C for Beginners

Wednesday, 4 September 2013

Infix to Prefix Conversion using Stacks in C

Infix to Prefix Conversion using Stacks in C

Algorithm to Convert Infix to Prefix Form

Suppose A is an arithmetic expression written in infix form. The algorithm finds equivalent prefix expression B.

```
Step 1. Push ")" onto STACK, and add "(" to end of the A
Step 2. Scan A from right to left and repeat step 3 to 6 for each element of A until the STACK is empty
Step 3. If an operand is encountered add it to B
Step 4. If a right parenthesis is encountered push it onto STACK
Step 5. If an operator is encountered then:
          a. Repeatedly pop from STACK and add to B each operator (on the top of STACK) which has same or higher precedence than the operator.
          b. Add operator to STACK
Step 6. If left parenthesis is encontered then
          a. Repeatedly pop from the STACK and add to B (each operator on top of stack until a left parenthesis is encounterd)
          b. Remove the left parenthesis
Step 7. Exit
```

Program

```
#define SIZE 50 /* Size of Stack */
#include<string.h>
#include <ctype.h>
#include<stdio.h>
char s[SIZE]; int top=-1; /* Global declarations */
push(char elem)
{ /* Function for PUSH operation */
s[++top]=elem;
}
char pop()
{ /* Function for POP operation */
return(s[top--]);
}
int pr(char elem)
{ /* Function for precedence */
switch(elem)
{
case '#': return 0;
case ')': return 1;
case '+':
case '-': return 2;
case '*':
case '/':return 3;
}
}
main()
{ /* Main Program */
char infx[50],prfx[50],ch,elem;
int i=0,k=0;
printf("\n\nRead the Infix Expression ? ");
scanf("%s",infx);
push('#');
strrev(infx);
while( (ch=infx[i++]) != '\0')
{
if( ch == ')')
push(ch);
else if(isalnum(ch))
prfx[k++]=ch;
else if( ch == '(')
```

About Me

Prathuish Kumar

G+ Follow 44

View my complete profile

Blog Archive

- ▼ 2013 (12)
  - ► November (1)
  - ► October (5)
  - ▼ September (2)
    - Infix to Postfix Conversion using stack in C
    - Infix to Prefix Conversion using Stacks in C
  - ► May (1)
  - ► February (1)
  - ► January (2)
- ► 2012 (7)

```
{
while( s[top] != ')')
prfx[k++]=pop();
elem=pop(); /* Remove ) */
}
else
{ /* Operator */
while( pr(s[top]) >= pr(ch) )
prfx[k++]=pop(); push(ch);
}
}
while( s[top] != '#') /* Pop from stack till empty */
prfx[k++]=pop();
prfx[k]='\0'; /* Make prfx as valid string */
strrev(prfx);
strrev(infx);
printf("\n\nGiven Infix Expn: %s \nPrefix Expn: %s\n",infx,prfx);
}
```

Out Put:

Posted by Prathuish Kumar at 19:29

G+1 +1 Recommend this on Google

Labels: C for Beginners, Conversions in C, Infix to Prefix, Infix to Prefix Conversion, Infix to Prefix Conversion using Stacks, Infix to Prefix Conversion using Stacks in C

No comments:

Post a Comment

Enter your comment...

Comment as: Select profile... ▼

Publish **Preview**

Newer Post Home Older Post

Subscribe to: Post Comments (Atom)