Aim: - evaluation of the postfix expression

Algorithm

START

For each character in the postfix expression, do

If operand is encountered, push it onto the stack.

Else if operator is encountered, pop two elements out.

A→TOP ELEMENT.

B→NEXT TO THE TOP ELEMENT.

RESULT=B OPERATOR A.

PUSH THE RESULT IN TO THE STACK.

RETURN THE ELEMENT OF THE STACK TOP.

END

PROGRAM: -

#include<stdio.h>//standard input output header file//

#include<ctype.h>//

#include// int stack[20];//initializing the stack size to 20//

int top=-1;//initializing the top value to -1//

void push(int x)//push function//

{

stack[++top]=x;//stack value to the x//

}

int pop()//pop function//

{

return stack[top--];//stack decrement//

}

int main()//main function//

{

char exp[20];//initializing the expression length to the 20//

char \*e;//declaring the pointer variable to \*e//

int n1,n2,n3,num;//declaring the n1,n2,n3,num//

printf("enter the expression\n");//prints the statement//

scanf("%s",exp);//scans the statement and store the value//

e=exp;//e is initialized to the exp//

while(\*e!='\0')//checks the condition//

{

if(isdigit(\*e))//checks the condition//

{

num=\*e-48;//do the operation//

push(num);//pushes the number in to the stack//

}

Else

{

n1=pop();

n2=pop();

switch(\*e)//condition case//

{

case '+':n3=n1+n2;//perform the operation when the condition satisfies//

break;

case '-':n3=n2-n1; //perform the operation when the condition satisfies//

break;

case '\*':n3=n1\*n2; //perform the operation when the condition satisfies//

break;

case '/':n3=n2/n1; //perform the operation when the condition satisfies//

break;

}

push(n3);//pushes the n3 in to the stack//

}

e++;//increment//

}

printf("the result of the expression is %d\n",pop());//prints the statement and the values//

}

