check the symbol whether it is an operator or an operand.
{
    if (inputch == '+' || inputch == '-' || inputch == '*' || inputch == '/')
        return (-1);
    else if (inputch >= 'A' || inputch <= 'Z')
        return (1);
    else if (inputch >= 'a' || inputch <= 'z')
        return (1);
    else
        return (-100);
}
void push(n *tree)//push elements in stack
{
    top++;
    a[top] = tree;
}
n *pop() {
    top--;
```

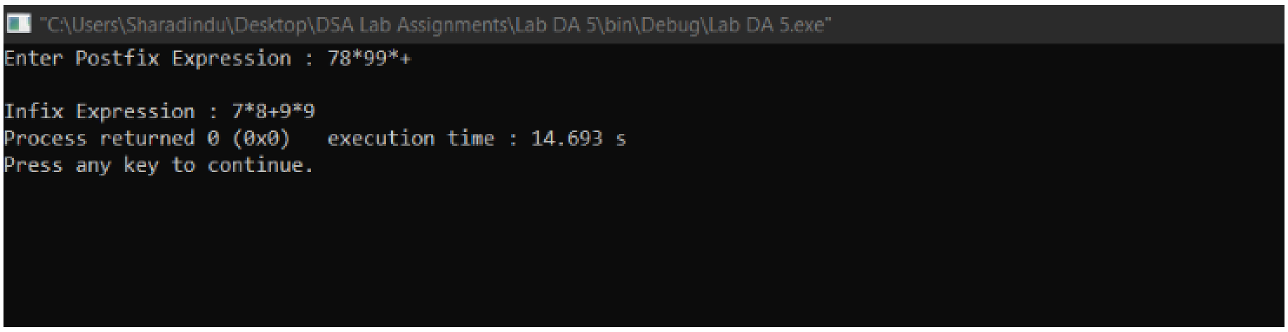
```
Set variable: PATH=.;C:\Program Files\CodeBlocks\MinGW\bin;C:\Program Files\CodeBlocks\MinGW;C:\Program Files (x86)\Common Files\Intel\Shared Libraries\redist\intel64\compiler;C:\Windows\System32;C:\Windows;C:\Windows\System32\wbem;C:\Windows\System32\WindowsPowerShell\v1.0;C:\Windows\System32\OpenSSH;C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common;C:\OrCAD\OrCAD_16.6_Lite\tools\pspice;C:\OrCAD\OrCAD_16.6_Lite\tools\capture;C:\OrCAD\OrCAD_16.6_Lite\tools\bin;C:\OrCAD\OrCAD_16.6_Lite\OpenAccess\bin\Win32\opt;C:\OrCAD\OrCAD_16.6_Lite\tools\fet\bin;C:\OrCAD\OrCAD_16.6_Lite\tools\pcb\bin;C:\Program Files\MiKTeX 2.9\miktex\bin\x64;C:\Program Files\Git\cmd;C:\Program Files\nodejs;C:\Program Files\MATLAB\R2020a\runtime\win64;C:\Program Files\MATLAB\R2020a\bin;C:\Program Files\NVIDIA Corporation\NVIDIA NvDLISR;C:\Users\Sharadindu\AppData\Local\Microsoft\WindowsApps;C:\Users\Sharadindu\AppData\Local\Programs\Microsoft VS Code\bin;c:\altera\10.1\modelsim_ase\win32aloem;C:\Users\Sharadindu\AppData\Roaming\npm;C:\Users\Sharadindu\AppData\Local\atom\bin
Executing: "C:\Program Files\CodeBlocks/cb_console_runner.exe" "C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5\bin\Debug\Lab DA 5.exe"  (in C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5\.)
```

main.cpp [Lab DA 5] - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5\main.cpp
C/C++ Windows (CR+LF) WINDOWS-1252 Line 21, Col 8, Pos 472 Insert Read/Write default
09:23 PM



"C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5\bin\Debug\Lab DA 5.exe"

```
Enter Postfix Expression : AB+C+D+

Infix Expression : A+B+C+D
Process returned 0 (0x0)   execution time : 1.561 s
Press any key to continue.
```



"C:\Users\Sharadindu\Desktop\DSA Lab Assignments\Lab DA 5\bin\Debug\Lab DA 5.exe"

```
Enter Postfix Expression : 78*99*+

Infix Expression : 7*8+9*9
Process returned 0 (0x0)   execution time : 14.693 s
Press any key to continue.
```