A SMART SOURCE CODE EDITORFOR C

Amit Barve[1] and Brijendra Kumar Joshi[2]

[1]Asst. Professor,CSE, VIIT Pune,India
[2]Professor, MCTE,Mhow, India

ABSTRACT

With the availability of code editors, program writing becomes very easy. The Available code editors are for writing sequential programs. These editors are unable to perform preprocessing on source code that is required for parallel lexical analysis. In this paper we present a smart editor which automatically performs preprocessing while writing a program. By the use of this editor substantial amount of time can be saved in lexical analysis process.

KEYWORDS

Parallel Lexical Analysis, Pivot locations, Constructs.

1.INTRODUCTION

With the advancement in programming languages, the program writing becomes a tedious job because of several reasons viz. various overloaded functions, support for other programming languages, deprecation etc. To solve such programming problems Integrated Development Environments (IDEs) came into existence. An IDE is a software program designed to help programmers by providing comprehensive facilities for software development. Most of the IDEs contain a source code editor, a compiler and/or interpreter. An IDE is beneficial for auto-completion, automatic code generation, refactoring, maintaining a view of files, showing errors and warnings and running unit tests etc.

Most IDEs are designed to write sequential code and follow sequential programming approach. There is still scope for writing parallel programs, preprocessing and performing parallel lexical analysis and syntax analysis. In this paper we present a smart editor which enables writing source code in a way that suits parallel lexical analysis of programs by marking pivot locations without requiring a separate pass of the source code for the purpose.

2. IMPORTANT IDEs

NetBeans IDE provides code analyzers and editors for working with Java technologies and other latest technologies like Java SE Embedded, and Java ME Embedded. The IDE also has a variety of new enhancements that further improve its support for Maven [1] and Java EE with

PrimeFaces [2]; and enhancements to C/C++ support. The detailed description about this software can be found in [3].

Eclipse [4] contains a base workspace and an extensible plug-in system for customizing the environment. It can be used to develop applications by means of several plug-ins and also be used to develop

applications in Java, Ada, ABAP, C, C++, COBOL, Fortran, Haskell, JavaScript, Lasso, Perl, PHP, Python, Ruby (including Ruby on Rails framework), Scala, Scheme etc.

Qt Creator [5] provides a cross-platform, complete IDE for application developers to create applications for multiple desktop and mobile device platforms. It is available for Mac OS X, Linux, Android and Windows operating systems.

Visual Studio [6] is a comprehensive group of tools and services to help create a wide variety of applications basically for the Microsoft platforms.

3. NEED OF A SMART EDITOR

IDE made working environment so much easier that they are used by almost every programmer now a days. The IDEs which are discussed above basically support languages like C, C++, Java etc. but for parallel lexical analysis methods proposed by Barve and Joshi [7][8][9] where preprocessing of the program plays an important role there are no suitable editors. For such type of methods a smart editor is required which will perform all necessary preprocessing tasks while keying a program and eliminates the preprocessing required.

4. PIVOT LOCATIONS IN THE PROGRAM

There can be three types of pivot locations in the program for splitting the source code into number of blocks. The pivot locations can be marked:

a) Based on constructs.
b) Based on white space characters.
c) Based on lines.

Here, by a construct we mean to highlight parts of a program that can be processed almost independently from rest of the code. Generally loops and decision making statements are considered for this purpose.

The white space character means space and tab characters only and splitting is done on the new white space character.

In Line based the splitting of source code is done into number of blocks on encountering of new line characters.

The details of detecting pivot locations can be found in [8].

5. THE SMART EDITOR

The smart editor is based on approach designed by Barve and Joshi for detecting potentially parallel constructs [9] in C programs and parallel lexical analysis of the program by marking pivot locations based on constructs, new line and white space characters [7][8]. It has a code editor and four menus File, Font, Preprocessing options and Lexical Analysis. File menu contains general file options like *New*, *Open*, *Save*, *Save As*, and *Exit*. Font menu contains font setting options like style and size. Preprocessing menu contains preprocessing options like *Newline based*, *White space character based* and *Construct based* to mark the pivot locations. The lexical analysis menu has two options *serial* and *parallel run* which perform serial and parallel lexical analysis of a program respectively.

6. EXPERIMENTAL RESULTS

The Smart editor is designed using JAVA Swing API [10]. The Figures 1-5 show the main window of the editor and all the options available in it.

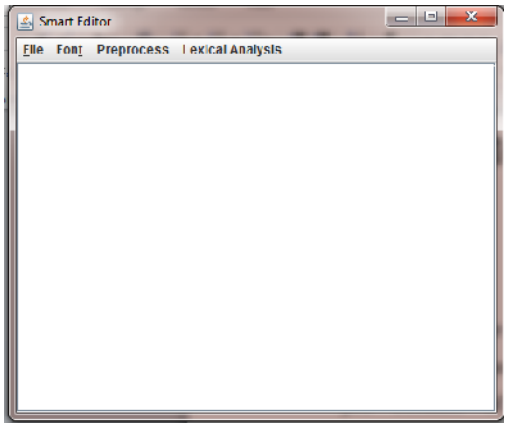

Figure.1 The Smart Editor

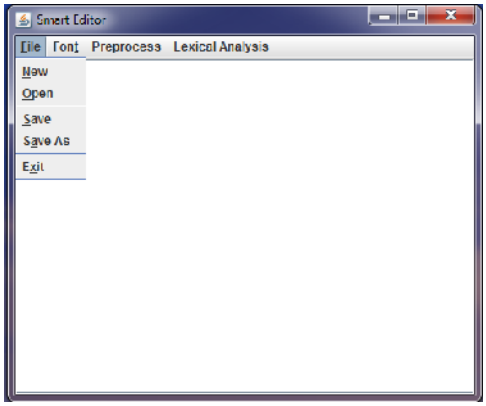

Figure.2 File Menu of Smart Editor

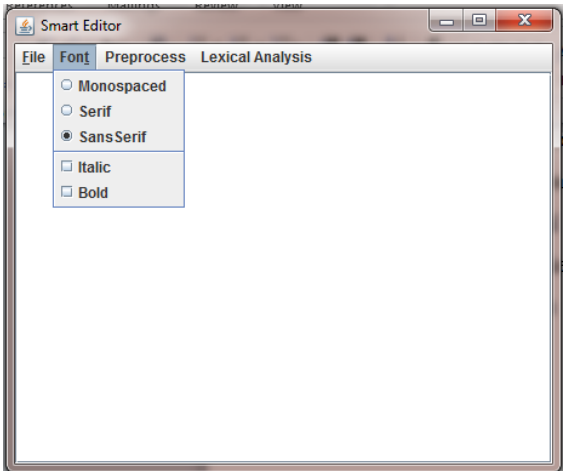

Figure.3 Font Menu of Smart Editor

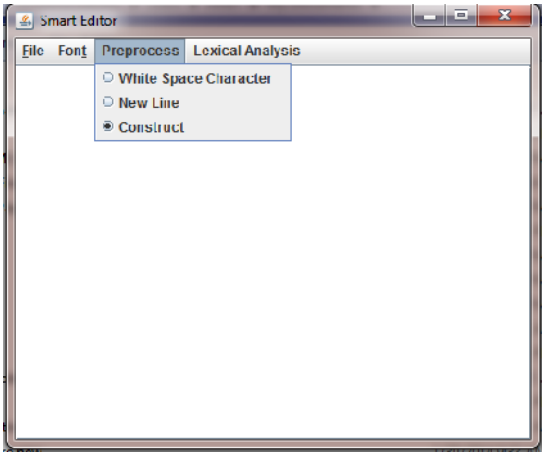

Figure.4 Preprocess Menu of Smart Editor

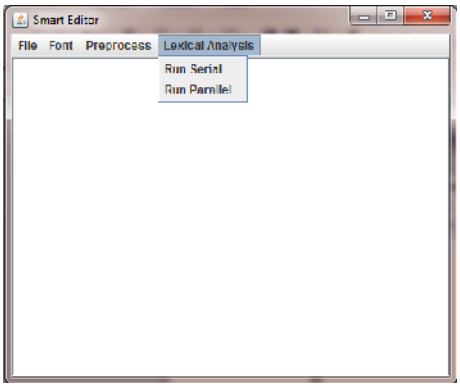

Figure.5 Lexical Analysis Menu of Smart Editor

Table 1: Preprocessing Time in construct based C programs generated automatically

No of Constructs in programs	Time taken In (Micro Second) preprocessing
1	0.71176
2	0.774586
4	1.004644
8	1.483398
16	2.745358
32	4.473972
64	13.800802
128	42.02757
256	137.62924
512	318.359199

Total 9 C programs were generated using automatic C code generation [11] and preprocessing time was calculated which is shown in Table 1. It is clear from the table that substantial amount of time can be saved by using smart editor.

7. CONCLUSION

The limitation of past work is preprocessing time required for detection of pivot locations in programs. A number of programs were written manually which require variable times as typing speeds of programmers vary. The advantage of using smart editor is that as soon as typing finishes, the preprocessing also finishes, thereby saving substantial preprocessing time. The project can be extended to IDE version and support for other languages like C++/JAVA with all compilation options.

REFERENCES

[1] maven.apache.org/
[2] primefaces.org/
[3] https://www.netbeans.org
[4] http://www.eclipse.org/ide
[5] http://qt-project.org
[6] www.visualstudio.com
[7] AmitBarve and Brijendrakumar Joshi, "A parallel lexical analyzer for multi-core machines," CONSEG- 2012 CSI Sixth International Conference on Software Engineering, pp.1,3, 5-7 Sept. 2012 doi: 10.1109/CONSEG.2012.6349505
[8] AmitBarve and Brijendrakumar Joshi, "Parallel lexical analysis on multi-core machines using divide and conquer," NUiCONE- 2012 Nirma University International Conference on Engineering , pp.1,5, 6-8 Dec. 2012 doi: 10.1109/NUICONE.2012.6493218.

[9] AmitBarve and Brijendra Kumar Joshi;"Detection of Potentially Parallel Constructs in a C Program forParallel Compilation"; Communicated to Journal of Military College of Telecommunications and DataProcessing (JMTDP); MCTE Mhow.
[10] http://docs.oracle.com/javase/7/docs/api/javax/swing/package-summary.html
[11] AmitBarve, Brijendra Kumar Joshi"Automatic C Code Generation for Parallel Compilation"; International Journal on Advance computer Theory and Engineering; pp.26-28; Vol. 2; Issue 4; 2013

Authors:

Mr. AmitBarve is an Assistant Professor in Computer Engineering at Vishwakarma Institute of Information Technology, Pune (M.H.) India. He has completed BE in Computer Science and Engineering from MIT Ujjain; M.Tech. in Computer Engineering from VJTI Mumbai. His research interests are parallel processing, HPC, and compiler design.

Dr. Brijendra Kumar Joshi is a Professor in Electronics & Telecommunication and Computer Engineering at Military College of Telecommunication Engineering, Mhow (M.P.), India. He has obtained BE in Electronics and Telecommunication Engineering from Govt. Engg. College Jabalpur; ME in Computer Science and Engineering from IISc, Banglore, and Ph.D. in Electronics and Telecommunication Engineering from Rani Durgavati University, Jabalpur, and M.Tech in Digital Communication from MANIT, Bhopal. His research interests are programming languages, compiler design, digital communications, mobile ad hoc and wireless sensor networks, software engineering and formal methods.