**Aim:** Evaluate Postfix Expression using Stack ADT.

#include <ctype.h>

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

struct Stack { int top;

unsigned capacity; int\* array;

};

struct Stack\* createStack(unsigned capacity)

{

struct Stack\* stack

= (struct Stack\*)malloc(sizeof(struct Stack)); if (!stack)

return NULL; stack->top =-1;

stack->capacity = capacity; stack->array

= (int\*)malloc(stack->capacity \* sizeof(int)); if (!stack->array)

return NULL; return stack;

}

int isEmpty(struct Stack\* stack)

{

return stack->top ==-1;

}

char peek(struct Stack\* stack)

{

return stack->array[stack->top];

}

char pop(struct Stack\* stack)

{

if (!isEmpty(stack))

return stack->array[stack->top--]; return '$';

}

void push(struct Stack\* stack, char op)

{

stack->array[++stack->top] = op;

}

int evaluatePostfix(char\* exp)

{

struct Stack\* stack = createStack(strlen(exp));

int i;

if (!stack)

return-1;

for (i = 0; exp[i]; ++i) {

if (isdigit(exp[i]))

push(stack, exp[i]-'0');

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);

}

int main()

{

char exp[] = "231\*+9-";

printf("postfix evaluation: %d", evaluatePostfix(exp)); return 0;

}