Practical Projects

Project #1
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and if … statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and if … statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens / Symbol Table

2- Syntax analyzer (parser phase using **Left Most Derivation** Method).

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and if … statement**)
Output : steps of **LMD** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #2
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and if … statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and if … statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Right Most Derivation** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and if … statement**)
Output : steps of **RMD** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #3
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and if … statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and if … statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Parse Tree** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and if … statement**)
Output : steps to draw **parse tree** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #4
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and if … statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and if … statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Top Down (LL) parser** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and if … statement**)
Output : steps of **LL parser** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #5
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and if … statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and if … statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Bottom Up (LR) parser** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and if … statement**)
Output : steps of **LR parser** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #6
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and for… statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and for … statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Left Most Derivation** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and for … statement**)

Output : steps of **LMD** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #7
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and for… statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and for … statement**)

1- Lexical analyzer (scanner phase)
Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Right Most Derivation** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and for … statement**)
Output : steps of **RMD** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #8
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases of Using VC++ , C# , Java, or any other computer programming language :

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and for… statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and for … statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Parse Tree** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and for … statement**)
Output : steps to draw **parse tree** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #9
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and for… statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and for … statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Top Down (LL) parser** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and for … statement**)
Output : steps of **LLparser** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #10
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and for… statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and for … statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Bottom Up (LR) parser** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and for … statement**)

Output : steps of **LR parser** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #11
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and while …..statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and while …..statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Left Most derivation** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and while …..statement**)

Output : steps of **LMD** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #12
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and while …..statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and while …..statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Right Most Derivation** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and while …..statement**)

Output : steps of **RMD** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #13
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and while …..statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and while …..statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **parse tree Method**)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and while …..statement**)
Output : steps to draw **parse tree** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #14
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and while …..statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and while …..statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Top down (LL) parser** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and while …..statement**)

Output : steps of **LL parser** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #15

Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code (**variable declaration, Arithmetic Expression, and while …..statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and while …..statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Bottom Up (LR) parser** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and while …..statement**)
Output : steps of **LR parser** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #16
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and Do …while …..statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and Do …while …..statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Left Most Derivation** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and Do …while …..statement**)

Output : steps of **LMD parser** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #17
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and Do …while …..statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and Do …while …..statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Right most derivation** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and Do …while …..statement**)

Output : steps of **RMD** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #18
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and Do …while …..statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and Do …while …..statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **parse tree Method**)
Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and Do …while …..statement**)
Output : steps to draw **parse tree** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #19
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and Do …while …..statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and Do …while …..statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Top down (LL) parser** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and Do …while …..statement**)

Output : steps of **LL parser** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #20
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code (**variable declaration, Arithmetic Expression, and Do …while …..statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and Do …while …..statement**)

1- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

2- Syntax analyzer (parser phase using **Bottom Up (LR) parser** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and Do …while …..statement**)
Output : steps of **LR parser** to show if the input string is legal or illegal using the CFG

What you should submit

1. All your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #21
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and switch….case …..statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and switch…case …..statement**)

3- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

4- Syntax analyzer (parser phase using **Left Most Derivation** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and switch….case …..statement**).

Output : steps of **LMD parser** to show if the input string is legal or illegal using the CFG

What you should submit

4. All your source files and executable file.
5. A report in which you discuss the details important to your solution.
6. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #22
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and switch….case …..statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and switch…case …..statement**)

2- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

3- Syntax analyzer (parser phase using **Right most derivation** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and switch….case …..statement**).

Output : steps of **RMD** to show if the input string is legal or illegal using the CFG

What you should submit

4. All your source files and executable file.
5. A report in which you discuss the details important to your solution.
6. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #23
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and switch….case …..statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and switch…case …..statement**)

2- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

3- Syntax analyzer (parser phase using **parse tree Method**)
Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and switch….case …..statement**).
Output : steps to draw **parse tree** to show if the input string is legal or illegal using the CFG

What you should submit

4. All your source files and executable file.
5. A report in which you discuss the details important to your solution.
6. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #24
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code(**variable declaration, Arithmetic Expression, and switch….case …..statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and switch…case …..statement**)

2- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

3- Syntax analyzer (parser phase using **Top down (LL) parser** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and switch….case …..statement**).

Output : steps of **LL parser** to show if the input string is legal or illegal using the CFG

What you should submit

4. All your source files and executable file.
5. A report in which you discuss the details important to your solution.
6. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #25
Compiler

Objective:

Design and implement a simulator to a Compiler module that simulate the Compiler job in the following two phases Using VC++ , C# , Java, or any other computer programming language:

- Assume the lexical and syntax analysis phases for a part of c-language code (**variable declaration, Arithmetic Expression, and switch….case …..statement**).
- You have to write the CFG that represent a C-language code (**variable declaration, Arithmetic Expression, and switch…case …..statement**)

2- Lexical analyzer (scanner phase)

Input : program text (file)
Output : sequence of tokens

3- Syntax analyzer (parser phase using **Bottom Up (LR) parser** Method)

Input : input string (stream of tokens that output from scanner phase) and the CFG for (**variable declaration, Arithmetic Expression, and switch….case …..statement**).

Output : steps of **LR parser** to show if the input string is legal or illegal using the CFG

What you should submit

4. All your source files and executable file.
5. A report in which you discuss the details important to your solution.
6. Test inputs and outputs should also be appended to your report.

Practical Projects

Project # 26

Assembler

Objective:

Design and implement a simulator to an assembler that simulate sub set of an assembler job using VC++ or C# programming language

The program should be able to read an assembly program from a text file and the output will be as follows:

a) The contents of the **symbol table**, memory map of **data section**, and the memory map of the **code section**. Assume that both Code and Data sections start at offset **0000h**
b) The **code size and data size** (in Bytes) needed to load and execute the code?

What you should submit

1. A CD containing all your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.

Practical Projects

Project #27

Linker

Objective:

Design and implement a simulator to a Linker module that simulate the Linker job using VC++ or C# programming language.

The program should be able to read the section tables of a set of object files and the output will be as follows:

a) The Combined Section Table (CST)

b) The public definition table (PDT)

What you should submit

1. A CD containing all your source files and executable file.
2. A report in which you discuss the details important to your solution.
3. Test inputs and outputs should also be appended to your report.