**­BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI**

Batch No. :

**DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION SYSTEMS**

**Compiler Construction (CS F363)**

**II Semester 2017-18**

**Compiler Project (Stage-2 Submission)**

**Coding Details**

**(April 20, 2018)**

*Instruction: Write the details precisely and neatly. Places where you do not have anything to mention, please write NA for Not Applicable.*

1. ID Number: 2015A7PS0163P

Name: Rahul shiv chand

1. Mention the names of the Submitted files ( Include Stage-1 and Stage-2 both)

ast.c ast.h symbol\_table.c symbol\_table.h codeGen.c codeGen.h lexer.c lexer.h lexerDef.h parser.c parser.h parserDef.h hash\_table.h hash\_table.c makefile grammar.txt testcase1.txt testcase2.txt testcase3.txt testcase4.txt testcase5.txt c1.txt c2.txt c3.txt inter\_code.c inter\_code.h set.c set.h stack.c stack.h drive.c testcase6.txt codingdetails.docx

1. Total number of submitted files: 34 (All files should be in ONE folder named exactly as your ID)
2. Have you compressed the folder as specified in the submission guidelines? (yes/no) yes
3. **Status of Code development**: Mention 'Yes' if you have developed the code for the given module, else mention 'No'.
   1. Lexer (Yes/No): yes
   2. Parser (Yes/No): yes
   3. Abstract Syntax tree (Yes/No): yes
   4. Symbol Table (Yes/ No): yes
   5. Type checking Module (Yes/No): yes
   6. Semantic Analysis Module (Yes/ no): yes (reached LEVEL 3 as per the details uploaded)
   7. Code Generator (Yes/No): yes
4. **Execution Status**:
   1. Code generator produces code.asm (Yes/ No): yes
   2. code.asm produces correct output using NASM for testcases (C#.txt, #:1-3): 1,2,3
   3. Semantic Analyzer produces semantic errors appropriately (Yes/No): yes
   4. Type Checker reports type mismatch errors appropriately (Yes/ No): yes
   5. Symbol Table is constructed (yes/no) yes and printed appropriately (Yes /No): yes
   6. AST is constructed (yes/ no) yes and printed (yes/no) yes
   7. Name the test cases out of 9 as uploaded on the course website for which you get the segmentation fault (testcase#.txt ; # 1-6 and c@.txt ; @:1-3): None
5. **Data Structures** (Describe in maximum 2 lines and avoid giving C definition of it)
   1. AST node structure

AST node structure contains the lexeme, line number, children and siblings of the corresponding token, the substructures children and sibling are used for traversal.

* 1. Symbol Table structure: HASH TABLE (OPEN CHAIN HASHING)
  2. Matrix type expression structure: SIBLINGS OF ROWS, WHERE EACH ROW IS A SIBLING OF NUMBERS.
  3. Input parameters type structure:

INPUT PARAMETER FOR AST IS ROOT OF PARSE TREE.

* 1. Output parameters type structure:

OUTPUT OF AST IS N-ARRAY COMPRESSED TREE STRUCTURE.

* 1. Structure for maintaining the three address code(if created)

LINKED LIST CONTAINING 4 FIELDS, ONE OPERATOR AND THREE AST NODE STRUCTURES.

1. **Semantic Checks:** Mention your scheme NEATLY for testing the following major checks (in not more than 5-10 words)[ Hint: You can use simple phrases such as 'symbol table entry empty', 'symbol table entry already found populated', 'traversal of linked list of parameters and respective types' etc.]
   1. Variable not Declared

CHECK IN THE CURRENT SYMBOL TABLE AS WELL AS SYMBOL TABLE OF PARENTS IF VARIABLE ENTRY EXISTS.

* 1. Multiple declarations:

CHECK IN THE CURRENT SYMBOL TABLE AS WELL AS SYMBOL TABLE OF PARENTS IN SCOPE IF VARIABLE ENTRY ALREADY EXISTS, IF YES THEN MULTIPLE DECLARATION.

* 1. Number and type of input and output parameters: EACH FUNCTION ENTRY IN SYMBOL TABLE STORES THE TYPE AND NUMBER OF ITS INPUT AND OUTPUT PARAMETERS.
  2. assignment of value to the output parameter in a function

CHECK IF INSIDE FUNCTION DEFINITION VARIABLE IS INITIALIZED.

* 1. function call semantics: Check if output parameter matches the variable given in function call output.
  2. type checking : check if variables across an operator are compatible (for example two strings are not across a MINUS operator) and if LHS type matches RHS.
  3. return semantics: using function
  4. Recursion : IF FUNCTION CALL IS MADE INSIDE A SCOPE OF THE SAME FUNCTION THEN RECURSION IS CAUGHT.
  5. module overloading: IF FUNCTION IS ALREADY DEFINED IN CURRENT SCOPE OR SCOPE OF PARENTS.
  6. 'If' semantics : CHECK IF TOKENS INSIDE IF ARE ONLY OF TYPE NUM OR MATRIX ELEMENT .
  7. Matrix semantics and type checking of matrix type variables:

Using the row and column information stored of each variable in symbol table.

* 1. register allocation (your manually selected heuristic)

EBP is used as base pointer for accessing memory locations, while EAX is used for intermediate operations.s

* 1. Scope of variables and their visibility : WE STORE THE SCOPE OF VARIABLES IN THE SYMBOL TABLE.

1. **Compilation Details**:
   1. Makefile works (yes/No): YES
   2. Code Compiles (Yes/ No): YES
   3. Mention the .c files that do not compile: NONE
   4. Any specific function that does not compile: NONE
   5. Ensured the compatibility of your code with the specified gcc version(yes/no) YES
2. **Driver Details**: Does it take care of the options specified earlier?(yes/no):YES
3. Specify the language features your compiler is not able to handle (in maximum one line)

NONE

1. Are you availing the lifeline (Yes/No): YES
2. Write exact command you expect to be used for executing the code.asm using NASM simulator [We will use these directly while evaluating your NASM created code]

nasm -f elf code.asm

ld -m elf\_i386 code.o -o code

./code

1. **Strength of your code**(Strike off where not applicable): (a) correctness (b) completeness (c) robustness (d) Well documented (e) readable (f) strong data structure (f) Good programming style (indentation, avoidance of goto stmts etc) (g) modular (h) space and time efficient
2. Any other point you wish to mention:

THE CODE GENERATION WAS DONE USING A MODULE WHICH FIRST GENERATES THE INTERMEDIATE CODE.

1. **Declaration:** I, RAHUL SHIV CHAND (your name) declare that I have put my genuine efforts in creating the compiler project code and have submitted the code developed by me. I have not copied any piece of code from any source. If my code is found plagiarized in any form or degree, I understand that a disciplinary action as per the institute rules will be taken against me and I will accept the penalty as decided by the department of Computer Science and Information Systems, BITS, Pilani.

Sign:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ID 2015A7PS0163P

Name:RAHUL SHIV CHAND

Date: 21/04/2018

-------------------------------------------------------------------------------------------------------------------------------------------------

/\*not to exceed three pages\*/