

**Program 3. Design, Develop and Implement a Program in C to Evaluation of Suffix expression with single digit operands and operators: +, -, \*, / using Stack.**

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

#define MAX 50

int stack[MAX];
char post[MAX];
int top=-1;

void pushstack(int tmp);
void calculator(char c);

void main()
{
    int i;
    printf("Insert a postfix notation :: ");
    //gets(post);
    scanf("%s",post);

    for(i=0;i<strlen(post);i++) /*reading input postfix notation */
    {
        if(post[i]>='0' && post[i]<='9') /* checking if input is number
or not and then add into stack */
        {
            pushstack(i);
        }
        if(post[i]=='+' || post[i]=='-' || post[i]=='*' || post[i]=='/' ||
post[i]=='^')
        { /* if input is operator then calculate result and update in stack
*/
            calculator(post[i]);
        }
    }

    printf("\n\nResult :: %d",stack[top]);
}
```

```

void pushstack(int tmp)
{
    top++;
    stack[top]=(int)(post[tmp]-48);
}
void calculator(char c)
{
    int a,b,ans;

    a=stack[top];
    stack[top]='\0';
    top--;
    b=stack[top];
    stack[top]='\0';
    top--;

    switch(c) /* calculate result based on operator + , - , * , /
    {
        case '+': ans=b+a;
            break;
        case '-': ans=b-a;
            break;
        case '*': ans=b*a;
            break;
        case '/': ans=b/a;
            break;
        default: ans=0;
    }
    top++;
    stack[top]=ans;

}

```