BITS Pilani, Hyderabad Campus
Department of Computer Science and Information Systems
Second Semester, 2024-25
CS F363 Compiler Construction
Project part 3 (Final part) (12 Marks)

The description of the toy language is the same as given in Part 1.

1 Tasks

1. ***[6 marks]*** Write a Bison program to generate three-address code (3AC) for the given input program.

2. ***[6 marks]*** Finally, extend your Bison program to print the output of the given input program. Also, report any semantic errors like type mismatch, undeclared variables, or using a variable before a value is set to it. Further, print the symbol table.

   One way to print the output of the given program is to simulate the AST of the input program. The idea of simulating the AST (Abstract Syntax Tree) means that once your parser (using Bison) constructs an AST for the input program, you can walk through or "interpret" this tree to evaluate expressions, execute assignments, handle conditionals, etc.—basically simulating what would happen if the program were actually executed.

   Note: You must simulate the execution of the program by traversing the AST within your own Bison-based interpreter. You are not allowed to translate the parsed input program into another programming language like C, C++, Python, etc., and then run it using a compiler or interpreter for that language.

   Any such attempt to offload execution to another language will result in a score of 0 out of 32 marks for the entire project.

2 Input and output format

The input program will be given in a text file, whose name will be given at the runtime. Use the following code for the same purpose.

```
int main(int argc, char *argv[]) {
    if (argc != 2) {
        fprintf(stderr, "Usage: %s <input file>\n", argv[0]);
        return 1;
    }
    yyin = fopen(argv[1], "r");
    if (!yyin) {
        perror("Error opening file");
        return 1;
    }
    yyparse();
    fclose(yyin);
    return 0;
}
```

The output should be displayed on the screen.

3 Submission guidelines

1. You work on the assignment with your group members and are strongly discouraged from discussing it with others.
2. You can refer to the Internet/Web resources only to understand the syntax of the Bison/Flex tools.
3. Taking code from the Internet/Web is strictly prohibited. If you do, it will lead to severe penalties.
4. Create a zip file with the following folders/files in it.
   (a) Create separate folders for the 3AC and Symbol table. Keep the respective files and all necessary files in the folders.
   (b) Create a `make` file to compile bison and flex codes and to generate an executable file (let it generate a default file `a.out`). Keep the `make` file in both folders.
   (c) A PDF file that contains the information of the group members (BITS ID, name, and email ID).
5. Due date is 27 April 2025, 11:59 PM. Late submissions will be allowed up to 12 hours after the deadline with a penalty of `2%` per hour.