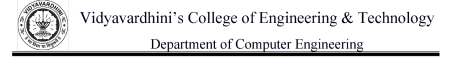
Experiment no 3:Evaluaon of posix Expression using stack ADT Aim: Implementaon of Evaluaon of Posix Expression using stack ADT

Objecve:

1) Understand the use of stack

2) Understand imporng an ADT in an applicaon program

3) Understand the instanaon of stack ADT in an applicaon Program

4) Understand how the member funcon of an ADT are accessed in an applicaon program Theory:

A Queue is an ordered collecon of items from which items may be deleted at one end and into which items may be inserted at the other end .Queues remember things in first-in-first-out(FIFO)order.The basic operaons in a queue are:Enqueue-Adds an item to the end of queue.Dequeue-Removes an item from the front.

Algorithm:

Step 1=If a character is an operand push it to stack.

Step 2=If a character is an operator,Pop two elements from the stack.

Step 3=Step 1 and step 2 will be be repeated until end has reached.

Step 4=End.

Code :

#include<stdio.h>

#include<conio.h>

#include<ctype.h>

int stack[20];

int top=-1;

void push(int x){

stack[++top]=x;

}

int pop(){

return stack[top--];

}

int main(){

char exp[20];

char \*e;

int n1,n2,n3,num;

clrscr();

prin("Enter the expression::");

scanf("%s",exp);

e=exp;

while(\*e!='\0'){

if(isdigit(\*e)){

num=\*e-48;

push(num);

}

else{

n1=pop();

n2=pop();

switch(\*e){

case '+':

{

n3=n1+n2;

break;

}

case '-':

{

n3=n2-n1;

break;

}

case '\*':

{

n3=n1\*n2;

break;

}

case '/':

{

n3=n2/n1;

break;

}

}

push(n3);

}

e++;

}

prin("\n The Result of expression %s=%d\n\n",exp,pop()); getch();

return 0;

}

Output:

Conclusion : From the above output,we can conclude evaluation of postfix expression stack byS using ADT.