Home   Project Ideas »   Training Programs **New** »   Downloads »   Campus Experience »   Blog »   Contact Us »

Search...   [Go]

Infix To Postfix Conversion And Expression Evaluation

Code Id	19
Date Updated	3/7/2010
Title	infix to postfix conversion and expression evaluation

Description

This is a program for conversion of infix expression to postfix expression and ev

Codes Snippet

```
#include
#include
#include
#define Blank ' '
#define Tab 't'
#define MAX 50
long int pop ();
long int eval_post();
infix_to_postfix();
push(char);
prec(char);
white_space(char);
char infix[MAX], postfix[MAX];
long int stack[MAX];
int top;
main()
{
        long int value;
        char choice='y';
        while(choice == 'y')
        {
                top = 0;
                printf("Enter infix : ");
                fflush(stdin);
                gets(infix);
                infix_to_postfix();
                printf("Postfix : %sn",postfix);
                value=eval_post();
                printf("Value of expression : %ldn",value);
                printf("Want to continue(y/n) : ");
                scanf("%c",&choice);
        }
}/*End of main()*/
infix_to_postfix()
{
        int i,p=0,type,precedence,len;
        char next ;
        stack[top]='#';
        len=strlen(infix);
        infix[len]='#';
        for(i=0; infix[i]!='#';i++)
        {
        if( !white_space(infix[i]))
                {
                        switch(infix[i])
                        {
                        case '(':
                                push(infix[i]);
                                break;
                        case ')':
                                while((next = pop()) != '(')
                                        postfix[p++] = next;

                                break;
                        case '+':
                        case '-':
                        case '*':
                        case '/':
```

Online Enquiry

Course Registration

Recent Posts

Types of Cloud Computing

What is Cloud Computing ?

How to pass a multi-dimensional array to a function?
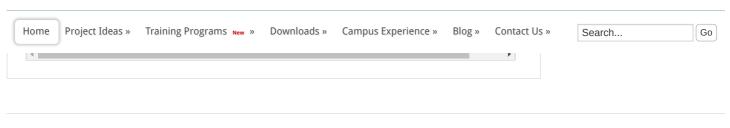
Memory Layout of a C Program

PHP and Its Advantages

Home | Project Ideas » | Training Programs New » | Downloads » | Campus Experience » | Blog » | Contact Us »   Search... Go

```
                while(stack[top]!='#')
                        postfix[p++] = pop();
        postfix[p] = '' ; /*End postfix with'' to make it a string*/
}/*End of infix_to_postfix()*/
/* This function returns the precedence of the operator */
prec(char symbol )
{
        switch(symbol)
        {
        case '(':
                return 0;
        case '+':
        case '-':
                return 1;
        case '*':
        case '/':
        case '%':
                return 2;
        case '^':
                return 3;
        }/*End of switch*/
}/*End of prec()*/

push(long int symbol)
{
        if(top > MAX)
        {
                printf("Stack overflown");
                exit(1);
        }
        else
        {
                top=top+1;
                stack[top] = symbol;
        }
}/*End of push()*/
long int pop()
{
        if (top == -1 )
        {
                printf("Stack underflow n");
                exit(2);
        }
        else
                return (stack[top--]);
}/*End of pop()*/
white_space(char symbol)
{
        if( symbol == Blank || symbol == Tab || symbol == '')
                return 1;
        else
                return 0;
}/*End of white_space()*/
long int eval_post()
{
        long int a,b,temp,result,len;
        int i;
        len=strlen(postfix);
        postfix[len]='#';

        for(i=0;postfix[i]!='#';i++)
        {
                if(postfix[i]<='9' && postfix[i]>='0')
                        push( postfix[i]-48 );
                else
                {
                        a=pop();
                        b=pop();

                        switch(postfix[i])
                        {
                        case '+':
                                temp=b+a; break;
                        case '-':
                                temp=b-a;break;
                        case '*':
                                temp=b*a;break;
                        case '/':
                                temp=b/a;break;
```

Home   Project Ideas »   Training Programs **New** »   Downloads »   Campus Experience »   Blog »   Contact Us »

Search...   Go