C & C++ WEB PROGRAMMING JAVA

LEX-YACC

PROJECTS & SEMINARS

PLACEMENT



HOME

SEMESTER 1-2 »

SEMESTER 3 »

SEMESTER 4 »

SEMESTER 5 »

SEMESTER 6 »

SEMESTER 7 »

SEMESTER 8 »

## RECURSIVE PREDICTIVE PARSER - COMPILER LAB - C **PROGRAM**

Parse the following grammar.

E --> E + T/T

```
T --> T * F / F
F --> ( E ) / id
Program:
#include<stdio.h>
#include<ctype.h>
#include<stdlib.h>
#include<string.h>
void err(const char* s)
{
perror(s);
exit(0);
}
int factor()
{
int val,i;
char ch[0];
scanf("%s",ch);
switch(ch[0])
{
case '(':
val=expr();
scanf("%s",ch);
if(ch[0]!=')')
err("Missing closing paranthesis in factor.");
break;
default:{
for(i=0;i < strlen(ch);i++)
if((ch[i]>'0')\&\&(ch[i]<='9'))
continue;
else
err("Illegal character sequence in place of factor.");
}
val=atoi(ch);
}
}
return val;
}
int term()
{
int val;
char ch[10];
val=factor();
while(1)
{
scanf("%s",ch);
if(ch[0]=='*')
```

```
PAGES
C & C++
Java
Web Programming
Assembly Language
Lisp
LEX-YACC
Projects & Seminars
Placement
```

**Blog Archive** Recent Tags

### **BLOG ARCHIVE**

- ▶ 2014 (1)
- ▶ 2013 (9)
- 2012 (59)
- ▼ 2011 (179)
  - ▶ December (2)
  - November (26)
  - October (6)
  - ► September (18)
  - ► August (36)
  - ▶ July (4)
  - ▶ June (14)
  - ► May (13)
  - ► April (12)
  - ▼ March (19)

Parser for IF-THEN Statements - YACC - LEX - Compi...

Data Communications and Networking -Behrouz A. Fo...

Computer Networking: A Top Down Approach Featuring..

Postfix Evaluaion YACC LEX S6

Expression Evaluation YACC S6

Implement Finite Automata in lex-Odd no of

Java RMI - Distributed Computing

```
val=val*factor();
else
break:
ungetc(ch[0],stdin);
return val:
int expr()
int val;
char ch[10];
val=term();
while(1)
scanf("%s",ch);
if(ch[0]=='+')
val=val+term();
else
break;
ungetc(ch[0],stdin);
return val;
}
main()
printf("\nEnter the expression: ");
printf("\n Result: %d\n",expr());
Output
nn@linuxmint ~ $ gcc rec.c
nn@linuxmint~$./a.out
Enter the expression: 5 * (3 + 1)
Result: 20
nn@linuxmint~$
You might also like:
• <u>Library Management System - C Program S3</u>
  Addition of n Numbers - Assembly Language - MASM
String Search and Replacement - Client-Server Program - Network-DBMS Lab - C Program
• Sorting Algorithms - JAVA
```

Linkwithin

Labels: C, Compiler Lab, Parser, Recursive Predictive Parser, Semester 6

# Korose & Ross - S6 DCCN Application layer slide-Select Lines Starting with 'm' - Lex Program -Com... Count The Number of Lines. Words and Characters - L... Remove All The Occurrences of a Word or Letter - L.. Count The Number of Occurrences of a Particular Wo... Replace Sequences of White Spaces by a Single Blan... Comment Removal - Lex Program - Compiler Design Number Each Line - Lex Program - Compiler Design Recursive Predictive Parser - Compiler Lab -C Pro... Comment Removal Program - Compiler Lab -C Program... Palindrome Check - Client-Server Program -Network... TCP - Client-Server Program - Networks & DBMS Lab ... ► February (27) ▶ January (2) 2010 (55) 2009 (41) 2008 (2)

## SUBSCRIBE VIA FEEDBURNER

Enter your email address

Subscribe

## 2 comments:

Anonymous October 5, 2015 at 11:03 AM blaahh Reply



Anna December 31, 2015 at 12:42 PM

Great and Useful Article.

Java Online Course

Java Online Training

Java EE training

Java Course in Chennai

Java Training in Chennai

Java Training Institutes in Chennai

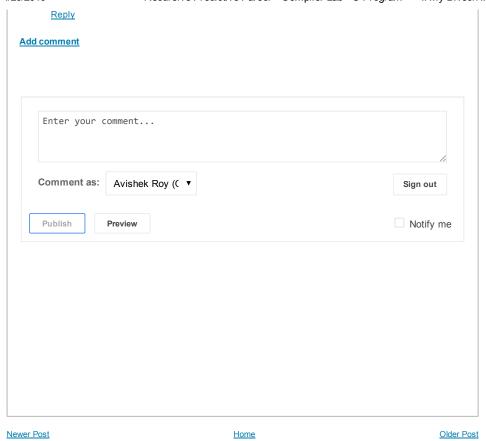
Java Interview Questions

Java Interview Questions

LIKE THIS BLOG?

**TOTAL PAGEVIEWS** 

**₩**₩₩ 902,359



Newer Post <u>Home</u>

Subscribe to: Post Comments (Atom)

■.. My B.Tech ... 2K8618.blogspot.com - 2K8CSE

www.mybtech.tk | www.2k8cse.tk | www.2k8cse.cu.cc | www.2k8cse.tk |. Powered by Blogger.