**Practical – 1**

AIM: Implementation and time analysis of Stack and its applications (infix, postfix, prefix).  
  
#include <iostream>

using namespace std;

int stack[100], n = 100, top = -1;

void push(int value)

{

if (top >= n - 1)

{

cout << "stack is full" << endl;

}

else

{

top++;

stack[top] = value;

}

}

void pop()

{

if (top <= -1)

{

cout << "stack is empty" << endl;

}

else

{

cout << "The popped element is " << stack[top] << endl;

top--;

}

}

void showTop()

{

if (top < 0)

{

cout << "Empty stack" << endl;

return;

}

cout << "top element is : " << stack[top] << endl;

}

void display()

{

if (top >= 0)

{

for (int i = top; i >= 0; i--)

{

cout << stack[i] << " ";

}

cout << endl;

}

else

{

cout << "Stack is empty" << endl;

}

}

int main()

{

int choice, value;

cout << "1) Push in stack" << endl;

cout << "2) Pop from stack" << endl;

cout << "3) Display stack" << endl;

cout << "4) Display Top" << endl;

cout << "5) Exit" << endl;

do

{

cout << "Enter choice: ";

cin >> choice;

switch (choice)

{

case 1:

{

cout << "Enter value to be pushed:";

cin >> value;

push(value);

break;

}

case 2:

{

pop();

break;

}

case 3:

{

display();

break;

}

case 4:

{

showTop();

break;

}

case 5:

{

cout << "Exit" << endl;

break;

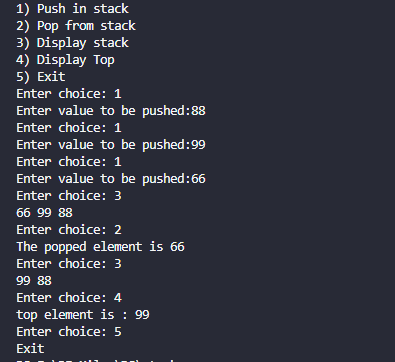
}

}

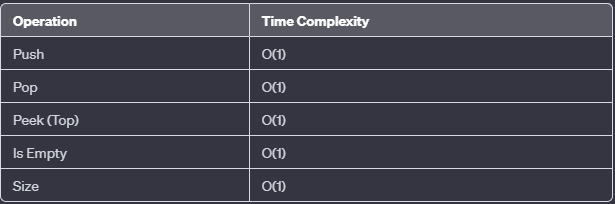
} while (choice != 5);

}

**OUTPUT**



Time analysis



Applications

* Call center systems.
* Function Call Management
* Expression Evaluation
* Backtracking Algorithms
* Undo Mechanisms in Software
* Memory Management
* Parsing and Syntax Analysis
* Task Scheduling
* in Algorithmic Problems
* The history of a web browser is stored in the form of a stack.
* Call logs, E-mails, and Google photos in any gallery are also stored in form of a stack.
* YouTube downloads and Notifications are also shown in LIFO format(the latest appears first ).
* Allocation of memory by an operating system while executing a process.